

SESSION
SIMULATION AND APPLICATIONS

Chair(s)

TBA

Kinematics-Based Simulation and Animation of Articulated Rovers Traversing Rough Terrain

Mahmoud Tarokh

Department of Computer Science, San Diego State University
San Diego, CA 92182-7720, U.S.A.

Abstract

The paper proposes a simple and computationally efficient method for the modeling and animation of highly articulated rovers traversing rough terrain. The method is based on the propagation of position and orientation velocities through wheels and various joints and linkages of the rover. These velocity equations are combined to form the Jacobian matrix of the rover that relates the position and orientation of the rover to various active (actuated) and passive rover joint variables. A rearrangement of the Jacobian equation allows determining the actuation of suspension joints for balance control to avoid tipover. This is done through a pseudo-inverse method which optimizes a balance performance criterion. To illustrate the kinematics modeling and balance control concepts, the method is applied to a rover similar to the NASA's Sample Return Rover and simulation and animation results are presented.

1. Introduction

Articulated rovers are being used in variety of applications, especially in planetary explorations. Rovers with active suspension mechanisms are capable of modifying their configurations by adjusting their suspension linkages and joints so as to change their center of mass, thus avoiding tipover while traversing rough and sloppy terrain.

Rovers with adjustable suspension system have been researched in recent years. Iagnemma and Dubowsky [1] presented stability-based suspension control for a specific rover using an essentially geometric approach and performing a rather complex optimization procedure. Other contributions in the area of rover kinematics include [2] and [3]. We have developed a comprehensive method for rover kinematics modeling [4] and included motion control in [5]

In a subsequent paper [6], we presented an alternative and more efficient method for kinematics modeling and balance control. The purpose of the present paper is to demonstrate the modeling and control using simulations and animation of a rover over uneven terrain.

2. Overview

In this section we provide an overall view of our proposed simulation and animation environment which has been developed in Matlab/Simulink. The simulations consist of a number of modules as shown in Fig. 1. The trajectory generation module provides either a time trajectory for the desired rover position (x, y) and heading, or a desired path. These quantities are assumed to be available from a path planner. These desired quantities are compared with their respective sensed values in a trajectory following module that produces the desired forward speed and turn (yaw) rate of the rover. The actuation or control module then receives these speed and turn rate commands as well as the current sensed rover roll, pitch and suspension systems joint angles to determine the actuation commands for the rover steering, wheel motors and active suspension system joints. These commands are applied to the rover model which interacts with the terrain and produces the actual (sensed) rover quantities. Since trajectory generation and trajectory following have been covered in the literature, we will not discuss these modules in this paper.

Animation consists of drawing the terrain, rover body, suspension system, wheels and their interaction with the terrain. The animation module receives various sensed quantities from the rover simulation and incrementally moves and displays the motion in a smooth manner.

3. Kinematic Based Actuation and Control

In this section we develop a kinematics model for and articulated rover with active suspension system (ARAS), which will be used to determine the actuation commands. Such a rover is a wheeled mobile robot consisting of a main body connected to wheels via a set of linkages and joints that can be adjusted, some actively and some passively for keeping the rover balanced. The active linkages and joints have actuators through which their values can be controlled, whereas passive ones change their values to comply with the terrain topology.

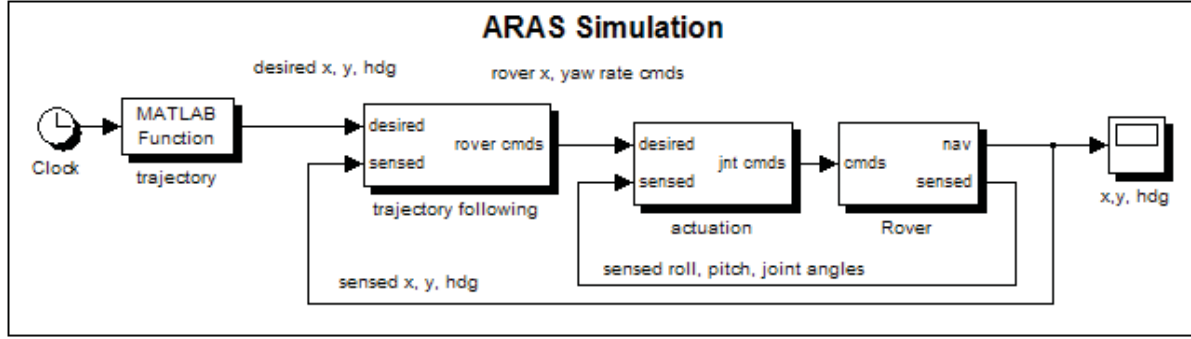


Fig. 1 The main simulation modules

For modeling, we must determine the contributions of each wheel and suspension mechanism to the overall motion of the rover. We attach a number of frames starting from the wheel-terrain contact frame then going through the steering and suspension frames and finally to the rover reference frame. Since we are interested in the motion, we relate the translational and rotational velocities of the next frame in terms of the previous frame. Let $u_a = [x_a \ y_a \ z_a]^T$ and $u_b = [x_b \ y_b \ z_b]^T$ denote the position of the current and next frames, respectively. Similarly, let $\phi_a = [\alpha_a \ \beta_a \ \gamma_a]^T$ and $\phi_b = [\alpha_b \ \beta_b \ \gamma_b]^T$ be the orientation of the current and next frames, respectively, where α, β and γ are the rotation around x, y and z axis, or roll, pitch and yaw, respectively. The 3×1 translation velocity vector of the next frame b is dependent on the translational and rotational velocities of the current frame a plus any translational velocity added to the frame b itself. This can be written as [7]

$$\dot{u}_b = R_{b,a}(\dot{u}_a + \dot{\phi}_a \times p_b) + \dot{u}_{ob} \quad (1)$$

where $R_{b,a}$ and p_b are, respectively, the rotation matrix and position vector of the frame b relative to the frame a , and \dot{u}_{ob} is the translational velocity added to the frame b . The latter is zero if the joint associated with the frame b is not prismatic. The rotational velocity of the next frame b is dependent on the rotational velocity of the frame a plus any rotational velocity $\dot{\phi}_{ob}$ added to the frame b itself, i.e. [7]

$$\dot{\phi}_b = R_{b,a} \dot{\phi}_a + \dot{\phi}_{ob} \quad (2)$$

We start at wheel i ($i = 1, 2, \dots, n$) contact frame c_i which has the translational and rotational velocities $\dot{u}_{ci} = [\dot{x}_{ci} \ \dot{y}_{ci} \ \dot{z}_{ci}]^T$ and $\dot{\phi}_{ci} = [\dot{\alpha}_{ci} \ \dot{\beta}_{ci} \ \dot{\gamma}_{ci}]^T$, and perform the frame to frame velocity propagation until we reach to the rover reference frame to obtain rover velocities

$\dot{u}_r = [\dot{x}_r \ \dot{y}_r \ \dot{z}_r]^T$ and $\dot{\phi}_r = [\dot{\alpha}_r \ \dot{\beta}_r \ \dot{\gamma}_r]^T$. Let the joint variable vector that includes each wheel-terrain contact

angle, steering angle, and various prismatic and revolute joint variables be denoted by the $v_i \times 1$ vector η_i . Then we will obtain an equation of the general form

$$\begin{pmatrix} \dot{u}_r \\ \dot{\phi}_r \end{pmatrix} = J_i \begin{pmatrix} \dot{u}_{ci} \\ \dot{\phi}_{ci} \\ \dot{\eta}_i \end{pmatrix}; \quad i = 1, 2, \dots, n \quad (3)$$

where J_i is the Jacobian matrix of the wheel i . Note that the wheel translational and rotational velocity vectors \dot{u}_{ci} and $\dot{\phi}_{ci}$ include various slips.

Equation (3) describes the contribution of individual wheel motion and the connecting joints to the rover body motion. The net body motion is the composite effect of all wheels and can be obtained by combining (3) into a single matrix equation as

$$\begin{pmatrix} I_6 \\ \vdots \\ I_6 \end{pmatrix} \begin{pmatrix} \dot{u}_r \\ \dot{\phi}_r \end{pmatrix} = J \begin{pmatrix} \dot{u}_c \\ \dot{\phi}_c \\ \dot{\eta} \end{pmatrix} \quad (4)$$

where the composite identity matrix on the left is $6n \times 6$, $\dot{u}_c = (\dot{u}_{c1} \ \dot{u}_{c2} \ \dots \ \dot{u}_{cn})^T$ and $\dot{\phi}_c = (\dot{\phi}_{c1} \ \dot{\phi}_{c2} \ \dots \ \dot{\phi}_{cn})^T$ are $3n \times 1$ vectors of composite wheel velocities at the contact points, and $\dot{\eta}$ is the $v \times 1$ vector of the joint variables which has both active (actuated) and passive joints. Note that in general some wheels share common suspension links and joints so that $v \leq \sum_{i=1}^n v_i$. The composite Jacobian matrix of the rover J has a dimension of $6n \times (6n + v)$.

The composite equation (4) reflects the contribution of various position and angular rates to the overall motion of the rover. In order to control, we must determine commands to the wheels, steering and joints actuators. For this, we rearrange (4) into an equation of the form

$$A \dot{\chi} = B \dot{q} \quad (5)$$

where $\dot{\chi}$ is the $n_x \times 1$ vector of unknown quantities to be determined, and \dot{q} is the $n_q \times 1$ vector of known quantities.

The unknown vector consists of actuation signals such as active suspension joints, wheel roll rates, and un-measurable quantities such as appropriate slips. The known vector consists of desired quantities such as the desired forward rover velocity \dot{x}_d and heading $\dot{\gamma}_d$. The matrices A and B are obtained from the elements of J and the identity matrices I_1, I_2, \dots, I_6 in (4). After partitioning (4) into the form (15), the dimensions of A and B are $6n \times n_x$ and $6n \times n_q$. In general $n_x > n_q$, in which case we have an underdetermined system of equations. The extra parameters can be used to achieve an optimization goal. For a rover with an active suspension system, the optimization goal would be balancing the rover by adjusting its suspension joints and linkages so as to avoid tip over when traversing rough terrain. A possible optimization function is

$$f = a_1 \mu + a_2 \|\eta_a - \hat{\eta}_a\| \quad (6)$$

The first term is a tip over measure which reflects the degree to which the rover configuration deviates from the perfectly balanced configurations. The latter occurs when the rover moves over a flat surface. The quantity η_a in the second term is the vector of actuated suspension joints and $\hat{\eta}_a$ its nominal or desired values under normal operating conditions (e.g., operating over flat surface). Note that without the second term, minimizing f would result in a rover configuration that is maximally flat or spread out even when the rover moves over a flat surface. The weighting factors a_1 and a_2 place relative emphasis between achieving rover balancing and the desire to operate near the nominal configuration.

We solve (5) and minimize (6) by using the null space of A , to get [8]

$$\dot{\chi} = A^\# B \dot{q} - k(E - A^\# A) \begin{pmatrix} \partial f / \partial \sigma \\ 0 \end{pmatrix} \quad (7)$$

where $A^\#$ is the pseudo-inverse of A , k is a scalar, E is $n_x \times n_x$ identity matrix, and the partial derivative are gradients of the performance function with respect to the active suspension joints, roll and pitch, respectively. In the next section we specify the above quantities for our ARAS. The gradient can be computed numerically or analytically or numerically.

4. Modeling of an Active Suspension Rover

The articulated rover with active suspension (ARAS) to be considered here is similar to the JPL Sample Return Rover shown in Fig.2. The schematic diagram of ARAS to be

analyzed is shown in Fig. 3. The rover has four wheels with each independently actuated and rotation angles subscripted with a clockwise direction so that θ_1, θ_4 are for the left side and θ_2, θ_3 are for the right side. At either side of the rover, two legs are connected via an adjustable hip joint. In Fig. 3 the hip angles on the left and right sides are denoted as $2\sigma_1$ and $2\sigma_2$, respectively. These joints are actuated and used for balancing the rover. The two hips are connected to the body via a differential which has an angle ρ on the left side and $-\rho$ on the right side. On a flat surface ρ is zero but becomes non-zero when one side moves up or down with respect to the other side. The differential joint ρ is passive (unactuated) and provides for the compliance with the terrain. The wheels are steerable with steering angles denoted by ψ_i . The wheel terrain contact angle δ_i is the angle between the z-axes of the i-th wheel axle frame A_i and contact coordinate frame c_i as shown in Fig. 4.



Fig. 2 The JPL's sample return rover

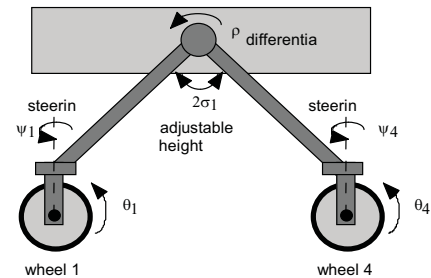


Fig. 3 Schematic diagram of the left side of ARAS

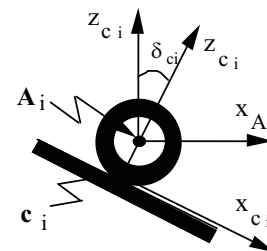


Fig. 4 Definition of contact angle

In order to derive the kinematics equations, we must assign coordinates frames. Fig. 5 illustrates our choice of

coordinate frames for the left side of the rover. The right side is assigned similar frames. In Fig. 5, R is the rover reference frame whose origin is located on the center of gravity of the rover, its x-axis along the rover straight line forward motion, its y-axis across the rover body and its z-axis represents the up and down motion. The differential frame D has a vertical (along z-axis) offset denoted by k_1 and a horizontal distance of k_2 from D. The distance from the differential to the hip, denoted by k_3 , is half the width of the rover. We now introduce three more frames, all of which have origin at the wheel axle. The length of the legs from the hip to the wheel axle is k_4 . The hip frames H_1, \dots, H_4 for the four wheels are obtained from the differential frame by rotation and translation as shown with the Denavit-Hartenberg (D-H) parameters $\gamma_{dh}, d_{dh}, a_{dh}$ and α_{dh} in Table 1 and in Fig 5. Similarly the steering frames S_1, \dots, S_4 and axle frames A_1, \dots, A_4 are defined in Table 1 and Fig 5.

We now use the basic frame to frame equations (1)-(2) and go through the frames sequentially from wheel i terrain contact c_i , wheel axle A_i , steering S_i , hip H_i , differential D, and finally to the rover reference R. Equation (1)-(2) for the contact to the axle becomes

$$\begin{aligned} \dot{u}_{Ai} &= R_{Ai,ci} (\dot{u}_{ci} + \dot{\phi}_{ci} \times (0 \ 0 \ r)^T) \\ \dot{\phi}_{Ai} &= R_{Ai,ci} \dot{\phi}_{ci} + (0 \ -\dot{\sigma}_{ci} \ 0)^T \end{aligned} \quad (8)$$

where the rotation matrix is $R_{Ai,ci} = \begin{pmatrix} c\delta_{ci} & 0 & s\delta_{ci} \\ 0 & 1 & 0 \\ -s\delta_{ci} & 0 & c\delta_{ci} \end{pmatrix}$, as

evident from Fig. 4.

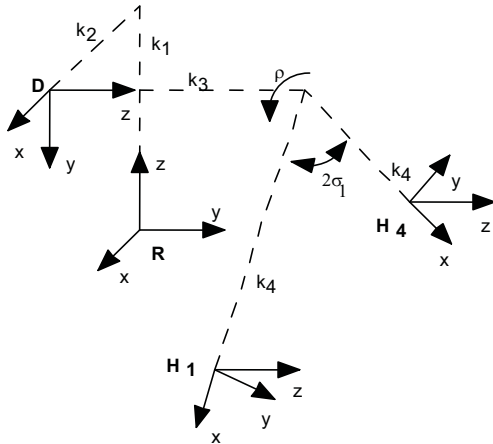


Fig. 5 Reference R, differential D, and hip H coordinate frames

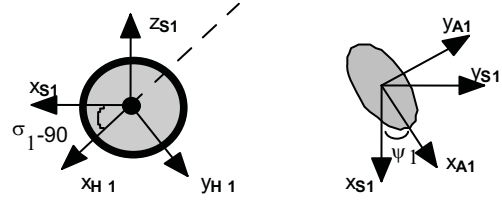


Fig. 6 Side (left figure) and top views of wheel 1

TABLE 1- D-H PARAMETERS FOR THE ARAS

Frame	γ_{dh}	d_{dh}	a_{dh}	α_{dh}
D	0	k_1	k_2	-90
H1	$90 - \sigma_1 + \rho$	k_3	k_4	0
H2	$90 - \sigma_2 - \rho$	$-k_3$	k_4	0
H3	$90 + \sigma_3 - \rho$	$-k_3$	k_4	0
H4	$90 + \sigma_4 + \rho$	k_3	k_4	0
S1	$\sigma_1 - 90$	0	0	90
S2	$\sigma_2 - 90$	0	0	90
S3	$\sigma_3 - 90$	0	0	90
S4	$\sigma_4 - 90$	0	0	90
A1	ψ_1	0	0	0
A2	ψ_2	0	0	0
A3	ψ_3	0	0	0
A4	ψ_4	0	0	0

Next we form wheel i axle to steering velocity propagation as

$$\begin{aligned} \dot{u}_{Si} &= R_{Si,Ai} (\dot{u}_{Ai} + \dot{\phi}_{Ai} \times (0 \ 0 \ 0)^T) \\ \dot{\phi}_{Si} &= R_{Si,Ai} \dot{\phi}_{Ai} + (0 \ 0 \ -\dot{\psi}_i)^T \end{aligned} \quad (9)$$

where $R_{Si,Ai} = \begin{pmatrix} c\psi_i & -s\psi_i & 0 \\ s\psi_i & c\psi_i & 0 \\ 0 & 0 & 1 \end{pmatrix}$. The next in the chain is

the hip frame, and we can write

$$\begin{aligned} \dot{u}_{Hi} &= R_{Hi,Si} (\dot{u}_{Si} + \dot{\phi}_{Si} \times (0 \ 0 \ 0)^T) \\ \dot{\phi}_{Hi} &= R_{Hi,Si} \dot{\phi}_{Si} + (0 \ 0 \ -h_i \dot{\sigma}_i)^T \end{aligned} \quad (10)$$

with $R_{Hi,Si} = \begin{pmatrix} s(h_i \sigma_i) & 0 & -c(h_i \sigma_i) \\ -c(h_i \sigma_i) & 0 & s(h_i \sigma_i) \\ 0 & 1 & 0 \end{pmatrix}$, $\sigma_4 = \sigma_1$, $\sigma_3 = \sigma_2$, and $h_i = \begin{cases} 1 & i = 1, 2 \\ -1 & i = 3, 4 \end{cases}$. Similarly the differential

frame, and rover reference frame velocities are obtained using (1)-(2) and Table 1.

Substituting recursively (5) through (8) into (9) we obtain an equation of the form (3) where $\dot{\eta}_i = \begin{bmatrix} \dot{\rho}_i & \dot{\sigma}_i & \dot{\psi}_i & \dot{\delta}_{ci} \end{bmatrix}^T$. Due to space limitation, the Jacobian matrices J_i and their elements are not given here but can be found in our technical report [9]. The elements of J_i are trigonometric functions of the joint variables ρ_i, σ_i, ψ_i and δ_{ci} .

5. Simulation and Animation Results

In this section we present the results of balance control for the ARAS introduced in Section II.A.. The full Jacobian equation is not given here due to space limitations but is provided in [9]. For the ARAS, the vector $\dot{\chi}$ in (5) is

$$\dot{\chi} = \begin{bmatrix} \dot{y}_r & \dot{z}_r & \dot{\alpha}_r & \dot{\beta}_r & \dot{\sigma} & \dot{\rho} & \dot{u}_c & \dot{\phi}_c \end{bmatrix}^T \quad (11)$$

where \dot{y}_r, \dot{z}_r and $\dot{\alpha}_r, \dot{\beta}_r$ are the unknown rover velocities and attitude angles rates and $\dot{\sigma} = \begin{bmatrix} \dot{\sigma}_1 & \dot{\sigma}_2 \end{bmatrix}^T$ is the actuated hip rate vector. Note that \dot{u}_c contains only the wheel rolling rates \dot{x}_i and side slip rate \dot{y}_i , and wheels are assumed to be in contact with the terrain so that $\dot{z}_i = 0$. In addition, $\dot{\phi}_c$ consist of only wheel turn slip rates and tilt rate, with other components are removed due to the particular geometry of the rover. Finally; $\dot{\delta}_c$ is set to zero as contact angle rates are very noisy and their estimation is prone to large errors. Thus $\dot{\chi}$ is a 23×1 vector and A is a $6n \times 23 = 24 \times 23$ matrix. Note that in general some rows of A are linearly dependent and thus $\text{rank}(A) \leq 24$. The vector of the known quantities in (5) is

$$\dot{q} = \begin{bmatrix} \dot{x}_d & \dot{y}_d & \dot{\psi} \end{bmatrix}^T \quad (12)$$

where \dot{x}_d and \dot{y}_d are the desired (specified) rover forward velocity and heading (yaw) rate, respectively, and $\dot{\psi} = \begin{bmatrix} \dot{\psi}_1 & \dot{\psi}_2 & \dot{\psi}_3 & \dot{\psi}_4 \end{bmatrix}^T$ is the steering angle rates vector. Note that since the axes of steering and wheel turn slip are coincident in this rover, the steering angles are indistinguishable from turn slip. In this case, a geometric approach is used to determine the steering angle rates $\dot{\psi}$ using the desired forward velocity \dot{x}_d and \dot{y}_d , and thus $\dot{\psi}$ is a known quantity. Thus the dimension of the known vector q is 6×1 , and B has the dimension 24×6 .

We have developed a simulation and animation environment using Matlab/Simulink. The overall scheme is was described in Section 2. Fig 7 and Fig. 8 show the simulation environment where various quantities such as rover position and orientation, and various joint angle values are shown during the traversal. Various views of the rover can be observed.

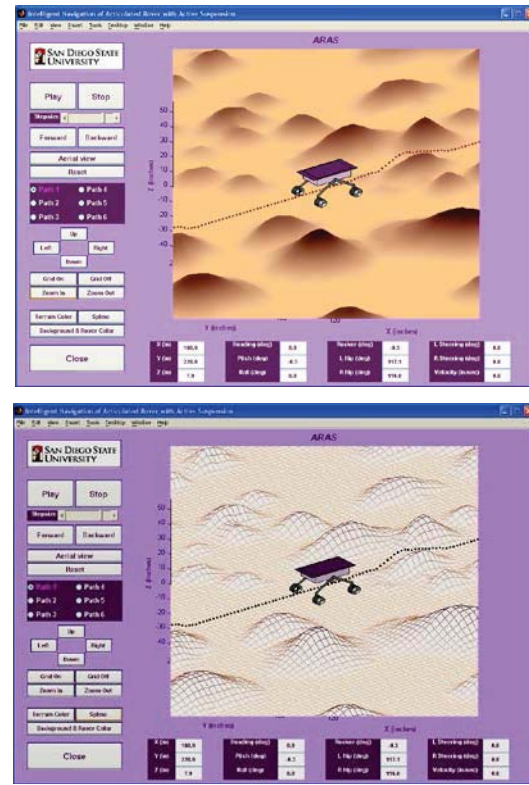
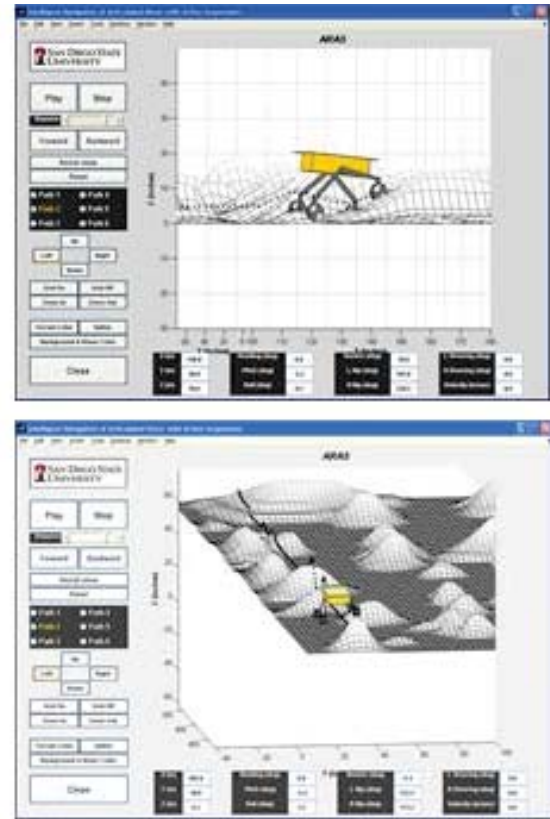


Fig. 7 The simulation environment



Side view/Front view

Fig. 8 Side and front views

We study the behavior of the rover for three terrains. The rover speed is set at 1 cm/s to ease the conversion between distance and time on the graphs.

Case 1: Inclined Terrain

The terrain and the trace of the rover wheels are shown in Fig. 9. The terrain is flat but has a 45 degree slope which could result in tipover without actuated suspension. The hip angles start at their nominal values, e.g. $2\sigma_1 = 2\sigma_2 = 90$ degrees. The hip joint angles as given in Fig. 10 shows that the right joint has decreased to raise the right side but the left angle is increased to lower the left side. This has almost leveled the rover as is evident by the rover roll angle shown in Fig. 11 where the initial roll angle of about 38 degrees has been reduced to about 13 degrees.

Case 2: Wavy and Bumpy Terrain

The terrain shown in Fig. 12 consists of a bump which is wavy (sinusoidal) under the left wheels and smooth under the right wheels. The hip joint angles are depicted in Fig. 13. It is interesting to note that the right and left hip joint values also go through sinusoidal type changes but in the opposite directions to maintain a level and balanced rover body. The rocker also shows sinusoidal behavior. The rover roll and pitch angles are seen in Fig. 14. The rover roll exhibits small changes due to the hip adjustments, whereas the rover pitch goes through relatively large variations due to the traversing up and then down the wavy bump.

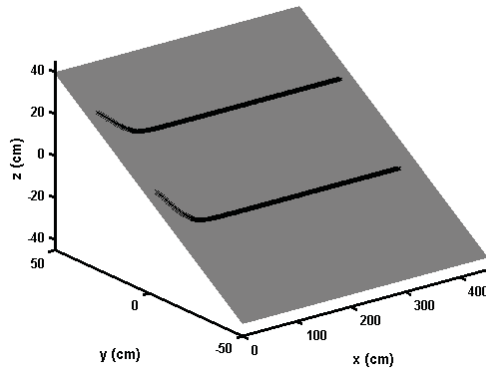


Fig. 9 Inclined terrain and traces of rover wheels

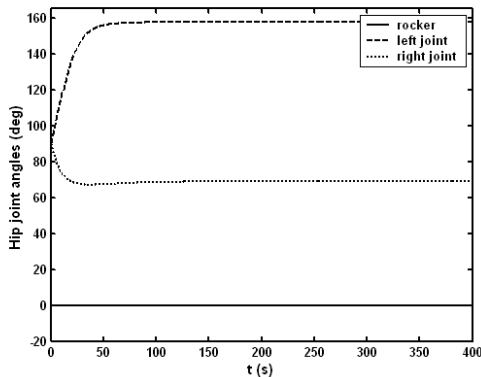


Fig. 10 Hip joint angle trajectories

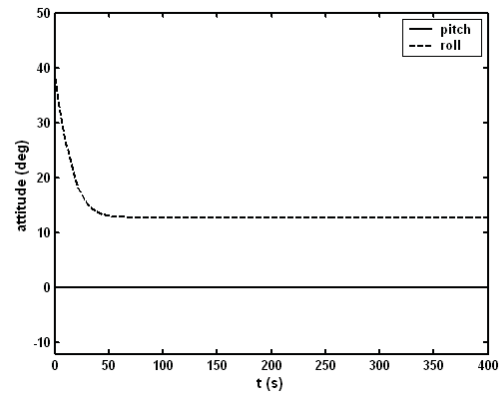


Fig. 11 Rover body roll and pitch angle profiles

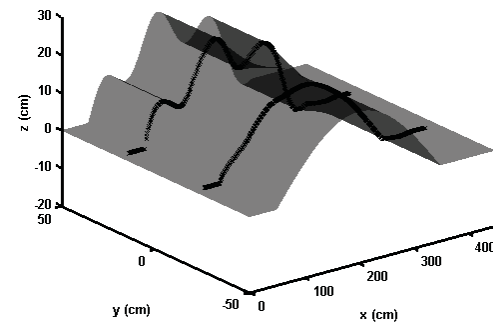


Fig. 12 Wavy and bumpy terrain

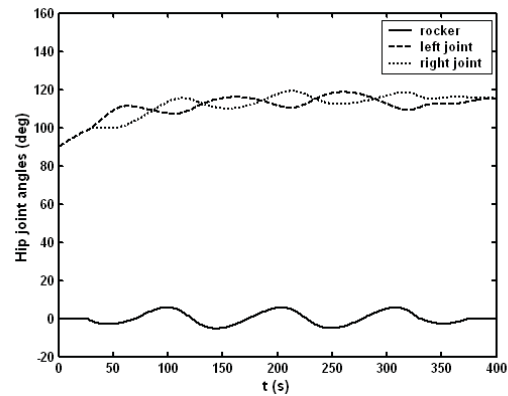


Fig. 13 Hip joint angle trajectories

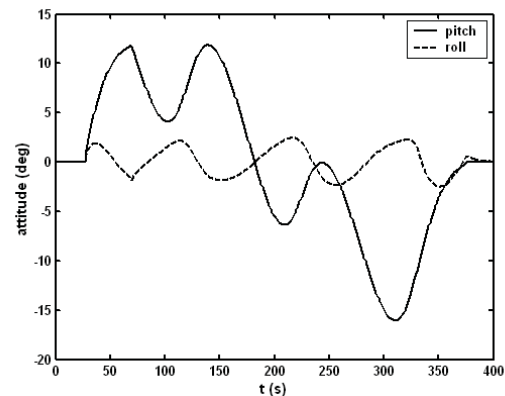


Fig. 14 Rover roll and pitch trajectories

Case 3: Inclined Ditch and Bump Terrain

In this case, the terrain has a slope of 45 degrees with a bump under the left side wheels and a ditch under the right side wheels as shown in Fig. 15. The hip angles are adjusted accordingly to balance the rover as shown in Fig. 16. The rover exhibits a maximum roll angle of about 16 degrees as seen from Fig. 17; much smaller than the 45 degree terrain slope.

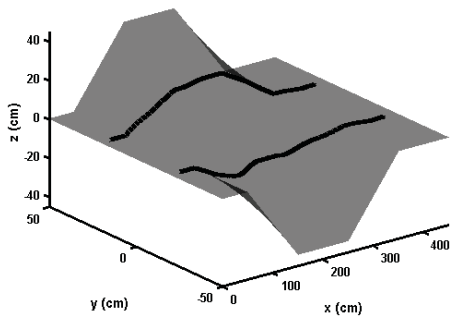


Fig. 15 Inclined ditch and bump terrain

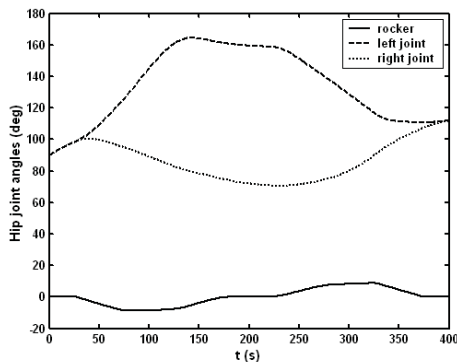


Fig. 16 Hip joint angle trajectories

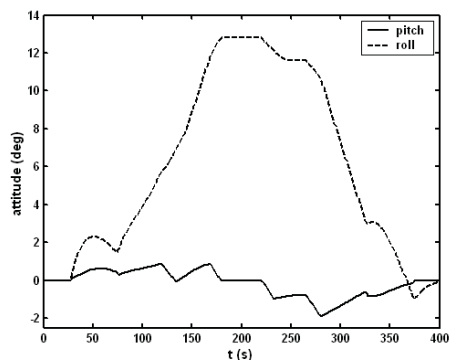


Fig. 17 Rover roll and pitch trajectories

6. Conclusions

A methodology is presented for the kinematics modeling and control of articulated rovers for achieving rover balance when traversing rough and sloppy terrain. The main feature of the work is the formulation of rover kinematics which uses simple velocity propagation starting from wheel-terrain contact and going through various joints and linkages to finally reach the rover reference frame. This formulation makes the modeling and computer implementation very efficient through simple repeated function calls. Rover balance control is achieved through a pseudo-inverse method which optimizes balance criterion. Simulation and animation package using Matlab have been developed that show the motion of the rover over rough terrain.

References

- [1] Iagnemma, K. and S. Dubowski, "Traction Control of wheel mobile robots in rough terrain with applications to planetary rovers," *Int. J. Robotics Res.*, vol. 23, No. 10-11, pp. 1029-1040, 2003.
- [2] H. D. Nayar, I. A. D. Nesnas, "Re-usable kinematics models and algorithms for manipulators and vehicles", *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, San Diego, 2007.
- [3] J. Balaram, "Kinematic observers for articulated rovers," *Proc. IEEE Int. Conference on Robotics and Automation*, pp. 2597-2604, San Francisco, CA, 2000
- [4] M. Tarokh, M. and G. McDermott, "Kinematics Modeling and Analysis of Articulated Rovers", *IEEE Trans. Robotics*, vol. 21, No. 4, 539-553, 2005.
- [5] G. McDermott, M. Tarokh, L. Mireles, "Balance control of articulated rovers with active suspension systems," *Proc. 7th IFAC Int. Conf. on Robot Control*, vol 8, Pt. 1, identifier: 10.3182/20060906-3-IT-2910.00121, Italy, Sept. 2006.
- [6] M. Tarokh, H.D. Ho and A. Bouloubasis, "Systematic kinematics analysis and balance control of high mobility rovers over rough terrain," *J. Robotics and Autonomous Systems*, vol. 61, pp. 13-24, 2013.
- [7] Craig, J. *Introduction to Robotics, Mechanics and Control*, pp.144-146, Pearson Prentice-Hall, 2005.
- [8] Nakamura, N., *Advanced Robotics- Redundancy and Optimization*, Chapt. 4, Addison and Wesley, 1991.
- [9] Mireles, L., G. McDermott and M. Tarokh, "Two approaches to kinematics modeling of articulated rovers", <http://www-rohan.sdsu.edu/~tarokh/lab/publications.html>.

Design of Multithreaded Simulation Software through UML for a Fully Automated Robotic Parking Structure

J. K. Debnath and G. Serpen

Electrical Engineering & Computer Science, University of Toledo, Toledo, Ohio, USA

Abstract - This paper presents UML-based design and development of simulation software for a multi-story car parking structure that is fully automated. The software is conceived to simulate parking demand including morning and evening rush hours at the center of a metropolitan city, movement of cars loaded onto robotic autonomous carts between floors, scheduling of elevators, and path planning for carts to and from their parking spots across a floor. There are two distinct software modules. One module models the parking demand around the clock by generating parking requests through statistical modeling, movement of carts and elevators in compliance with the laws of physics, and breakdowns of carts. The second module models the assignment, planning, scheduling and optimization algorithms for robotic carts and elevators to minimize the average customer waiting times.

Keywords: UML, simulation software design, multithreaded, automated robotic parking structure, path planning, scheduling, optimization

1 Introduction

The space utilization rate of conventional parking structures with driving lanes is low and inefficient [5]. For busy urban areas where real estate space is at premium, the utilization rate issue with the conventional parking structures is of high prominence. Approaches to increase the space utilization rate are of current interest. Automated robotic parking structures with no driving lanes offer a promising option to explore along this venue. Automated robotic parking structures do not need dedicated driving lanes like conventional parking structures, as they are equipped with robotic mechanisms to facilitate fully automated storage and retrieval for vehicle parking. However, an automated and multi-storied parking structure needs sophisticated space assignment, path planning, and scheduling and optimization algorithms to manage storage and retrieval processes effectively and efficiently. Simulation is a very effective tool to test and evaluate such complex planning and scheduling algorithms for automated parking structures. In this paper, we describe a real time simulation model based on unified modeling language in order to test and evaluate intelligent assignment, path planning and resource scheduling algorithms developed for chess-type [1][2][3][4] (or puzzle type) multi-story, robotic, and fully automated parking structures.

2 Multi-story parking structure

A chess type parking structure may have multiple floors, where each floor is considered as a rectangular area shown in Figure.1. There are no roadways or driving lanes in comparison to a traditional parking structure. On each floor, the parking space is represented as a 2-D grid based rectangular layout. Parking spaces (aka spots or cells) are allocated for a variety of uses on a given floor: such an allocation scheme is illustrated for the ground floor in Figure 1. Upper floors will have only the storage cells, blank cells, elevator space and load/unload bays.

- **Storage Cell:** these are parking spots for vehicles.
- **Blank Cell:** on a given floor, a number of parking cells must always be left empty to facilitate movement of vehicles in transit across a floor for either storage or retrieval.
- **Elevators:** each elevator occupies space for two adjacent spots.
- **Load/Unload Bay:** each elevator has a particular loading/unloading area next to it and each loading area occupies one parking spot.
- **Delivery Bay:** there is a dedicated spot in the middle of any one of the outermost columns or rows of the ground (or entry) floor of the parking structure in order to accept customers' vehicles for parking. Customers who want to park their cars leave their vehicle in the "Delivery Bay".
- **Pickup Bay:** the spots along the outermost row or columns of the ground floor layout are reserved as "Pickup Bay". A vehicle that is being retrieved from its parking location to be delivered to its driver arrives at one of these bays for pickup.

Elevator cells are distributed in the center portion of the layout. "Delivery Bay" and "Pickup Bay" cells are situated on the ground (or entry) floor only. These cells are usually located on any one of the outermost columns or rows, which are considered entry/exit points for the parking structure. The number of elevators is derived from the bound developed in [2] by modeling customer arriving as waiting line model [6] and depends on the number of floors, average customer arriving rate as dictated by the demand, the speed of elevators, and the speed of robotic carts that transport the vehicles.

Each rectangular parking cell may have tracks or roller beds mounted on it to facilitate entry/exit from any of the four sides. Robotic carts with or without vehicles loaded on them can move along the tracks or the vehicles can be moved on the roller beds from one cart to another in neighboring cells. "Customer(s)" make both "Storage Request(s)" and "Retrieval

Request(s)” in order to store or park their vehicles and later collect or pick up their vehicles, respectively. The request to store or park a vehicle inside the parking lot is called “Storage Request” and the request to collect or pick up a previously stored vehicle from the parking lot is called “Retrieval Request”. The autonomous process of parking a vehicle inside the parking lot after customer leaves the vehicle at a “Delivery Bay” is called “Storage Process”. This process includes moving of the vehicle from the “Delivery Bay” to a dedicated “Storage Cell”. If dedicated “Storage Cell” is located in a floor other than the entry or the ground floor, the process also includes the use of an elevator in order to move between floors. The autonomous process of retrieving a “Stored Vehicle” from its “Storage Cell” and delivering to the “Pickup Bay” is called “Retrieval Process”. If dedicated “Storage Cell” is located in a floor other than the entry or ground floor, the process also includes the use of an elevator in order to move between floors. During the “Morning Rush Hour”, most of the “Customer(s)” arrive with “Storage Request(s)” and leave their vehicles at “Delivery Bays” of the ground (or entry) floor. Consequently, the “Storage Process” is completed autonomously with the help of a suit of space assignment, path planning [1] and elevator scheduling [2] algorithms. Similarly, during an evening rush hour most of the “Customer(s)” issue “Retrieval Request(s)” and wait at an “Pickup Bay” of the ground (or entry) floor for successful completion of “Retrieval Process”. The “Retrieval Process” is also completed autonomously with the help of a suit of path planning [1] and elevator scheduling [2] algorithms.

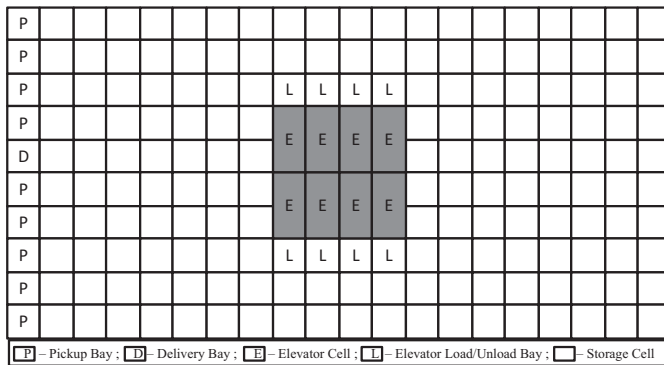


Figure 1. 10×20 ground floor layout for a multi-story automated parking structure

In order to assess the performance of the overall parking management system that employs assignment, path planning and resource scheduling algorithms, we strived to model the simulation environment to resemble as closely as possible a real life scenario. We modeled the arrival of “Customer(s)”, generation of “Storage Request(s)” and “Retrieval Request(s)”, according to a Poisson distribution and employed the queuing theory. The activities regarding “Storage Process(s)” and “Retrieval Process(s)” are modeled such that these processes for each active request can run simultaneously through concurrent threads. The processes regarding the

movement of robotic carts and the elevators can also run simultaneously again by means of concurrent threads. Movement of robotic carts and elevators are modeled in compliance with the laws of physics under certain assumptions.

3 Modular simulation architecture

The overall functionality of simulation is modeled through five major activity modules, which are (a) Automated Parking Lot, (b) Automated Storage Controller, (c) Automated Retrieval Controller, (d) Elevator Controller, and (e) Elevator Scheduler as presented in Figure 2. Separate actors within dedicated threads can execute the activities of each of these five modules simultaneously. A database module named Central Database stores all of the global variables. These global variables are shared among all activity modules. Activity modules on individual threads can communicate with each other and exchange messages through the global variables (Busy-wait synchronization) in the Central Database. We model the communication among concurrent modules (such as between “Elevator Controller” and “Automated Storage/Retrieval Controller”) using busy-wait synchronization technique [7]. The busy-wait synchronization, presented in Figure 3, is implemented using global data structures in the “Central Database” module.

3.1 Automated parking lot module

The simulation starts with the activities of this module through the main thread of the multi-threaded simulation environment. At the very beginning of this module, one parallel thread is created to execute the activities of “Elevator Scheduler” module. After that, another parallel thread is created for each elevator to execute the activities of associated “Elevator Controller” modules. Finally throughout the 24-hour daylong overall simulation time frame, which includes two hours for the “Morning Rush Hour” plus another two hours for the “Evening Rush Hour”, “Storage Request(s)” and “Retrieval Request(s)” are generated continuously at specific time intervals. The number of total requests generated for each time interval is simulated through a Poisson distribution [11], which provides a specific value for the average number of vehicles (or customers) arriving per hour. These requests are then divided into two groups of requests, namely “Storage Request(s)” and “Retrieval Request(s)”, according to the specific rush hour period model. During the morning rush hour, which lasts for two hours starting at 6:30 am and ending at 8:30 am, 95% of requests are for storage and 5% are for retrieval. On the other hand, for the evening rush hour, 95% of requests are for retrieval, and 5% of requests are for storage. Figures 4 and 5 presents the activities during the initial phases of the simulation as performed by the Automated Parking Lot module.

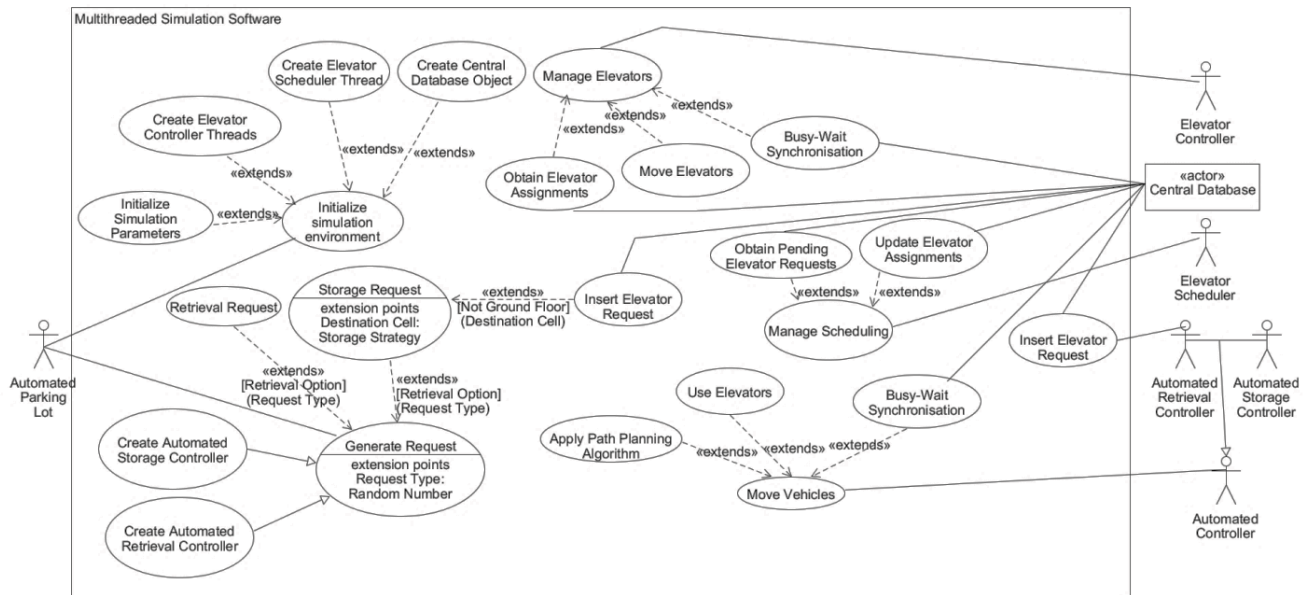


Figure 2. Simulation architecture use-case diagram

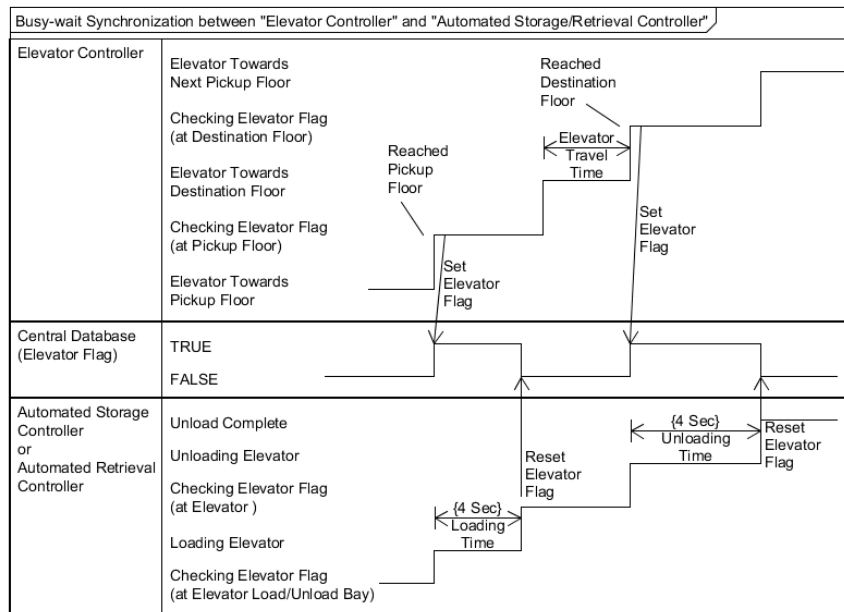


Figure 3. Busy-wait synchronization diagram for Elevator Controller and Automated Storage/Retrieval Controllers

The activities described in this paragraph are repeated sequentially for every "Storage Request" generated in each time interval. First a unique "Storage ID" (a positive integer value) is assigned for the request. This is a unique identifier for the vehicle associated with this request throughout the rest of the simulation period. After that an elevator request containing starting cell location, which is the "Delivery Bay" cell on entry floor, destination cell location, which is assigned through a storage management strategy as described in [1], request arrival time, and a unique "Storage ID" of the request is created. This elevator request is added to a global list variable in the Central Database named *Elevator Request List* if the destination cell for the request is other than the entry (or

ground) floor. Then the "Storage ID" of the request is inserted into a global list variable, *Start Queue*, in the Central Database. *Start Queue* contains the "Storage ID(s)" of the "Storage Request(s)" waiting to enter the parking structure or lot sorted by the ascending order of their arrival time. Finally, a parallel thread is created for each "Storage Request" to execute the activities of "Automated Storage Controller" module which takes care of the rest of the storage process for this "Storage Request". Figure 6 presents the activity diagram representing, in part, the steps described in this paragraph.

The activities described in this paragraph are implemented in order for every "Retrieval Request" generated during each time interval of the simulation. Identifying the

vehicle to be retrieved next is realized through a uniform random distribution: a “Storage ID” from the *Stored Cell List*, a global list variable in the Central Database, is selected randomly (uniform distribution) and associated with the

retrieval request being generated. This list variable stores the “Storage ID(s)” of every vehicle stored successfully in the parking structure.

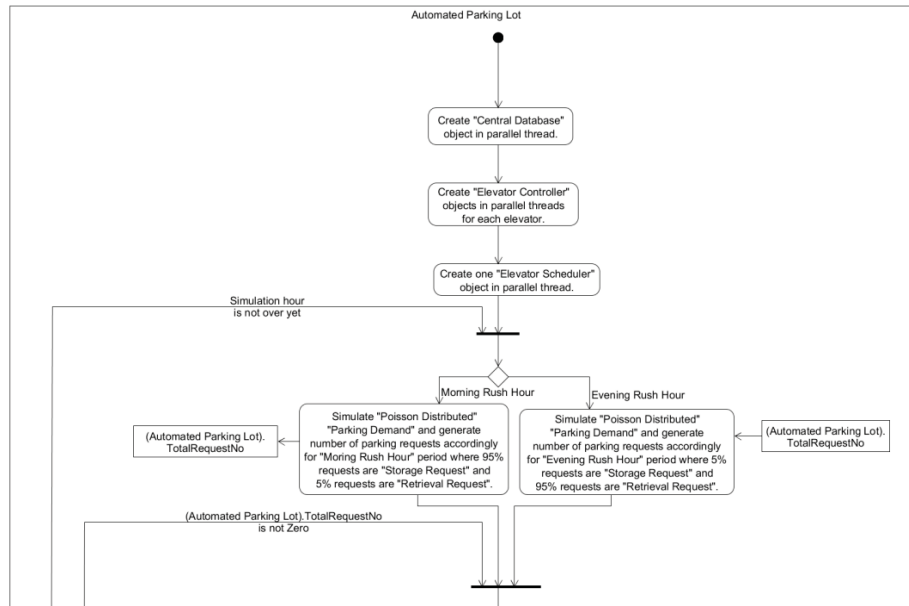


Figure 4. Part 1 of 2 - partial activity diagram for Automated Parking Lot

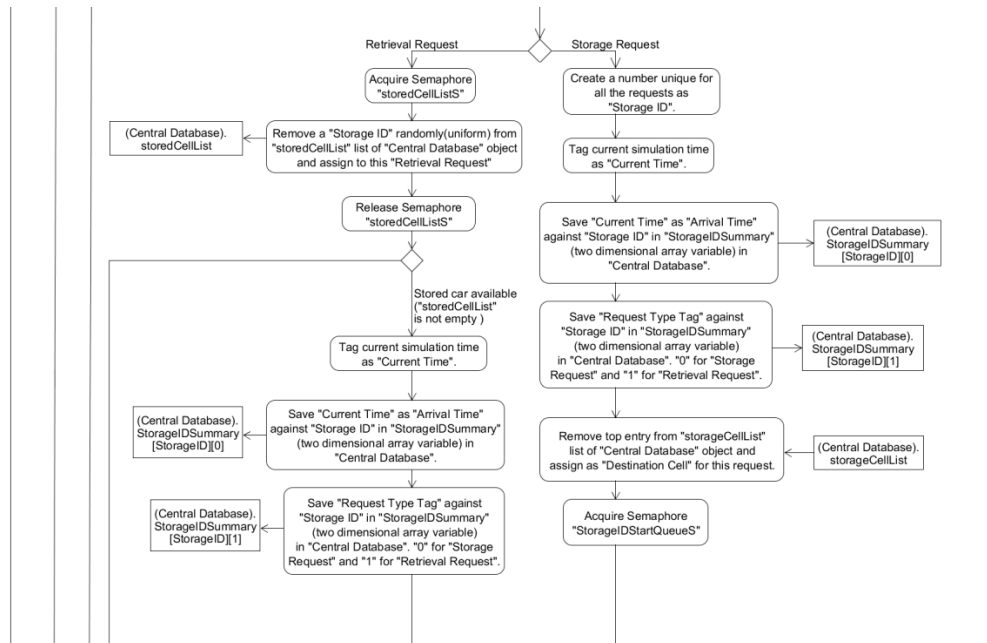


Figure 5. Part 2 of 2 - partial activity diagram for Automated Parking Lot

Finally, a parallel thread is created for each “Retrieval Request” to execute the activities of “Automated Retrieval Controller” module, which takes care of the rest of the retrieval process for this “Retrieval Request”. Figure 7 presents the activity diagram representing, in part, the steps described in this paragraph.

3.2 Automated Storage Controller Module

The functionality required to serve a “Storage Request” including moving the vehicle all the way from the starting location (Delivery Bay) on the entry (or ground) floor to the “Storage Cell” on the destination floor is implemented in the “Automated Storage Controller” module. This module also implements and employs a suite of path planning algorithms, namely the D* Lite and uniform cost search, to move a

vehicle across the floor. For each “Storage Request” a separate thread is created to execute the activities of this module independently and simultaneously with all other threads in the simulation environment.

The activities of this module start with checking if an elevator is assigned for this storage request ID by the “Elevator Scheduler” module. Elevator assignment status is checked continuously until the “Elevator Scheduler” makes an assignment. After an elevator assignment, the status of *Start*

Queue is checked continuously until “Storage ID” of the request comes up as the first or front element of the list. As soon as the “Storage ID” of a request is determined to be the first or front element of the list, it is removed from the list and the vehicle corresponding to the request is moved to the “Delivery Bay” of the parking lot. Then the vehicle is moved from the “Delivery Bay” to the assigned elevator’s “Load/Unload Bay” using a set of path planning algorithms

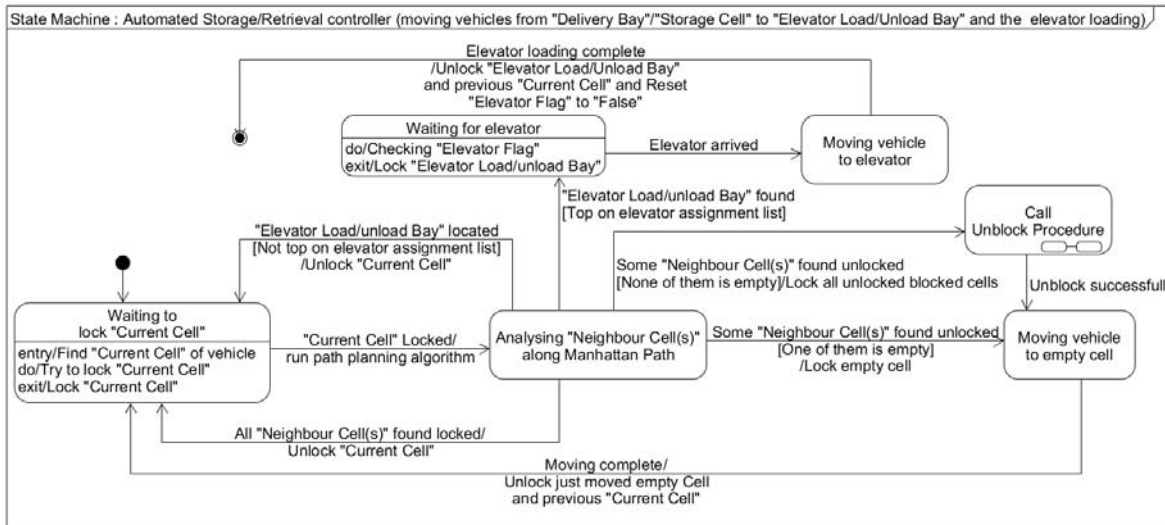


Figure 6. State machine diagram for Automated Storage/Retrieval Controller: vehicle cart moving towards elevators

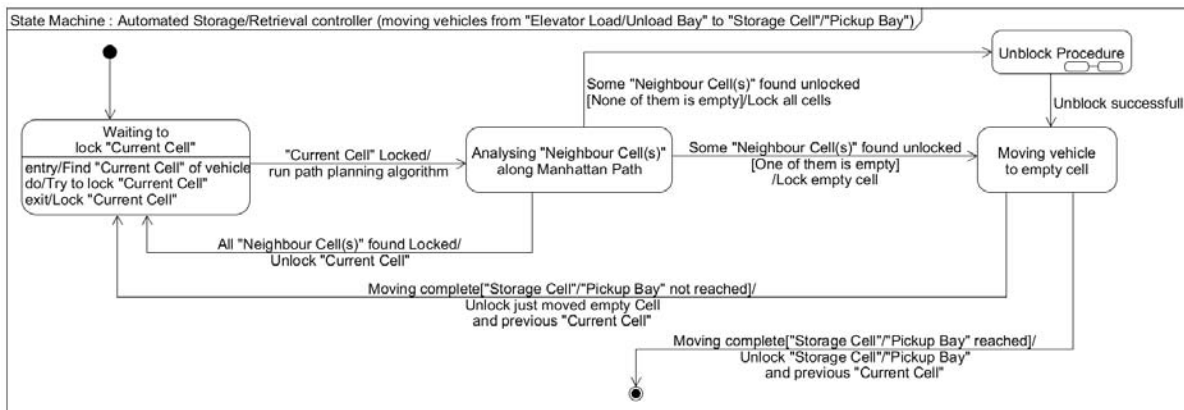


Figure 7. State machine diagram for Automated Storage/Retrieval Controller: vehicle cart moving from elevators

which include the D* Lite and uniform cost search. After reaching the elevator’s “Load/Unload Bay”, the elevator flag status of the assigned elevator is continuously checked until the flag status is found to be *true* (“Elevator Controller” change the elevator flag to *true* when it successfully reaches the pickup floor and ready to pick up). This means assigned elevator is ready to pick up and it is safe to load the vehicle into the assigned elevator. After successful loading operation, the flag status is changed to *false* by “Automated Storage Controller” module. The elevator flag status is again changed to *true* by the “Elevator Controller” when the elevator is done with the transportation of a vehicle from the pickup floor (ground floor) to the destination floor (storage floor). Upon

arrival at the destination floor, the vehicle is unloaded from the elevator to the elevator “Load/Unload Bay” and the elevator flag status is again changed to *false* by “Automated Storage Controller” module. Finally, the vehicle is moved from the elevator’s “Load/Unload Bay” to the destination cell for storage, aka “Storage Cell”, using the set of path planning algorithms.

3.3 Automated retrieval controller module

The activities required to serve a “Retrieval Request” including moving the vehicle from its “Storage Cell” on a given floor (starting floor of retrieval request) to the “Pickup Bay” on the entry (or ground) floor (destination floor of

retrieval request) are executed by the “Automated Retrieval Controller” module. For each “Retrieval Request” a separate thread is created which executes the activities entailed by this module. The initial activity is checking the current location (Storage Cell) of the retrieval vehicle through its “Storage ID”. Next, the “Storage Cell” of the vehicle is locked so that no other threads can lock and move the vehicle for another purpose. After that, an elevator request containing the starting cell location which is where the vehicle is parked, the destination cell location (which is one of the “Pickup Bays” on the entry (or ground) floor), the time of retrieval request and the unique “Storage ID” of the request is created. This elevator request is added into a global list variable *Elevator Request List* in the Central Database if the starting cell for the request is located on a floor other than the entry (or ground) floor.

Elevator assignment status is checked continuously until the “Elevator Scheduler” module makes an assignment. After an elevator assignment is found, the vehicle is moved from the “Storage Cell” to that elevator’s “Load/Unload Bay” using the path planning algorithms. After reaching the elevator “Load/Unload Bay”, the elevator flag status of the assigned elevator is continuously checked until the flag status is found to be *true* (“Elevator Controller” change the elevator flag to *true* when it successfully reaches the pickup floor and ready to pick up). This means the assigned elevator is ready to pick up and the vehicle is safe to load into the elevator. After a successful loading operation, the flag status is changed to *false* by “Automated Retrieval Controller” module and again continuously checking the status of it until the flag status is found *true* again. The elevator flag status is again changed to *true* by the “Elevator Controller” when the elevator is done with the transportation of vehicle from pickup floor (Stored Floor) to destination floor (ground floor). Now, the vehicle is unloaded from the elevator to its “Load/Unload Bay” and the elevator flag status is changed to *false* by “Automated Retrieval Controller” module. Finally, the vehicle is moved from destination floor’s elevator “Load/Unload Bay” to the designated “Pickup Bay” using the path planning algorithms.

3.4 Elevator controller module

This module, presented in Figure 8, executes the activities of the elevators that transport vehicles from one floor to another floor within the multi-storied parking lot. For each elevators a separate thread is launched to control the activities of an associated elevator independently and simultaneously with all other threads in the simulation. Initially at the start of the simulation, all elevators are positioned at the ground floor and ready to transport vehicles to their destinations. This module periodically checks the data structure *Elevator Assignment List* to obtain the assigned elevator requests for an associated elevator. The assigned requests are served sequentially one after another. To serve a request, the elevator is moved to the starting floor for the request, and the flag status of the elevator is set to *true*. Then the Elevator Controller Module continuously checks the flag

status until it becomes *false* which indicates that the associated vehicles was loaded into the elevator. The flag status is set to *false* by the “Automated Storage Controller” or “Automated Retrieval Controller” once the vehicle is inside the elevator. The elevator then transports the vehicle from the pickup floor to the destination floor. After successful transportation of vehicle, the flag status of the elevator is set to *true* by the Elevator Controller Module which then continually monitors the flag status until it becomes *false*. The flag status is set to *false* by the “Automated Storage Controller” or “Automated Retrieval Controller” indicating that the associated vehicle is unloaded successfully from the elevator. This completes the task associated with the assigned request and the elevator request is removed from the *Elevator Assignment List*.

3.5 Elevator scheduler module

This module implements the functionality for the Hybrid Nested Partitions Genetic Algorithm (HNPGA) scheduling algorithm, which is encapsulated within a separate concurrent thread. The main role of this module is assigning an available elevator for all the elevator requests according to the HNPGA scheduling algorithm [2]. Other activities of this module contain periodical scanning of the Elevator Request List to obtain the pending elevator requests by the active “Storage Request(s)” and “Retrieval Request(s)”. On every scan, the pending elevator requests are fulfilled or scheduled and inserted into the Elevator Assignment List with proper elevator assignment information. Next, all assigned (or scheduled) elevator requests are removed from the Elevator Request List.

3.6 Models for processes and physics

We modeled the rush (or busy) hour period based on the parking demand patterns of highly commercialized urban metropolitan cities. There are two different rush hour time periods: “Morning Rush Hour” is from 6:30 AM to 8:30 AM in the morning and “Evening Rush Hour” is in the late afternoon from 4 PM to 6 PM. During “Morning Rush Hour” period, most of the vehicles are being stored and only a small number of them are being retrieved. The parking demand generator will distribute the total number of requests generated for each time interval (using Poisson distribution [11]) with the following probabilities: 95% are for storage, and 5% are for retrieval. This period is terminated once 95% of the parking lot capacity, excluding the blank cells, is occupied. During the “Evening Rush Hour” period, the demand profile is just the opposite when compared to the “Morning Rush Hour” period in that 95% of all the requests are for retrieval while 5% of them are for storage. During this period, the number of parked vehicles on the floor will steadily decrease. Once the capacity utilization reduces to 5% of the total, again excluding the blank cells, this period is considered as over. The demand for parking outside these two rush hour time periods reduced to 50% of rush (or busy) hour

period demand whereas half of them is storage and other half is retrieval requests.

Mechanical roller beds are mounted with carts on each cell across the entire floor. Vehicles to be parked are positioned on mechanical roller beds, which are moved from one cell to another by the sliding mechanisms placed onto the surface of the parking area and built into the parking cell design. Each parking cell has guide-and-travel rails on which the mechanical roller bed mounted vehicle slides along one of the four directions, namely forward (north), backward (south),

leftward (west) and rightward (east). We modeled the movement speed of vehicles from one cell to one of the neighbor cell (among available four cells) is about 1 m/sec [10].

We assume that elevators will reach maximum velocity of $V_{E,max}$ starting from zero initial velocity with constant acceleration a_E , after they travel a distance of $V_{E,max}^2/2a_E$. Similarly, an elevator needs to travel the same

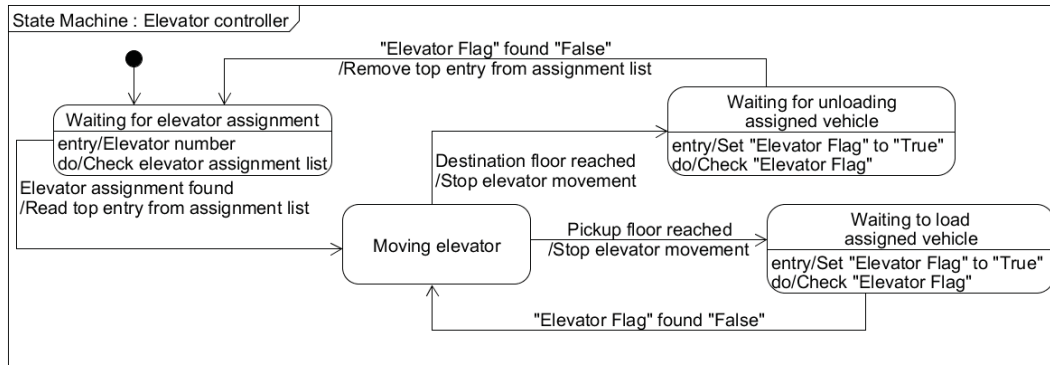


Figure 8. State machine diagram for Elevator Controller

distance to make a complete stop starting with the maximum velocity and down to zero velocity with constant deceleration of $-a_E$. There are two scenarios for the elevator travel, depending on the travel distance between the starting and the destination floors. If the distance between the starting and the destination floors is greater than $V_{E,max}^2/2a_E$, an elevator travels with constant acceleration until reaching its maximum velocity permitted by its design. After that, it travels with constant velocity and starts decreasing its velocity at the point where the distance from the destination is $V_{E,max}^2/2a_E$. Hence, there are three states of motion and they are speeding up, traveling at constant velocity and slowing down. On the other hand, if the distance between the starting and the destination floors is less than $V_{E,max}^2/2a_E$, an elevator goes halfway with constant acceleration and after that (before reaching its maximum speed) slows down with constant deceleration to a complete stop at the destination floor. There are two travel modes as speedup and slowdown. In our modeling, we used industry standard value for elevator model specifications: $V_{E,max}=1400$ ft/min, $a_E=5$ ft/sec², lift capacity = 3500 lb [8], and height between floors = 8.8 ft [9].

4 Conclusions

This paper presented design and development of a highly complex simulation framework for a fully automated robotic multi story parking structure. We used unified modeling language (UML) to model a real-life and real time simulation scenario and successfully implemented the simulation software using the multi-threaded Java programming language. Unified

modeling language proved to be a very effective tool to model and implement complex real time simulation software successfully.

5 References

- [1] Serpen, Gursel, and Chao Dou. "Automated robotic parking systems: real-time, concurrent and multi-robot path planning in dynamic environments." *Applied Intell* 42.2 (2015): 231-251.
- [2] Debnath, Jayanta K., and Gursel Serpen. "Real-Time Optimal Scheduling of a Group of Elevators in a Multi-Story Robotic Fully-Automated Parking Structure." *Procedia Computer Science* 61 (2015): 507-514.
- [3] Sven Koenig, Maxim Likhachev, "Improved Fast Replanning for Robot Navigation in Unknown Terrain", in *Proc. IEEE Int. Conf. Robotics and Automation*, vol. 1, 2002, pp 968 -975.
- [4] Sun J, Zhao Q, Luh PB. Optimization of Group Elevator Scheduling With Advance Information. *IEEE Transactions on Automation Science and Engineering* 2010; 7(2):352-363.
- [5] Parking lots at Beijing Airport. <http://beijing-pek.airports-guides.com/pek-airport-parking.html>. Accessed 17 July 2014
- [6] Kendall DG. *Stochastic Processes Occurring in the Theory of Queues and their Analysis by the Method of the Imbedded Markov Chain*. The Annals of Math. Statistics 1953; 24(3): 338.
- [7] Wikipedia®, "Busy waiting", Internet: http://en.wikipedia.org/wiki/Busy_waiting. Accessed 20 March 2016
- [8] Boulter BT. *Elevator Modeling and DC Drive Speed Controller Design*. Applied Industrial Control Solutions ApICS LLC; 2010.
- [9] What is the height of the levels in the Robotic Parking System? http://www.roboticparking.com/robotic_parking_faq.htm. Accessed 20 March 2016
- [10] Belt-driven actuators. <http://www.macrondynamics.com/belt-actuators>. Accessed 20 March 2016
- [11] Hall RW. *Queuing methods for services and manufacturing*. Randolph Hall; 2013.p. 98-104

Design of a Real-time Simulator Capable of Hardware-in-the-loop Simulation for an Automated Collision Prevention (ACoP) System for an Autonomous Electrical Vehicle

Idrees Alzahid, Fnu Qinggele, and Yong-Kyu Jung

{Alzahid001, qinggele001, and jung002}@gannon.edu

Electrical and Computer Engineering, Gannon University, PA, USA

Abstract

A real-time simulator is an essential means for swiftly designing and verifying embedded time-sensitive and/or real-time applications especially in automotive and transportation. A cost-effective real-time simulator platform for research in automotive electronics has been developed and performed both hardware-/software-in-the-loop (HIL/SIL) simulations of an automated collision prevention (ACoP) system. The multi-core real-time simulator platform was developed for intuitively and swiftly managing different complexity levels and design scales while satisfying real-time constraints with sufficient accuracy of the parallel HIL simulation of models and/or hardware components. The real-time simulator platform was evaluated with the ACoP system integrated to the C/VHDL models of the electric vehicle. We evaluated the HIL simulation with 50 μ s real-time clock resolutions. The HIL simulation achieved 2.5x faster acceleration and deceleration of engine (i.e., motor) than the SIL simulation while maintaining 0.3% of the HIL simulation difference of the speed overshoot and undershoot compared with an ideal Simulink simulation.

I. Introduction

Contemporary transportation systems, such as automobiles and locomotives, are necessary to employ embedded electronics and computing systems for satisfying the rapidly increasing complexity and accuracy of demanding requirements. In addition, research and engineering society has been challenged to convey such swiftly evolving transportation systems under the tight time-to-market pressure. Real-time simulation-based prototyping is one of the proven solutions for embedded system developers in automotive industry. Therefore, a real-time simulator with hardware-in-the-loop (HIL) simulation [1, 2, 3] capability must address the seamless integration of various complex subsystems and the accurate real-time simulation capability with existing and developing hardware and software modules while continuously supporting developers with an intuitive but precise design refinement and effective evaluation.

Various sensors have been employed for electronics and computing parts of automotive and transportation [4]. Cameras are popular for visual identification of the

objects surrounding a vehicle. The GPS systems installed in a vehicle are aimed for navigating and positioning purposes. In particular, an automobile is one of the high potential candidates for applying Internet-of-Things (IoTs), which gather various forms of information from numerous types of sensors via wireless communication and process the information on embedded or application specific processing engines [5].

Efficient and perceptive HIL simulations are generally require swift integration, flexible extension, input/output configuration, precise execution, and systematic verification. Since the HIL simulation permits models of a part of the system to be simulated in real time with the actual hardware of the remainder of the system, developers can promptly evaluate hardware and software subsystems of the electric vehicles (EVs), including a control strategy, various I/O interfaces, different signaling, and signal conditioning. Application-specific real-time simulators, therefore, were developed for different applications, including electric control units in EVs [6], fuel cells in hybrid EVs [7], and electric and hydraulic systems in avionics [8].

As the demand for real-time simulators increases in industry, significant growth in the number of RT-Sims has been evident during the last decade in academia [9]. Unlike industry, academic version of real-time simulators [10, 11] are expected to embrace specific features including meaningful and relevant experience without being limited by laboratory equipment, user-friendly interfaces with increasing sophistication, the flexibility for continuous expansion, and preferred cost-effectiveness. We have developed a cost-effective academic real-time simulator (RT-Sim) with the HIL capability in order to successfully utilize the RT-Sim in academia, especially for different disciplines including Embedded Systems, and Communications [12]. In particular, the developed RT-Sim with the HIL simulation is beneficial to successfully perform automotive electronics research including an automated collision prevention (ACoP) system, which comprises of the hardware and software subsystems of a wireless sensor system (WiSS). Section 2 describes the architecture and operation of the ACoP. Section 3 expresses evaluation of the ACoP integrated to an EV model via a flexible wireless interface module for the rapid HIL simulations. The evaluation results and

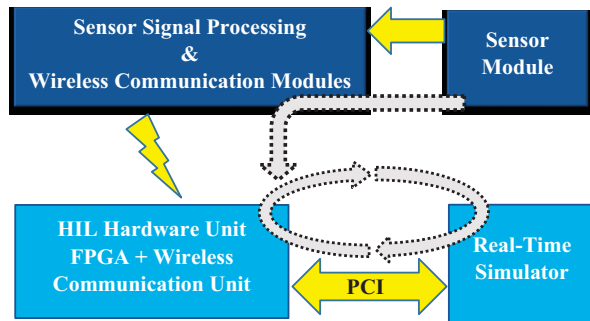


Figure 1. A Block Diagram of the Wireless Sensor System (WiSS) integrated for a Hardware-in-the-loop (HIL) Real-time Simulation

analysis of the WiSS on the EV are also described in Section 3. Section 4 depicts the conclusions.

II. Architecture and Operation of a HIL Real-time Simulation for Wireless Sensor System (WiSS)

Figure 1 illustrates a block diagram of the WiSS integrated for the HIL real-time simulation architecture and operation. The WiSS consists of a sensor module and a sensor signal processing and wireless communication module. The presented WiSS was implemented with ultrasonic sensors [13] for detecting and measuring objects, an Arduino board [14] for processing sensor signals, and transmitting information to and receiving it from the flexible wireless interface module consisting of a wireless integrated field-programmable gate array (FPGA) [15] for the accurate and swift HIL simulations. In order to establish a cost-effective wireless communication network, a pair of Xbee modules [16] is employed between the WiSS and the flexible wireless interface module.

As seen in Figure 1, the flexible wireless interface module is integrated to RT-Sim via PCI [17] with wires and to the sensor signal processing and communication module without wires. The HIL simulation with WiSS is an extended form of the HIL real-time simulation. A traditional HIL simulation can be found between the EV models running in the RT-Sim and a time-critical subsystem implemented in FPGA running on the interface module. The interface module encompasses a cost-effective wireless capability in order to provide a viable means for extending various sensor systems without further physical hardware modification of the current FPGA-based interface module for the presented HIL simulation. Packets of information generated by the sensor signal processing module are delivered through the wireless channels established between the sensor signal processing module and the interface module.

III. Design of An Automated Collision Prevention (ACoP) system with WiSS

The presented WiSS is designed for researching an ACoP system of an EV. The ACoP is an emergency driving assistant system that can provide a safety means for drivers and passengers in the EV under uncontrollable and/or sudden disrupted situations on the road. The ACoP system can automatically detect distance of any objects surrounding the vehicle. In addition, the ACoP system is capable of generating a warning alarm for the driver as well as possibly taking control over the EV to avoid collision between the EV and any other objects including other vehicles, unmovable objects, such as trees, street lights, and so on, as well as pedestrians. Therefore, the ACoP system can not only slow down and stop the EV, but also change the direction of the EV in order to prevent any collisions.

The ACoP system can consist of a hardware and software subsystems including the proposed WiSS. The WiSS is to detect objects and measure the distance to the objects. A hardware subsystem of the WiSS consists of (1) an ultra-sonic sensor and driving system, (2) a pair of wireless communication system (i.e., Xbee 2.4 GHz), (3) an interface control logic written in VHDL programmed to the FPGA module. A software subsystem includes Arduino script for operating the ultra-sonic sensor driving system. After the hardware and software partitioning and implementation of the subsystems, the WiSS was integrated and verified for exercising a top-down verification method and for rapid prototyping of

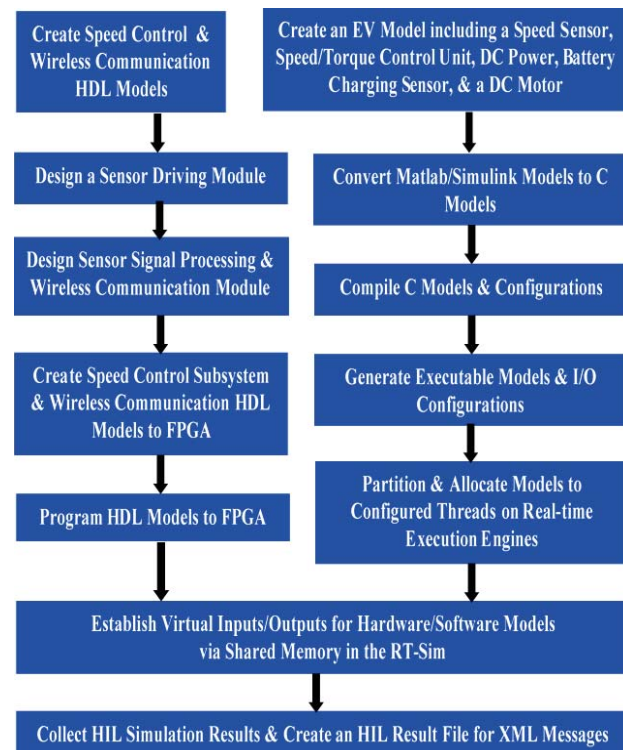


Figure 2. The WiSS Design and Operational Procedures for the HIL Real-time Simulations

the WiSS with the HIL real-time simulations. In order to expedite the HIL simulations, a series of the verification scenarios was developed and applied for developing a comprehensive ACoP algorithm that dynamically reflects various information on road size, road condition, driving direction, driving speed limit, and other information including weather and temperature via several expanded sensor and processing systems as a part of an IoT in future enhancement.

Figure 2 illustrates the WiSS design and operational procedure for the HIL simulation performed. The complete Simulink closed-loop EV model was designed. The EV consists of subsystems for sensing speed of EV and charging level of the battery, a control unit for controlling speed and torque of the DC motor, a battery for supplying DC power, and a DC motor for driving. The EV specifications includes a 25 horsepower, four quadrant operation wound DC motor, which is designed to execute in discrete time with a sampling rate of 1 micro-second. The speed control unit determines the speed of the armature. Inputs of the speed control unit are the desired speed in RPM, and the armature speed in RPM. Outputs of the speed control unit are the speed changes and armature currents. The speed control unit controls the armature current and prevents the current from surpassing the rated armature current.

The speed control unit receives the armature current, armature speed, speed change, and change in armature current, and generates the PWM pulses to set the armature voltage to the desired voltage in order to achieve the desired armature speed. The speed control unit also generates the control values for determining the PWM pulses. The DC Motor is powered by a 30 volt battery with a linear load torque. The DC Motor comprises a DC-DC converter connected to the PWM pulses to provide the desired armature voltage. I/Os of the motor are the torque load, PWM pulses, and battery voltage as the inputs and the armature current, armature speed, armature voltage, and the field voltage as the outputs.

The sensor module employs a ping ultra-sonic distance sensor to detect and measure the distance between the EV and any objects. The accuracy of the sensor employed is to measure the distance within 3 meters, which is about 3.3 yards. In addition, the sensor module is sufficiently durable for the outdoor usage. Since the sensor needs to interface to a microcontroller for further processing of the sensor outputs, an ATmega328 installed on an Arduino Uno, which is one the most popular for motor controlling applications, was selected. Analog and digital I/Os of the Arduino development board are used to integrate a cost-effective wireless device, such as an XBee S2 module. Therefore, a sensor signal captured and transmitted from the sensor module is processed by the microcontroller, which is also integrated to the wireless device.

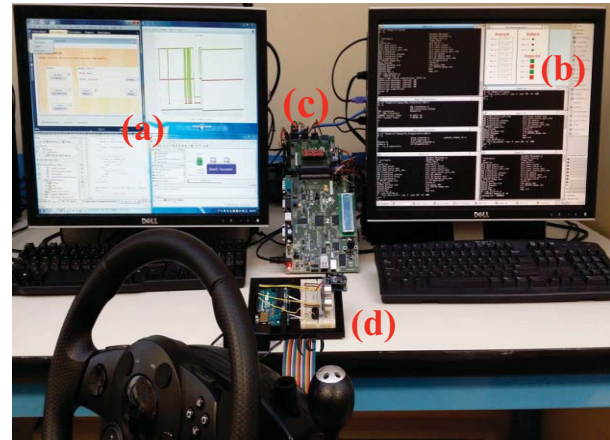


Figure 3. A WiSS Evaluation Setup for the HIL Real-time Simulations; (a) a console for user interface, (b) the RT-Sim developed, (c) the FPGA-based wireless interface module for HIL simulations, and (d) a prototype of the WiSS

The XBee device with its adapter offers an intuitive means of integration to the microcontroller in the Arduino Uno. The XBee device operating with 2.4 GHz clock establishes a wireless communication channel capable of interconnecting within 120 meters (i.e., 400 feet). A serial communication channel is established by the XBee devices between the WiSS and the FPGA-based interface module, which is attached to the RT-Sim for performing HIL simulations. A wireless communication HDL model was developed and programmed to the FPGA (i.e., Xilinx Spartan 3E). The inputs and outputs of the XBee devices are interconnected to I/Os of the FPGA.

IV. Evaluation of the ACoP System via the HIL Real-time Simulations

Figure 3 illustrates an overview of the HIL simulation setup for the WiSS. The RT-Sim we developed consists of three primary modules—(a) a console for user interface, (b) the RT-Sim for HIL simulations, and (c) an FPGA-based hardware interface module with wire/wireless connections for extended HIL simulations for WiSS. In particular, the FPGA-based hardware interface module with wireless capability provides additional flexibility to expand parallel and distributed HIL simulations without sacrificing a number of threads running in the RT-Sim. The WiSS consisting of the sensor and wireless signal processing modules is shown in Figure 3 (d).

A series of the WiSS evaluations has been developed based on the EV running on a highway. In order to develop the practical evaluation scenarios, we need to determine a few thresholds for identifying normal, warning, and extreme situations. According to

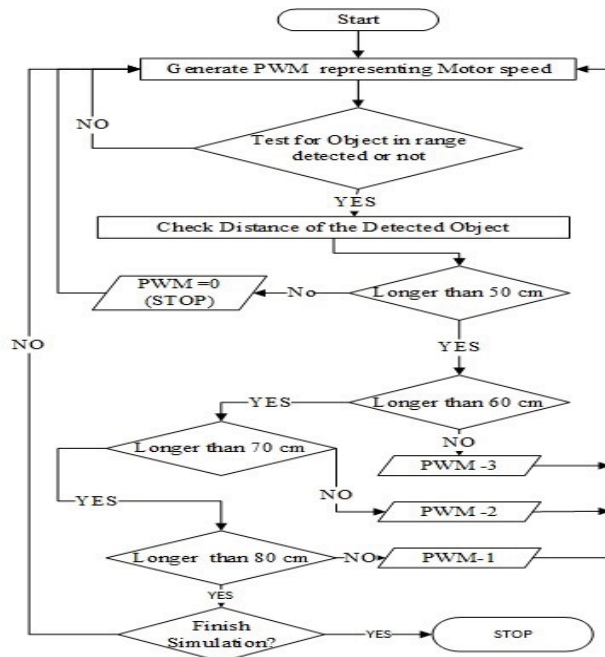


Figure 4. An Electric Vehicle Speed Controlling Flows based on a series of the evaluation scenarios for the WiSS

the federal highway administration, the width of a freeway road is 3.6 meters [18]. An average width (i.e., approximately 1.6 meters) of vehicles is used for our evaluations. The first warning threshold was determined by the distance between a vehicle and the EV equipped with the WiSS approaching the side or rear-end of the vehicle or vice versa. The distance identified for the warning is 80 cm. The other distances, such as the second warning and the extreme situation, are set in every 10 cm range. We assumed the EV can be driven faster than another vehicle can and the EV speed is 70 mph, which means the vehicle runs approximately 31.29 meters per second.

Figure 4 illustrates the different zones identified for the evaluations. In order to control the speed of the motor engine of the EV, four different acceleration and deceleration PWM pulse sequences are used as seen in Figure 4. Since the speed control depends on the initial speed, the primary PWM pulse sequences are accordingly modified. In addition, the PWM pulses are adjusted by receiving the feedback information transferred from the speed control unit running on the RT-Sim.

The HIL simulation period is 50 μ s. The EV's maximum acceleration and deceleration cycles are measured as 3600 HIL simulation cycles. Thus, an acceleration of the motor engine can be completed within 180 ms. For instance, the EV driven in 70 mph moves 31.29 meter per second. The EV can move 0.156 centimeter per HIL simulation cycle, which is 50 μ s. The

sensor module was tested alone for evaluating accuracy of the distance measured. We obtain less than 2.89% error within 1 centimeter. For instance, about 3 distances measured over the same 100 distances were different by more than 1 centimeter. Therefore, our HIL simulation is sufficient in accuracy of the simulation.

In addition, the processed sensor data take a maximum of 5,355 μ s for 100 cm and a minimum of 3,630 μ s for 50 cm. We monitored packets delivered to the FPGA via XBee devices is an average of 100 bytes per packet. The XBee's data rate is 250 Kbps. Therefore, the wireless communication latency is 3,200 μ s. We calibrated the maximum and minimum latencies of the sensor signals as 8,555 μ s and 6,830 μ s, respectively. Since there is another XBee wireless communication latency on the FPGA-based interface module, a range of the latencies of the WiSS is between 11,755 μ s and 10,030 μ s. According to the signal latencies measured, the WiSS HIL simulations run between 200 and 235 HIL simulation cycles per sensor signal received from the sensor. Consequently, more than 200 requests from similar sensors can be processed concurrently by the presented RT-Sim with HIL simulations.

Figure 5 illustrates the WiSS HIL simulation results captured by the RT-Sim. An object was moved from the WiSS installed on the EV to measure the various distances. The measured distances reflect the sensor signals transmitted by the sensor module. The reactions of the EV are shown as three curves in Figure 5. The green curve represents actual speed (RPM) of the motor engine, the red curve shows tracking speed of the motor, and the blue curve illustrates final speed of the motor given at zero. The X-axis is the number of RPM samples

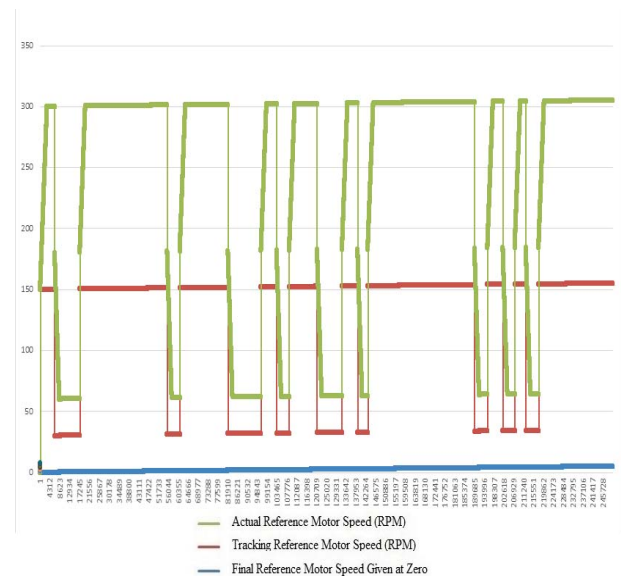


Figure 5. The WiSS HIL Simulation Results for Various Evaluation Scenarios; 250K samples are measured during the 5 second HIL simulation

collected during the 5-second HIL simulation. Total 250K samples were acquired. The Y-axis is the speed of the motor engine. The speeds (RPMs) were varied from 60 to 300 RPMs. The initial RPMs started at zero.

V. HIL real-time Simulations of the ACoP System with an Autonomous Electric Vehicle (AEV) Prototype

The ACoP system is implemented as an autonomous driving platform that is capable of performing wireless communication and data processing via a data processing and interface module for the HIL real-time simulations. The platform includes a four-wheel autonomous driving miniature vehicle assembled with a distance detector, a detection-angle controller, a motor driving module, and a wireless transmitter. Each of the sub-systems cooperates with each other under a microcontroller. The data processing and interface module consists of a wireless receiver and a processing terminal. The processing terminal handles graphical data and signal processing. All of the sub-systems in the platform are integrated to the real-time simulator.

The proposed platform operates autonomously while interacting with the real-time simulator via the wireless communication channel established between the platform and the real-time simulator via an FPGA-based HIL interface module. While the self-driving platform is moving on the testing ground, the distance detector scans the surrounding road-blocks and transfers the real-time driving circumstance to the DPI module. Then, it converts the data being collected to legible information and displays the information on the console monitor connected to the real-time simulator. At the same time, the light indicator on the FPGA-based HIL interface module blinks in accordance with specified rules to indicate successful data communication. The proposed concept of the platform-based HIL real-time simulation is to provide a viable and flexible means for swiftly designing and verifying important algorithms and components of the future automobiles in academic laboratories and design facilities.

The real-time simulation platform shown in Figure 1 has been evaluated for the HIL simulations with the ACoP system wirelessly connected to the FPGA module integrated to the two 14-bit output ADCs with 1.5 MHz sampling rate and 8 analog outputs (i.e., +/- 12 Volts), 3 digital outputs, 4 digital inputs (5 Volts), and a 12-bit resolution DAC with 6 analog outputs in the RTS via the voltage conversion units for various signaling standards including LVTTTL, LVCMOS2/18, and TTL in the HIL module.

Three key models of an electrical vehicle—(1) DC motor with a battery, (2) current controller, and (3) speed controller—are designed as Simulink models, which are converted by the s-functions configured in RTS. In particular, the current controller model was modified for

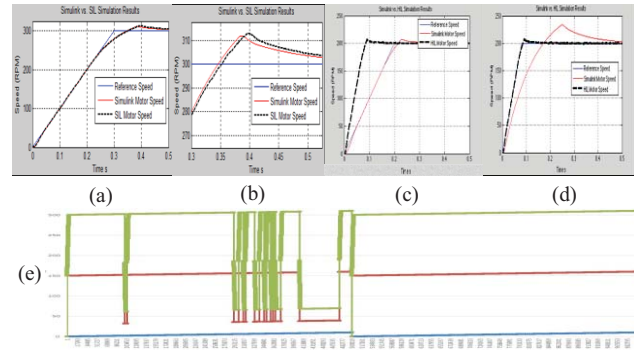


Figure 6. Engine (Motor) Acceleration HIL Simulation Results of an AEV Speed Controller: SIL vs Simulink (a) normal acceleration; (b) increased acceleration; & HIL vs Simulink (c) normal acceleration; (d) increased acceleration; and (e) ACoP HIL simulation.

the HIL simulation before distributing the executable codes of the associated models to three threads. The 0.625 MHz PWM signals are generated by a PWM generator implemented in the FPGA module.

The ACoP generates information on the objects detected and distances to the objects, processes the information with an Arduino board, and transmits the information via a 2.4 GHz operating Xbee adapter. The information is then received by another Xbee adapter installed to the FPGA module and passed to the ACoP algorithm implemented in HDL as an HW model for further processing with other HW models including the PWM generator.

As seen in Figure 6, the results of the Simulink, SIL real-time simulation, and HIL real-time simulation confirm that the RTS accurately simulated both of the HW/SW models. The simulation results are the reference speed and engine speed. The results, with a reference speed of 300 RPM, illustrate that the motor is able to achieve the desired speed within 0.3 seconds with a 10 RPM overshoot. These results are used as the baseline for the SIL and HIL evaluation of the real-time simulation. The overshoots measured in Simulink versus SIL and HIL simulations are 311 versus 312 and 205.8 versus 206.4. The percentage differences are 0.32% with SIL and 0.29% with HIL simulation. We discovered that the beginning of the SIL/HIL simulation between the virtual I/O and models are not synchronized in that the real-time simulation results are different within 0.1% of the Simulink simulation results. The results, however, still prove the accuracy of the real-time simulation. On the other hand, the acceleration times measured via the HIL and SIL simulation are more than 2.5 times different. Thus, the HIL real-time simulation proves to meet real-time constraints. Figure 4 (e) illustrates the engine speed control results of the ACoP HIL simulation. According to the distances detected from the objects, the engine (motor) speed decreased when the distance was within a warning zone and stopped when

the distance crossed into the threshold of the collision but the driver did not reduce nor stop the vehicle.

The results of determining the access time of the virtual I/Os via the shared memory, PCI card, and PCI ADC in the RTS module were measured. An average of 0.1486 μ s was for access data from/to the shared memory. The PCI access time measured was an average of 1.7889 μ s, due to the bus used to access the PCI. The accessing ADC was identified as a critical path with an average of 13.2093 μ s delay, which limits the faster model execution for the HIL simulation. The resolution of the real-time clock on the RTS is set to 10 μ s. Since the most reliable time to consistently operate at the same timing interval was identified as 50 μ s, the HIL execution on the RTS operates at a 50 μ s for the real-time HIL simulation evaluation.

VI. Conclusions

A real-time simulation platform for automotive electronics is introduced for rapid and intuitive management, accurate simulation, and cost-efficient real-time environment for research and education. The real-time simulation platform is capable of executing both SIL and HIL simulations with sufficient accuracy in terms of real-time constraints and model operations. In addition, the real-time simulation provides a means to integrate existing design tools, such as Matlab/Simulink and FPGA-based platform to user developed HW/SW model configuration expansion, and simulation result management. A WiSS is presented as an extended hardware/software system for the hardware-in-the loop simulations of an autonomous electrical vehicle with the real-time simulator developed. In order to enhance the expandability of the HIL simulations, the FPGA-based interface module was upgraded with wireless connectivity. According to the analysis of the WiSS HIL simulations, up to 200 sensor systems can be simultaneously supported by the presented HIL simulation environment. The WiSS was successfully developed as a preliminary version of the ACoP system. A version of the ACoP system is successfully evaluated by utilizing external HW module wirelessly interfaced to the real-time simulation platform. In particular, the proposed real-time simulation efficiently schedules, distributes, interfaces, and simulates underlying SW and HW models and prototypes. Furthermore, the real-time simulation offers various wireless connections to users' hardware prototypes via a standardized wireless module. The presented real-time simulation platform evaluated the accurate (<0.3% error) HIL simulation and 2.5x faster operations with 50 μ s real-time clock resolutions. The HIL simulation achieved 2.5x faster acceleration

and deceleration of engine (i.e., motor) than the SIL simulation while maintaining the HIL simulation difference of the speed overshoot and undershoot compared with the Simulink simulations.

VII. References

- [1] C. Dufour, S. Abourida, and J. Bélanger, "Hardware-in-the-Loop Simulation of Power Drives with RT-LAB," IEEE PEDS pp. 1646-1651, 2005.
- [2] R. McNeal and M. Belkhaty, "Standard Tools for Hardware-in-the-Loop (HIL) Modeling and Simulation," IEEE ESTS, pp. 130-137, 2007.
- [3] O. Mohammed, N. Abed, and S. Ganu, "Real-Time Simulations of Electrical Machine Drives with Hardware-in-the-Loop," IEEE PESGM, pp. 1-6, 2007.
- [4] P. Ranky, "Advanced Digital Automobile Sensor Applications," Sensor Review, Vol. 22, No. 3, pp. 213-217, 2002.
- [5] D. Georgoulas and E. In-Motes, "A Real Time Application for Automobiles in Wireless Sensor Networks," MSR, Vol.3, No. 5, pp.158-166, 2011.
- [6] L. Cheng and Z. Lipeng, "Hardware-in-the-Loop Simulation and Its Application in Electric Vehicle Development," IEEE VPPC, 2008.
- [7] C. Dufour, J. Bélanger, T. Ishikawa, and K. Uemura, "Advances in Real-Time Simulation of Fuel Cell Hybrid Electric Vehicles," Proceedings of the 21st Electric Vehicle Symposium (EVS-21), April 2-6 2005.
- [8] J. Casteres and T. Ramaherirany, "Aircraft integration real-time simulator Modeling with AADL for architecture tradeoffs," IEEE DATE, 2009.
- [9] P. Menghal and A. Jaya laxmi, "Real Time Simulation: A Novel Approach in Engineering Education," IEEE ECT, pp. 215-219, 2011.
- [10] C. Dufour, C. Andrade and J. Bélanger, "Real-Time Simulation Technologies in Education: a Link to Modern Engineering Methods and Practices," ETE, 2010.
- [11] S. Abourida, C. Dufour, J. Bélanger, and V. Lapointe, "Real-Time, PC-Based Simulator of Electric Systems and Drives," PST, pp. 1-6, 2003.
- [12] T. Silloway, Y. Jung, D. Mackellar, F. Mak, "Design of a Real-Time Simulator for an Electric Vehicle," MSV, pp. 163-168, 2012.
- [13] R. Dhole, V. Undre, C. Solanki, S. Pawale, "Smart Traffic Signal Using Ultrasonic Sensor," IEEE GCCEE, 2014.
- [14] Jeremy Blum and Scott Fitzgerald, Exploring Arduino: Tools and Techniques for Engineering Wizardry, John Wiley & Sons, 2013.
- [15] L. Junsong, et al, "FPGA Based Wireless Sensor Node with Customizable Event-Driven Architecture," JES, 2013.
- [16] Telecommunications Weekly, "Digi Launches Next Generation XBee and XBee-PRO ZigBee Modules," 2010.
- [17] Chan, Chris, "High-Performance PCI Card: Main/Lifestyle Edition," New Straits Times: 10. 2007.
- [18] http://safety.fhwa.dot.gov/geometric/pubs/mitigations/trategies/chapter3/3_lanewidth.cfm, Website accessed on March 20th, 2015.

EDUCATING DISCRETE SIMULATION BY AGENT-BASED ROLEPLAY

H.P.M. Veeke, J.A. Ottjes, G. Lodewijks

Dep. of Marine and Transport Technology

Faculty of Mechanical, Maritime and Materials Engineering, 3mE

Delft University of Technology

Mekelweg 2, 2628 CD Delft, the Netherlands

E-mail: H.P.M.Veeke@tudelft.nl, J.A.Ottjes@tudelft.nl, G.Lodewijks@tudelft.nl

Abstract

Since the introduction of real process oriented simulation it has been educated in two different ways. For students in informatics and mathematics it is educated in a strict formal way based on things like paradigms and finite state machines. For students that don't need to become professional programmers but do have to understand the principles of simulation it is mostly educated in an informal intuitive way when not learned by off-the-shelf click-and-play packages. Starting from a general programming platform, the major problem in educating simulation is the explanation of simultaneity and synchronization. This paper describes a recently developed method, by which students experience these problems themselves and –as far as the first results can show– master the techniques to solve synchronization problems. The method is based on roleplaying agents.

Keywords: simulation, process interaction, education, agent-based

1 Introduction

Many years of experience in educating simulation led us to the conclusion that the main problem is to let students understand how to describe unambiguously the “behavior of a system”, in a time-based manner. Most discrete simulation literature [e.g. 1,2] starts from a notion of a change in the state of a system defining this notion as an “event”. The ‘event’ however appears to be too abstract to understand completely and it leads to complex implementations of a system’s behavior.

The real difficulty in understanding behavior is to realize that we implicitly take some “events” for granted, while they are essential for the synchronization of processing activities. “Doing nothing” is also a type of activity. This paper explains an agent-based role-play approach that improves the understanding and leads in a natural way to the process-oriented approach for describing the behavior of discrete systems.

2. Events vs. Processes

In [3] the construction of a “Tool for Object oriented Modelling And Simulation” (TOMAS) has been presented. It is implemented as a toolbox in the general programming platform Delphi®, so the complete functionality of Delphi® can be used too. Using Delphi® is not essential, but Delphi® is based on Pascal and this offers many advantages for students that are not supposed to become experienced programmers. Delphi® offers all possibilities and flexibility of a general programming language, so there will be no restrictions other than the creativity of the student or researcher. The way of modelling, closely matches the qualitative modelling as defined in the Delft Systems Approach (DSA) [4,5]. It differs widely from the approaches used in well-known packages. DSA uses as its main modelling element the concept of a “function” that expresses why a particular process is executed and what its contribution is to the environment. By this the modeller takes the necessary distance from what he experiences, in order to make a general model for the situation under investigation. Within this notion of function, a process is described from the company’s viewpoint (as a repetitive series of activities of a department /group/person/machine that handles orders, materials, or even resources). Many packages use the viewpoint of the customer or the flowing element itself (a visitor’s view). The visitor and company views are really different; for example, Zeigler et al. [6] call it the “flow oriented” vs. “real process oriented” approach. The latter approach is characterized by the fact that the sequence in which program statements are executed, differs from the written sequence.

An example is shown in table 1 for two elements A and B (the arrows show the order in which statements are executed). Already in the nineteen seventies a first implementation of the real process approach was constructed by Sierenberg and De Gans [7], called PROSIM. At that time the lectures about simulation with PROSIM didn’t explain how to choose elements and corresponding processes. The lectures appealed to the common sense of the students, which worked very well for some of the students, but left others in confusion.

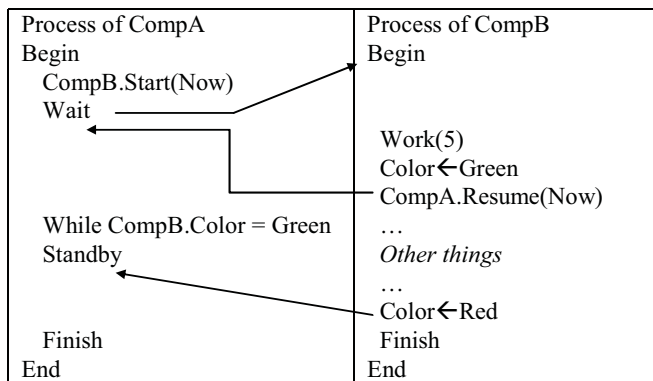


Table 1. Real Process Oriented Approach

The problem appeared to be twofold:

1. What is the selection criterion to choose elements?
2. How to describe and communicate the behaviour of the elements?

The first question has been answered in an earlier contribution [8], where it was found that recognizing the functions that need to be fulfilled, led to the elements fulfilling them. Here we will focus on the second question.

3. Behavior

A first short description will highlight the role of the elements in the model. Throughout this paper the example of an automated container terminal's import processes will be followed. Ships arrive from deep-sea at a berth of a container terminal. A number of containers should be unloaded, transported to a stacking area and stored there.

The (physical) elements in the model are Ships, Quay cranes, AGV's (Automated Guided Vehicles), ASC's (Automated Stacking Cranes), Stack and Containers.



Fig.1. artist impression of Automated Container Terminal

The role of each element is:

Ship : arrives with containers at berth
 Quay crane : unloads containers from ship
 AGV : transports containers from Quay crane to ASC
 ASC : stores containers into stack
 Stack : keeps containers in storage

Actually this is already a complete behavior description of the import processes, but there is no synchronization at all yet. Providing facilities to synchronize the different processes is the core problem of describing the behavior in order to construct a model like the one in the figure below.

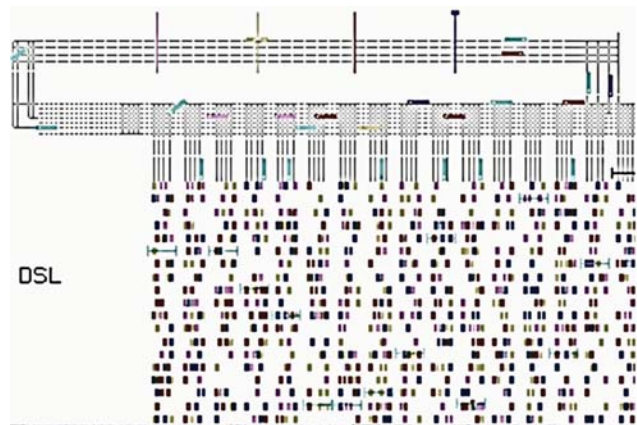


Fig.2. Screenshot of a model of container import processes [9]

We used to progress interactively with the students by expanding step by step the descriptions above. If we restrict the description to the synchronization between Quay cranes, AGV's and ASC's it could look like the table below.

Firstly it should be made clear that all equipment (or resources) repeat their actions during the whole simulation run, so its process description starts with "Repeat".

A Quay crane starts with unloading a container from a ship and needs a first synchronization with AGV activities; an AGV should be simply there to take over the container. Here we use a queue (QcQ) for this purpose, and let the Quay crane wait until there is at least one AGV in the queue. Queues are standard available in simulation packages, and one can use them for many purposes, here it is used for synchronization. Elements that have been placed in a queue should be removed from it too, so the quay crane removes the agv in front from the queue and puts a container on it. After that it signals the Agv by "Resume" to continue its independent part of the

process: driving to an ASC. At the ASC, the processes are synchronized in an analogous way.

Quay crane	Repeat ... <i>Wait</i> While QcQ is empty Remove first AGV from QcQ <i>Put</i> Container on AGV Resume AGV ...
AGV	Repeat Enter QcQ <i>Wait</i> <i>Drive</i> to ASC Enter ASCQ <i>Wait</i> <i>Drive</i> to Quay crane
ASC	Repeat ... <i>Wait</i> While ASCQ is empty Remove first AGV from ASCQ <i>Lift</i> Container from AGV Resume AGV ...

Table 2. Informal process descriptions

The “Process Description Language (PDL)” as presented in the table above, is being used to communicate on the behavior of the model. It is very useful both in teaching environments and practical design projects for the verification of the model. The big advantage of PDL is its simplicity and clarity, without the need of a special syntax and constructions that would be imposed by programming environments, I case we would try to describe the model’s behavior immediately in some programming language or package.

We used to describe situations in this way interactively with the students, but noticed they had difficulties to reproduce it for other situations. The majority managed to define the elements correctly, but not all students were capable of reproducing this way of thinking on behavior in other situations. Many students stranded in an attempt to re-invent the basic provisions already available in the simulation toolbox. We apparently have to prevent that students consider the technical needs of a simulation environment as “modeling”; Instead they should focus on the synchronization needs of a modeling situation with the tools available.

We decided to adopt the agent based approach of programming and to replace each element with its corresponding agent, a straightforward conversion. An agent is the natural owner of a process and differs in nothing with a general simulation element.

4 Synchronization

Synchronization can be achieved in many different ways. First of all one could use a general type of semaphore that turns green or red to show “continue” or “stop”. The disadvantage of this general approach is the loss of readability / understandability of the model.

We prefer to use the already available facilities of any simulation platform: attributes of elements and/or queues.

If the model contains an element “Fence” with a Boolean attribute Closed, one could easily make another element waiting for the Fence until it is open, by specifying “Wait while Fence.Closed = True” in its process description. It makes the use of semaphores very clear and natural in the descriptions. Even more powerful is the use of queues for synchronization. Many situations can be covered by one single queue status; it can be empty, it can be full, it can contain or just not contain one specific element. Depending on this status it is easy to stop or continue a process when one (or more) of these conditions is met.

We used to explain synchronization in this way, it seems trivial however when someone else is telling you how to implement it. Real difficulties arise when students have to construct it themselves.

In order to get the synchronization points clear, each student is assigned an agent of the model and together they should proceed in time as a system, a team of cooperating agents.

Each active time-consuming statement is assumed to take 5 seconds. In our example it concerns the statements Drive, and Put/Lift Container. Passive time-consuming statements should be solved with synchronization; it concerns the Wait statements in this case of which the time duration is unknown beforehand and depends on actions of other agents.

At the start of the role-play, each student is asked what his/her first action will be. The first question for each student should be: “Where am I?” Immediately followed by “What is my state?”. Most of the students start mentioning actions, but forget these questions. They already assume implicitly that everybody knows where they are and in what state. This should be made explicit however, because it determines the starting point of the processes of other agents. It is also very important to decide on the starting state, because the behavior of most agents is repetitive and it should be easy to start the process in any way from this state. We assume an AGV is empty and starts waiting at the Quay crane by entering the QcQ.

When it is inside the QcQ, an AGV can only proceed after it has received a container from the quay crane. The Quay crane is the only one who determines this moment, so the only thing an AGV has to do is waiting; it doesn’t have to stay looking (actively) until a container has been placed, it will get a signal from the Quay crane. The AGV agent waits until the Quay

crane agent tells it may continue. This is different from actively waiting like the Quay crane agent does. This agent waits until an AGV arrives in the QcQ, and after any “event” the agent should check this condition. It is very illustrative for the student to be asked by the lecturer every time to check this condition.

Now other students should start their process. The Quay crane will wait for a ship; after arrival it will start unloading container by container. When it has unloaded one container it will wait until an AGV arrives. If it is already there then it will put the container on the AGV, remove the AGV from AGVQ and wake up the waiting AGV by “Resume”. The AGV-agent can proceed now. The lecturer may interrupt the agents of resources now and explain that Quay crane and AGV were synchronized by using a queue QcQ. Many times it happens that the agent Quay crane already proceeds without signaling the AGV etc.

5 Sequencing the processes of agents

Now both Quay crane and AGV proceed with their actions simultaneously (Quay crane unloading another container, AGV driving to ASC). The lecturer is keeping track of the time, and gives turn to the student that has a first activity. To make the picture complete one could pay attention to the fact that the teacher actually performs the role of “sequence mechanism”.

The sequencing mechanism takes care of all state transitions and progress of time. During time an agent can be in one of three states:

1. Suspended or sleeping state. No moment in time has been defined for the agent to take action. It actually “sleeps” or plays the role of data element.
2. Scheduled state. A moment in time or a condition has been defined on which the agent should start or resume its actions.

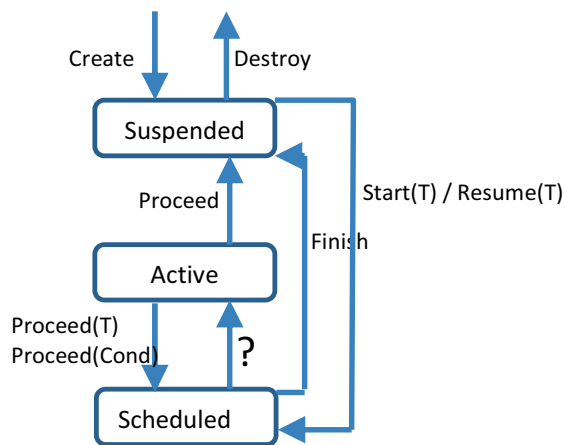


Fig. 3 Agent-states and transitions

3. Active state. The agent is actively executing actions (statements) until it tells the clock to proceed. The agent itself becomes scheduled or suspended then.

The question mark in the figure above shows that there is a mystery guest that changes states from “Scheduled” to “Active”. This mystery guest is the core of the simulation toolbox that operates according the real process oriented approach.

The lecturer should explain which of the transitions is being used at any moment when another agent becomes active. He/she could also illustrate what happens if two agents become Active at exactly the same moment. It will then be immediately clear that there is no real simultaneity, because only one processor is available for executing the statements of the active agent; only one agent can be active at any time. Both ways of synchronization, attributes (e.g. fence closed) and queues (e.g. QcQ is empty) are sensitive for the order of activation by the simulation toolbox.

6 Conclusions and recommendations

In most cases simulation of logistic systems is explained from the viewpoint of an observer who has to construct the model. He is supposed to have the necessary knowledge of the system and describes the system in terms of events, activities and/or processes.

The real process-interaction approach is a method that can represent the real system and its dynamics in a very natural way. To teach students to apply this method we used agent based role playing in which students are asked to identify themselves with or, in other words, to step into the shoes of the various elements in the system and to live their lives (process) as a function of time. In this approach difficult issues like element interactions and synchronisation appear in a clear natural way. This leads to more insight of the operation of the real system and a more deeply understanding of the simulation model.

We recently started to use this approach in practice. A first course has been completed with this type of role-playing and the results seem to be promising. The students seem to understand the synchronization of simulation modeling better and used queues and attributes of agents in a natural way when developing their own models. Roleplaying works very explanatory.

Now we will apply this method also in our research projects with industrial partners in order to construct and verify simulation models of design situations, and clearly focus and decide on problematic synchronization cases.

Fig. 3 Agent-states and transitions

Fig. 3 Agent-states and transitions

7 References

- [1] Fishman, G.S., "Discrete-Event Simulation", Springer-Verlag New York Inc., ISBN 0-387-95160-1, 2001
- [2] Kleijnen, J.P.C., Groenendaal, W.J.H. van, "Simulatie: technieken en toepassingen", Academic Service, Schoonhoven, ISBN 90 6233 322 2, 1988
- [3] Veeke, H.P.M., Ottjes, J.A., "TOMAS: Tool for Object-oriented Modelling And Simulation", Proc. Of Business and Industry Simulation Symposium, Washington, Ed. Maurice Ades, pp. 76 – 81, 2000
- [4] in 't Veld, Prof. J., " Analysis of organisation problems", Wolters-Noordhoff bv, Groningen, 8th edition, ISBN 90-207-3065-7, 2002
- [5] Veeke, H.P.M., Ottjes, J. A., Lodewijks, G., " The Delft Systems Approach", Springer, ISBN 978-1-84800-176-3, 2008
- [6] Zeigler, B.P., H. Praehofer, and T.G. Kim. 2000. "Theory of Modeling and Simulation", Academic Press, San Diego
- [7] Sierenberg, R.W., de Gans, O.B., "PROSIM text book", lecture notes Delft University of technology, Delft, 1982
- [8] Veeke, H.P.M., Ottjes, J.A., Lodewijks, G., "Experiences with process interaction based simulation in education and research" Proc. of the 2012 International Conference on Modeling, Simulation and Visualization Methods MSV 2012, Las Vegas, Ed. Hamid R. Arabnia, pp.109-113
- [9] Duinkerken, M.B., Ottjes J.A., "A simulation model for automated container terminals", Proc. of the Business and Industry Simulation Symposium (ASTC 2000). April 2000. Washington D.C. [SCS]. ISBN 1-56555-199-0

Air-Slag-Matte Interaction in a Peirce-Smith Copper Converter

Miguel A. Barron, Carlos A. Hernandez

Departamento de Materiales, Universidad Autonoma Metropolitana Azcapotzalco,
Mexico City, Mexico

Abstract - In order to analyze the interaction of fluids in a Peirce-Smith copper converter, the multiphase flow (air-slag-matte) is numerically explored in this work. The interaction of the matte, slag and air is studied by means of Computational Fluid Dynamics simulations. Jet velocities ranging from 1 to 100 m/s, which involve the bubbling and open jet regimes, were considered in the 2-D transient numerical simulations. A non-linear influence of the jet velocity on the matte velocity, is observed. Increasing jet velocity yields higher matte average velocity, however, beyond certain jet velocity further increments in this variable causes low matte velocity.

Keywords: Computational fluid dynamics, copper converter, multiphase flow, numerical simulation, Peirce-Smith converter.

1 Introduction

Nowadays, around 90% of the blister copper is produced by means of Peirce-Smith converter (PSC) using air to oxidize sulfur and reduce copper. Silica flux is added through the converter mouth to form a slag which collects matte impurities [1]. This device consists in a long horizontal cylindrical reactor where air is injected into a molten copper matte through submerged tuyeres [2].

Fluid flow in PSC is intentionally turbulent given that a high momentum transfer is required to obtain high chemical reaction rates and heat transfer. Many works have been reported regarding fluid flow in PSC. Some early works [3-4] report physical experiments to elucidate the bubbling to jetting flow regimes, and the effects of high pressure injection in copper converters. In [2] the influence of bath depth and tuyere submergence depth on the formation of standing waves in PSC is studied by means of water modeling. In the last decades, numerical simulations have been carried out to study the fluid mechanics in PSC. For example, in [5] the isothermal flow field in a PSC is obtained, and the numerical results are corroborated with $\frac{1}{4}$ scale water model. These authors report that large air bubbles increase the turbulence in the copper matte.

Recently, the fluid dynamics in a copper converter by means of physical and Computational Fluid Dynamics (CFD) modeling is studied [6-7]. In these works the bubbling and jetting flow regimes is analyzed by means of a dimensionless Froude number. The authors report the natural oscillation

frequency of the bath surface as a function of the bath and tuyere submergence depths.

In this work the multiphase flow (air-slag-matte) in a PSC is numerically studied varying the injection velocity from 1 to 100 m/s, considering constant the matte and slag initial depths. CFD software is employed to solve the transient 2D Navier-Stokes equations, the mass conservation equation, the K- ϵ turbulence model.

2 Mathematical Model

In a PSC, the isothermal momentum and mass conservation for the involved fluids (air, molten slag and molten matte) is represented through of the Navier-Stokes and the mass balance (continuity) equations [8]. The classical K- ϵ turbulence model is employed to simulate the turbulence [9]. Besides, a model is needed represent the multiphase flow in the PSC. The Volume of Fluid (VOF) [10] is employed for this, and its derivation comes from the assumption that two or more phases are not interpenetrating. In the VOF model is assumed that in each control volume the volume fractions of all phases sum to unity. The interface between the phases is obtained by solving the continuity equation for each phase. Finally, the Pressure Implicit with Split Operator (PISO) algorithm is employed in the computer simulations for the pressure-velocity coupling [11].

3 Numerical Solution

The conservation equations, together with the turbulence and VOF models, are solved employing the Computational Fluid Dynamics technique [11-12]. These equations are discretized using the 2D computational mesh shown in Fig. 1. The mesh contains 12 026 trilateral cells.

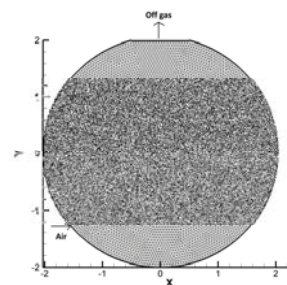


Fig. 1. The Peirce-Smith converter and the computational mesh employed.

The diameter of the considered PSC is 4 m, the nozzle diameter for air injection is 0.05 m, and the diameter of the converter mouth for off-gas exit is 0.5 m. The properties of the involved fluids are shown in Table 1. The considered inlet velocities of air are 1, 5, 10, 25, 50 and 100 m/s, in order to cover from the bubbling to the jetting regime [13].

To assure numerical stability, time steps employed for the integration of the balance equations are as follows: 0.001 s for air velocities of 1 and 5 m/s, and 0.0001 s for air velocities of 10, 25, 50 and 100 m/s. Besides, given that the process dynamics depends on the air injection velocity, the integration times were as follows: 10 s for velocities of 1, 5 and 10 m/s; 5 s for a velocity of 25 m/s, and 2 s for air injection velocities of 50 and 100 m/s.

Table 1. Physical properties of the converter fluids [1, 11].

Property	Air	Slag	Matte
Density, kg/m ³	1.225	2500	5200
Viscosity, kg/(m.s)	1.7894×10^{-5}	0.1	0.004

4 Results and Comments

In Fig. 2 are shown the molten matte distribution inside the PSC as function of the air injection velocities. The magnitude of matte agitation increases as the air injection velocity is increased. Fig. 2a show that for an air injection velocity of 1 m/s, an almost quiet flow in the matte is obtained, and the air goes upwards parallel to the lower left cylindrical wall of the converter. This situation is undesirable given that the matte agitation, and hence the heat transport and the chemical refining reactions are extremely low. As the air injection velocity is increased, the free surface of the matte becomes broken and the phenomenon of drop formation arises. In Fig. 2f, corresponding to an air injection velocity of 100 m/s, the matte flow is fully turbulent and the formed drops ascend beyond the geometrical center of the converter. If the injection velocity surpasses 100 m/s, the risk of spitting appears.

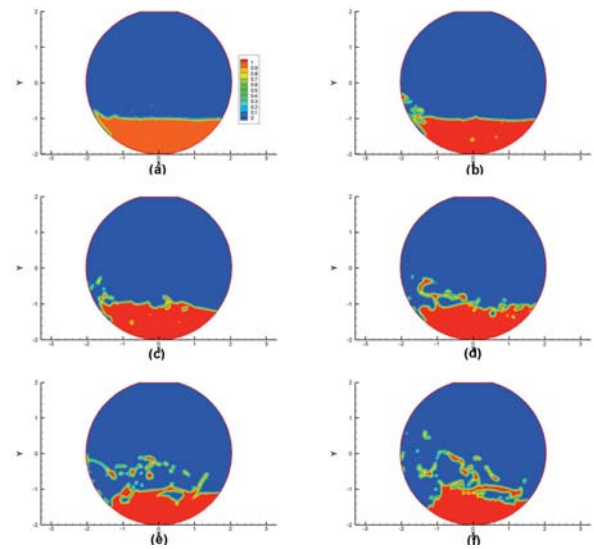


Fig. 2. Matte distribution inside of the copper converter as function of the considered air injection velocities. (a) 1, (b) 5, (c) 10, (d) 25, (e) 50 and (f) 100 m/s.

Fig. 3 depicts the slag distribution corresponding to the computer simulations of Fig. 2. For a low velocity of air injection of 1 m/s, Fig. 3a, the slag appears as an almost homogeneous layer floating quietly above the molten matte. Poor contact between the slag and the matte is appreciated, and given that the slag collects the matte chemical impurities, this is an undesirable state. As the air injection velocities are increased, the stirring and drop formation of slag is more evident, and from operational point of view, this conditions is favorable since the chemical reactions and heat transfer are enhanced.

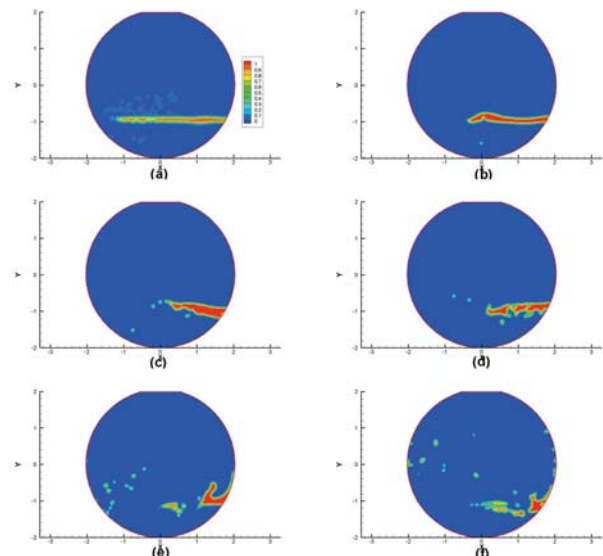


Fig. 3. Slag distribution inside of the copper converter as function of the air injection velocities. (a) 1, (b) 5, (c) 10, (d) 25, (e) 50 and (f) 100 m/s.

In [13, 14] it is reported that the bubbling to jetting transition is well measured by the Kutateladze dimensionless number (Ku). This number includes the basic forces that determine the above transition: air inertial forces, bubble buoyancy forces, molten matte gravity forces, and tension forces. The Kutateladze number is defined as [14]

$$Ku = \frac{U \sqrt{\rho_g}}{(\sigma_g(\rho_l - \rho_g))^{0.25}}$$

where U is the orifice gas velocity, ρ_g is the gas density, ρ_l is the liquid density, σ is the liquid surface tension (1.2 N/m), and g is the gravity force. Table 2 shows the Kutateladze number corresponding to the considered air injection velocities. In accordance to [13], for a similar PSC, the transition from bubbling to jetting regime occurs for air injection velocities from 50 m/s, i.e. $Ku \geq 3.4832$.

Table 2. Kutateladze numbers for the considered air injection velocities.

Air injection velocity, m/s	Kutateladze number
1	0.0704
5	0.3483
10	0.7037
50	3.4832
100	6.9664

5 Conclusions

The multiphase flow and the air-slag-matte interaction was studied in a Peirce-Smith copper converter Computational Fluid Dynamics simulations. Based on the numerical results, it can be concluded that an air injection velocity of 1 m/s causes non-significant stirring of the copper matte. If the jet injection velocity is raised, significant stirring of the matte and slag observed. The Kutateladze dimensionless number are calculated for the considered air injection velocities, suggesting that above 50 m/s the bubbling regime mutates to jetting regime.

6 References

[1] W.G. Davenport, M. King, M. Schlesinger, A.K. Biswas. *Extractive Metallurgy of Copper*, Pergamon, Oxford, UK, 2002.

[2] J.L. Liow, N.B.Gray. Slopping resulting from gas injection in a Peirce-Smith converter: water modelling. *Metallurgical Transactions B*, 21B (1990) 987-996.

[3] E.O. Hoefele, J.K. Brimacombe. Flow Regimes in submerged gas injection. *Metallurgical Transactions B*, 10B (1979) 631-648.

[4] J.K. Brimacombe, S.E. Meredith, R.G.H. Lee. High-pressure injection of air into a Peirce-Smith copper converter. *Metallurgical Transactions. B*, 15B (1984) 243-250.

[5] J. Vaarno, J. Pitkala, T. Ahokainen, A. Jokilaakso. Modeling gas injection of a Peirce-Smith converter. *International Conference on CFD in Mineral & Metal Processing and Power Generation*, CSIRO, Australia, 1997, pp. 297-306.

[6] A. Valencia, R. Paredes, M. Rosales, E. Godoy, J. Ortega. Fluid dynamics of submerged gas injection into liquid in a model of copper converter. *International Communications On Heat and Mass Transfer*, 31 (2004) 21-30.

[7] A. Valencia, M. Rosales, R. Paredes, C. Leon, A. Moyano. Numerical and experimental investigation of the fluid dynamics in a Teniente type copper converter. *International Communications On Heat and Mass Transfer*, 33 (2006) 302-310.

[8] R.B. Bird, W.E. Stewart, E.N. Lightfoot. *Transport Phenomena*, 2nd Ed. Wiley, New York, 2002.

[9] B.G. Thomas, Q. Yuan, S. Sivaramakrishnan, T. Shi, S.P. Vanka, M.B. Assar. Comparison of four methods to evaluate fluid velocities in a continuous slab casting mold. *ISIJ International*, 41 (2001) 1262-1271.

[10] C.W. Hirt, B.D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39 (1981) 201-225.

[11] Fluent 6.1 User's Guide. Lebanon, NH, 2003.

[12] J.H. Ferziger, M. Peric. *Computational Methods for Fluid Dynamics*, Springer, Berlin, Germany, 1999.

[13] M.A. Barron, C. Lopez, G. Plascencia, I. Hilerio. Large eddy simulation of bubbling-jetting transition in a bottom blown copper converter. *The 2010 International Conference on Modeling, Simulation and Visualization Methods*, WorldComp 2010, Las Vegas, NV, 2010.

[14] R. Sundar, R.B.H. Tan. A model for bubble-to-jet transition at a submerged orifice. *Chemical Engineering Science*, 54 (1999) 4053-4060.

Development of Cloud-Based HILS for Performance Verification of LNGC PMS

Junsang Seo¹, Sangoh Lee², Dukchan Jeon³, Jaemun Park⁴, Jun Soo Park⁵, and Kwangkook Lee^{*}

¹Institute of Convergence, USIS Co., Ltd, Ulsan, South Korea

²Institute of Convergence, USIS Co., Ltd, Ulsan, South Korea

³Institute of Convergence, USIS Co., Ltd, Ulsan, South Korea

⁴Institute of Technology & Research, OSLAB Co., Ltd, Changwon, South Korea

⁵Dep't of Naval Architecture & Ocean IT Engineering, Kyungnam University, Changwon, South Korea

^{*}Dep't of Naval Architecture & Ocean IT Engineering, Kyungnam University, Changwon, South Korea

Abstract – A power management system (PMS) has been an important part in a ship integrated control system. To evaluate a PMS for a liquefied natural gas carrier (LNGC), this study proposes a real-time hardware-in-the-loop simulation (HILS), which is composed of major component models such as turbine generator, diesel generator, governor, circuit breaker, and 3-phase loads on MATLAB/Simulink. In addition, a human machine interface (HMI) based on cloud system, real-control console, and main switchboard (MSBD) are constructed in order to develop an efficient control and a similar real environment in an LNGC PMS. More specifically, a comparative study on the performance evaluation of PMS functions is conducted using three test cases for sharing electric power to consumers in an LNGC. The result shows that the proposed system has a high verification capability for the operating function and failure handling evaluation as a PMS HILS.

Keywords: Cloud System, HIL (Hardware-in-the-loop), SIL (Software-in-the-loop), PMS (Power Management System), LNGC (Liquefied Natural Gas Carrier)

1 Introduction

With the increasing risk in building liquefied natural gas (LNG) vessels, pre-simulation with various scenarios is needed for system integration as well as safe operation. In particular, a power management system (PMS) in a liquefied natural gas carrier (LNGC) is an important part, which operates in tight integration with power control systems to achieve the desired performance and safety. A PMS can control system frequency and voltage as well as the generated real and reactive power [1]. In addition, it usually has a function to prevent breakdown of power generation and power consumption. Failure in the PMS will affect safety and lead to downtime, and may even cause accidents. For these reasons, electrical power management systems have been studied [2 – 3].

To evaluate the performance of a PMS, there are existing methods such as direct on-site verification and software-based simulation. Among these methods, the direct verification technique evaluates the function of a PMS directly by using analog/digital simulator on real condition. However, it has high physical cost and is risky. On the other hand, results of software-based simulations strongly rely on the model accuracy of power system components, and pure hardware testing lacks flexibility on establishing a complex power system [4].

To solve these problems, hardware-in-the-loop (HIL) simulation is released to enhance the quality of hardware testing. This system reduces the cost of verifying problems such as system malfunction, incorrect calculated configuration parameters, and system errors according to rules and regulations. HIL simulation can provide performance testing, verification, evaluation, development, and diagnosis of electronic equipment solutions [5]. However, domestic shipbuilding companies, which initiate PMS requests to international PMS evaluation agency, pay a high cost because the said companies and the institute of marine equipment research cannot verify it by themselves [6].

To address this problem, this study develops a localized real-time HIL system for marine equipment to evaluate the PMS controller in an LNGC. For operating the HILS, the major components of the LNGC are modeled using MATLAB/Simulink. The power supply model consists of two turbine generators, a diesel generator, and governor. The power consumer model is composed of side thrusters, cargo pumps, ballast water pumps, and lumped loads, which are mostly consumed in an LNGC.

These models are operated in NI PXI by using LabVIEW programming to simplify the complexity of HILS. Unlike existing simulators, the proposed methodology can also utilize a control console (CC) and a main switchboard (MSBD) onboard to model a real-ship environment. A method for communicating CC and MSBD is then developed based on the OPC server/client technology through Ethernet communication. To achieve a convenient monitoring system, cloud-based monitoring is implemented on HILS. Furthermore,

this system uses load sharing test cases for evaluating PMS functions with the proposed HIL test bed.

The rest of the paper is organized as follows. Section 2 includes the necessary background information about PMS roles in an LNGC and the standard SIL/HIL system. Section 3 introduces our proposed configuration of PMS HIL framework. Section 4 presents the experimental results, and the conclusions are provided in Section 5.

2 Background Information

2.1 Power Management System in LNGC

The term “power management system” was used before to describe procedures for the automatic starting and stopping of electrical generators to meet actual load requirements. However, it is now applied to a very wide range of control systems, even including what really are “energy management systems” [7].

A PMS in an LNGC is a programmable logic controller for high voltage (HV) and low voltage (LV) switchboards, generators, and prime mover control. The system operates the normal functions necessary to manage the diesel and turbine generators in order to balance power generation and power consumption. The PMS is interfaced with HV main switchboards, HV main cargo switchboards, and LV switchboards through hardware (digital inputs or outputs and analog inputs) or Ethernet cable.

2.2 HIL (Hardware-in-the-loop)

Hardware-in-the-loop (HIL) simulation is a technique that is used for developing and testing complex real-time embedded systems. HIL simulation provides an effective platform by adding the complexity of the plant under control to the test platform. The complexity of the plant under control is included in the test and development by adding a mathematical representation of all related dynamic systems [8]. In particular, HIL simulation has high expandability to apply to embedded systems in vehicles, aircrafts, vessels, and on/offshore plants.

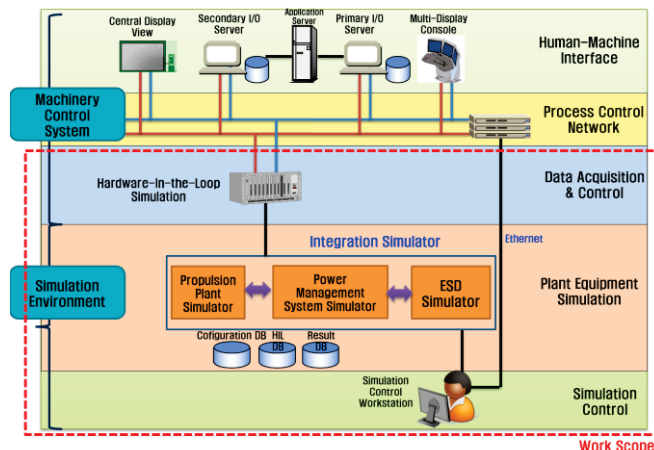


Figure 1. HILS Structure for Ship Management System

In shipbuilding and on/offshore plant fields, HILS is used for drilling operation control, power generation/distribution, and dynamic positioning. Marine Cybernetics provides services about SILS/HILS solutions in marine engineering since 2002. This service can reduce the enormous cost under a number of incidents caused by partial and complete blackout.

Figure 1 shows the HIL simulation structure for ship management system. HILS consists of human machine interface (HMI), process control network, data acquisition & control, plant equipment simulation, and simulation control part. Generally, HIL simulation that applies various fields has to develop equipment simulation and simulation control.

3 Configuration of PMS HIL Testing Bed

3.1 Simulation Model

The proposed power simulation model consists of a power supply and power consumer to operate two turbine generators, diesel generators, bow thrusters, cargo pumps, ballast pumps, and lumped loads in the LNGC. The specifications of the diesel and turbine generators are described in Table 1. In addition, the specific information of the power consumer is listed in Table 2.

Table 1. Specification of Diesel and Turbine Generator

Diesel and Turbine Generator	
Max Power	3.45 MW
Voltage	6,600 V
Frequency	60 Hz

Table 2. Specification of Power Consumer

	Max Power (kW)	Voltage (V)	Frequency (Hz)
Bow Thruster	1,800	6,600	60
Cargo Pump	530	6,600	60
Ballast Pump	330	6,600	60
Lumped Load	1,000	440	60

Figure 2 presents the overall circuit of the power plant simulation models by using MATLAB/Simulink. These models are implemented by SimPowerSys libraries, which provide power model library for easy modeling.

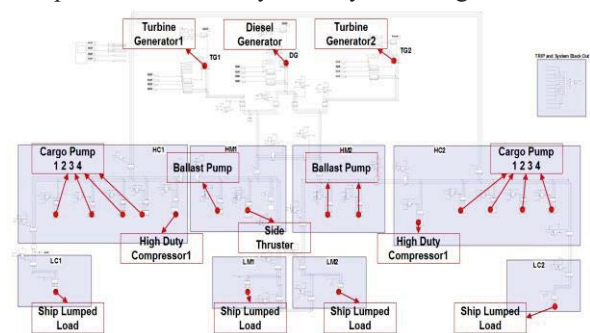


Figure 2. Overall Circuit of Power Supply and Power Consumer on MATLAB/Simulink

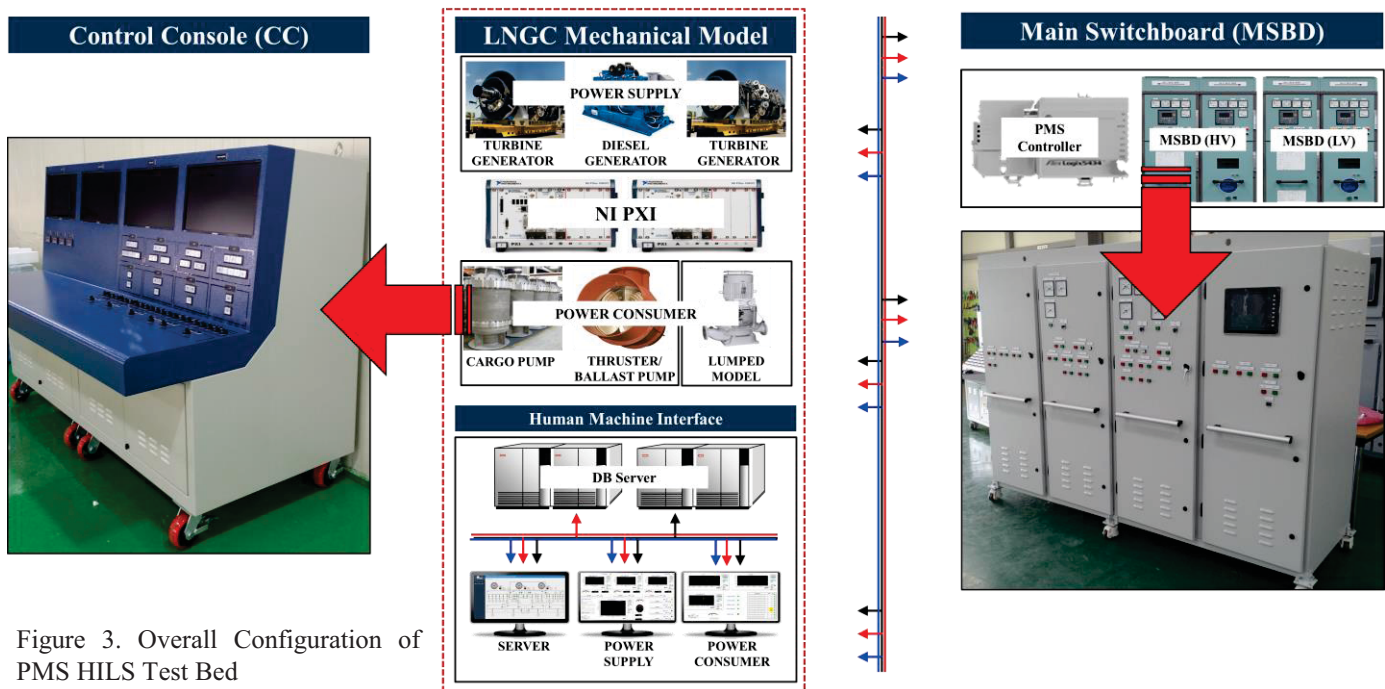


Figure 3. Overall Configuration of PMS HILS Test Bed

3.2 PMS HIL Interface

In this study, a control console (CC) and main switchboard (MSBD) are utilized in order to make a similar environment in the LNGC. The control console is installed in the main engine control unit of the ship, alarm monitoring device, and other control equipment, checking for her defects and malfunctions. For easy monitoring and simulation, the layout of the panel surface is sectioned into four separate groups such as power supply, power consumer, marine environment, and server parts. In this study, the marine environment part is not mentioned.

Table 3. Hardwire Converting Table

	Simulation Model (PXI)	MSBD
Voltage	0~8,000 V	4~20 mA
Frequency	55~65 Hz	4~20 mA
Power	0~4,500 kW	4~20 mA
Current	0~380 A	4~20 mA

Main switchboards are not only recognized as useful means of power source protection but also highlighted as central means for controlling power. The existing MSBD is commonly used for protection and switching of transformers, motors, generators, capacitors, buses, distribution feeder lines, and in general, for protection of any high and low voltage power circuit. For application in the power plant model and interfacing with PXI and CC, the MSBD is manufactured on operating current values (4~20 mA). The analog condition values of the simulation model are converted to fixed current values through hardwire in Table 3.

Under this process, the proposed system scales the output currents in PXI, which can simulate real-analog signals in the LNGC. In addition, this study deploys the PMS HIL simulator for similar LNGC environments in Figure 3. Analog signals are connected by hardwire while the server PC, National Instrument PXI, and Allen Bradley programmable logic controller (PLC) communicate through digital signals by OPC client/server technique on LabVIEW. OPC server provides minimum scan time of 50 μ s, which satisfies the PMS communication time of 100 μ s.

3.3 Human Machine Interface (HMI)

The implemented HMI of the PMS HIL simulator consists of two monitoring sections (condition monitoring and analysis monitoring) based on HTML5. This system mainly uses the HTML5 technology with web chart application, in which ASP and Javascript are used to build the web chart pages, and AJAX is utilized to reduce a page reloading time.

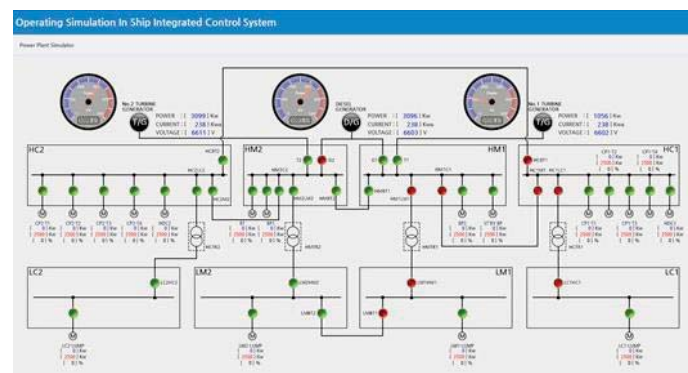


Figure 4. Condition Monitoring of PMS HIL Simulator

The first section is condition monitoring, which includes output values (circuit breaker condition, power, current, and voltage by power supply and consumer) shown in Figure 4. The HMI is connected to the programmable power equipment models in PXI with two types of signal: double type signal and Boolean signal. Double type signals are output values of the power supply and power consumer, and circuit breaker condition values are presented as Boolean type signals. Communication is achieved using Modbus TCP under a speed of 100 μ s.

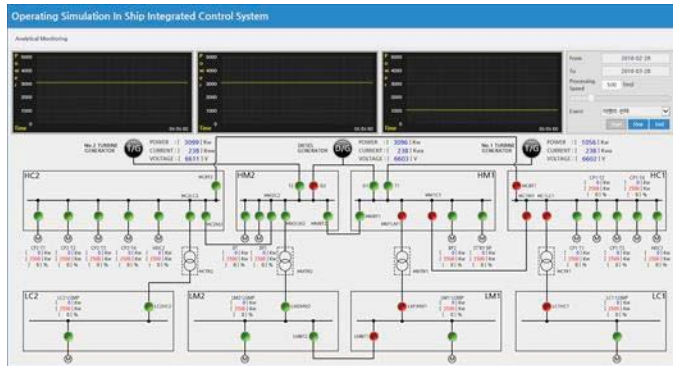


Figure 5. Analytical Monitoring of PMS HIL Simulator

The other section is analytical monitoring, which can analyze circuit condition values using the speed control function according to test cases. In Figure 5, this web page can evaluate a specific condition at current or previous time, unlike in real-time environment where a user cannot identify the time passed windows. In addition, if we need a detailed analysis when an unexpected occurrence and sudden situation are generated, it easily finds the specific condition and situation.

4 Experimental Results

To evaluate the PMS, an actual test is carried out under the PMS HIL simulation environment. The proposed HIL simulator is verified with the functional specification of the PMS. The performance verification between the power generator control and the PMS is to be confirmed through load sharing test, governor test, and parallel running test. Among these test cases, load sharing is the fundamental and most critical part for evaluating the entire functionality of the PMS. Therefore, this study performs a load sharing test as the main function of the PMS. Load sharing is classified into symmetric, asymmetric, and fixed load sharing.

4.1 Symmetric Load Sharing

In symmetric load sharing mode, the loads of generators running in parallel have to be equal within a small dead band ($\pm 3\%$) of rated power. As shown in Table 4, the experimental results present a similar condition, which generates total power (5775 kW).

Table 4. Result of Symmetric Load Sharing

Item	Conditions	Result
Symmetric load sharing	- Power Consumer: 5,775 kW	- Power Consumer: 5,775 kW
	- TG1: 1,925 kW	- TG1: 1,931 kW
	- TG2: 1,925 kW	- TG2: 1,931 kW
	- DG: 1,925 kW	- DG: 1,913 kW
	- Dead band: $\pm 3\%$ of rated power	Satisfied

4.2 Asymmetric load sharing

Asymmetric load sharing mode evaluates the operating functions of three generators; the selected generator (master) is loaded to 80% while the other generators (slave) will share the load. In this study, test conditions are selected as follows: Three generators are operated as TG1 (master), TG2 (off), DG (slave), and the total load is 4729 kW, including a side thruster (1,800 kW), three cargo pumps (each 530 kW), a ballast pump (330 kW), and a ship lumped load (1,000 kW).

If TG1 is changed in order to supply the remaining power to consumers in asymmetric load sharing mode, DG will be also changed automatically to master. If a non-slave generator takes heavy load (90%) or light load (20%), a master generator will increase/decrease its power respectively, to prevent overload or reverse power on the non-selected generator. Figure 6 demonstrates the same results that Table 5 provides in the PMS under asymmetric load sharing test condition.

Table 5. Result of Asymmetric Load Sharing (TG1 or TG2)

Item	Conditions	Result
Asymmetric load sharing	- Power Consumer: 4,720 kW	- Power Consumer: 4,746 kW
	- TG1: 2,760 kW (80%)	- TG1: 2,754 kW
	- TG2: OFF	- TG2: OFF
	- DG: 1,960 kW	- DG: 1,992 kW
	- Dead band: $\pm 3\%$ of rated power	Satisfied



Figure 6. Result of Asymmetric Load Sharing in PMS

4.3 Fixed load sharing

To maintain the load sharing function of the selected generator, it is possible to choose a generator in steady load. This mode cannot be selected when the generator is in standby mode or when it is only one generator online. If a non-selected generator takes heavy load (90%) or light load (20%), a selected generator will increase/decrease its power respectively to protect the dangerous situation when the non-selected generator is faced with overload or reverse power condition.

The test condition is similar to asymmetric load sharing when the percentage of TG1's total power is set to 70%. Figure 7 demonstrates the same results that Table 6 provides in the PMS under fixed load sharing test condition.

Table 6. Result of Fixed Load Sharing (TG1 or TG2)

Item	Conditions	Result
Fixed load sharing	- Power Consumer: 4,720 kW	- Power Consumer: 4,735 kW
	- TG1: 2,415 kW (70%)	- TG1: 2,426 kW
	- TG2: OFF	- TG2: OFF
	- DG: 2,304 kW	- DG: 2,309 kW
	- Dead band : $\pm 3\%$	Satisfied



Figure 7. Result of Fixed Load Sharing in PMS

5 Conclusions

HIL is an important solution to evaluate a PMS for an LNGC, and it is one of the most well-known evaluation techniques that is used in various fields. However, domestic shipbuilding companies and the institute of marine equipment research cannot evaluate PMS HIL by themselves. To address this issue, this study proposed a PMS HIL simulator, which is configured with power supply/consumer models, CC, MSBD, and HMI. The proposed HIL simulation platform used real-equipment data in marine industry in order to make a similar LNGC environment.

In addition, this study utilized load sharing test cases of a PMS. Comparative testing results indicate that the proposed system shows a great potential for symmetric, asymmetric, and

fixed load sharing. To make it more useful, various PMS test cases will be evaluated under the proposed PMS HIL simulator. In addition, further system developments will still be required for ship automation from PMS control as well as energy management system in future works.

Acknowledgment

This research was supported by National IT Industry Promotion Agency (Grants No. S0170-15-1078) and Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration (Grants No. C0333413) in 2016. All of the support is gratefully acknowledged.

6 References

- [1] Parizad, A., "Dynamic stability analysis for Damavand power plant considering PMS functions by DlgSILENT software", Environment and Electrical Engineering (EEEIC), 2013
- [2] X. J. Tang, T. Wang, C. Zhi, and Y. M. Huang, "The design of power management system for solar ship", Transportation Information and Safety, 2015
- [3] S. V. Giannoutsos and S. N. Manias, "Energy management and D/G fuel consumption optimization in the power system of marine vessels through VFD-based process flow control", Environment and Electrical Engineering, 2015
- [4] Yu Zhou, Jin Lin, Younghua Song, Yu Cai, Hao Liu, "A power hardware-in-loop based testing bed for auxiliary active power control of wind power plants", Electric Power Systems Research, 2015
- [5] H Huang, M Pan, Z Lu, "Hardware-in-the-loop simulation technology of wide-band radar targets based on scattering center model", Chinese Journal of Aeronautics, 2015
- [6] Parizad A., "Requirement analysis and architecture establishment for PMS FMEA simulator based on SILS", Proceedings of the Annual Autumn Conference, 2014
- [7] Wikipedia, the free encyclopedia, "Hardware-in-the-loop simulation", 2016

A Pilot Study to Explore the Possibilities of an Interactive Multipurpose Exergaming Simulator for Senior Activation

Nurkkala, V-M.¹, Kalermo, J.¹, Endo Y.² and Goto M.²

¹Kajaani University of Applied Sciences, Kajaani, Finland

²Sendai University, Funaoka, Japan

Abstract – Globally, the population aging is causing increasing healthcare expenditure. It has been commonly recognized, that preventive actions that activate aging people to exercise more is the key to reduce the demand for health care. Our objective was to explore the possibilities of exergaming simulator as means of elderly activation and promoting physical activity. This paper presents the pilot study carried out in Funaoka, Japan, to explore the possibilities and usability of the exergaming simulator in activating Japanese elderly people in their physical and cognitive capabilities.

Keywords: Exergaming, interactive, aging population, senior activation, simulator

1 Introduction

Nearly all countries in the world are facing the phenomenon of population aging. According to the United Nations, the number of older persons (aged 60 years or over) is expected to more than double globally, from 841 million people in 2013 to more than 2 billion in 2050 [1]. The aging of the population is associated with higher health expenditure as elderly people are more likely to demand more health care than do younger adults. Japan, which is the most aged country in the world, spends about \$3,120 per capita on health [1]. It has been commonly recognized, that preventive actions that activate aging people to exercise more is the key to reduce the demand for health care. This has motivated companies and research organizations to develop new products and services that activate physical and cognitive skills, and aim having a positive effect on the health and wellbeing of people.

New technologies and applications for health and wellbeing are introduced to markets in an increasing speed. Among others, exergaming is rapidly growing market sector. According to Oh and Yang, the most common definition of exergames is “video games that require physical activity in order to play” [2]. In this article, we define exergaming as playing a video game solution which inspires and motivates people to exercising by taking advantage of different technologies.

With the increasing fight against obesity, inactivity and increasing healthcare expenditure, as well as the trend of

seeking for healthy lifestyle, the exergaming market is likely to expand quickly and create new markets. The market volume of serious gaming, in which exergaming plays a notable role, is estimated to be \$4,8 billion up to \$12 billion, and the markets are expected to have annual growth of 15% up to year 2017 [3]. Augmented and virtual reality, and the interaction enabled by these technologies, may well be the next revolutionary feature in fitness and wellness products and services. Since 2013, partly as a response to the prospects of future demands, we have been developing an interactive multipurpose simulator for exergaming.

The aim of the development was to provide motivating and inspiring content for people who exercise with cardio devices. Video films have been integrated with e.g., exercise bikes and treadmills, which are likely to make training more appealing than cardio training without the video scenes. However, most of the available solutions lack one widely appreciated feature, interaction. In addition, most of the available solutions are linked to a certain device. Our objective was to create a multipurpose solution that could be easily integrated with versatile cardio and rehabilitation equipment as well as be suitable for different target groups.

In this paper we will first discuss the existing research on exergaming. Next we will present the software and hardware of our exergaming solution. We have created different content and exercises for different target groups and integrated the software with various equipment such as treadmill, exercise bike, cross cycle, and restorator bike. With different screen solutions, varying from virtual reality goggles or one small screen to three wall projection cave, sounds of nature, and motion control with Kinect controller, we have been able to provide immersive and interactive exercising, rehabilitation or physical activation experiences to people of all ages. We have been exploring the utilization and usability of the exergaming simulator for different uses, such as gym training, senior activation, stroke rehabilitation, exercise testing and exergaming for children. We will briefly present the pilot cases that we carried out within the development process to explore the suitability of our solution for different target groups, and present more thoroughly the pilot case in which we explored the possibilities of our exergaming simulator for senior activation.

2 Research on exergaming

Several research groups, individual researchers and laboratories have concentrated on the scientific study, development, and/or testing of exergaming products in order to examine possible benefits of the use of the exergame devices. Yet, the experimental research on exergaming is rather limited. The research has concentrated, for example, on possible physical and psychological benefits of exergames for different ages (e.g., children, seniors) [4][5][6] and for different target groups (e.g., inactive children, rehabilitation groups) [7]. Also the use of virtual environments in exercising has been studied [8]. The results indicate generally that exergames have positive psychological and physical impact to the studied groups. The studies have shown, for example, an increase in the exercise motivation [9], physical activity [7] and energy expenditure while playing exergames [6][10] as well as improvement of the balance [5], mood and attention after playing [11]. However, also some studies exist with no clear evidence of the benefits [12], but no harmful effects have been reported.

3 Description of the interactive multipurpose exergaming simulator

Our exergaming simulator is based on Unity game engine. The software includes various virtual environments with a free run option in which the user can explore the area freely, or use several routes of different lengths and difficulty levels. The first available virtual environments included Finnish forest and city environments, a tropical island and mountain scenery, but recently several new virtual environments have been added to the software. The software also includes variety of exercising modes, such as jogging, biking, orienteering and adventure. Gaming plays a major role in all exercising modes. For example, it is possible to challenge a friend in an adventure game or in future, participate in competitions in which several people are participating via Internet. Also in fitness testing the users may challenge themselves or their friends by competing against avatar characters based on their previous exercise results. The different exercising and rehabilitation devices are integrated with the software with a small in-house developed device called Athene Communication Device (ACD). The ACD enables the connection between the cardio device and exergaming software in a way that the speed in the virtual environment corresponds with the speed of the cardio device. With Kinect motion controller, the users use gestures to control the direction in which the virtual environment moves (Figure 1).



Figure 1. The exergaming simulator

4 Pilot cases

The exergaming simulator has been piloted for various target groups and purposes during its development in 2013-2014. First, the exergaming simulator was piloted in July 2013 as an orienteering simulator during the World Orienteering Championships in Vuokatti, Finland. The second pilot case was carried out in Health Club Hukka, Oulu, Finland for gym users, following by a pilot case in which the exergaming simulator was explored for exercise test use in Vuokatti Olympic Training Centre in Finland. The fourth pilot case was targeted to kids and youngsters while the exergaming simulator was tested in Angry Birds Vuokatti activity park in Finland. These pilot cases are presented in our previous article [13]. Finally, the exergaming simulator was piloted for stroke patient rehabilitation and senior activation purposes. In this paper, we concentrate on the pilot case in which we explored the possibilities and usability of the exergaming simulator for senior activation in Japan. The pilot study was made in collaboration with University of Sendai.

5 Methods

The aim of this pilot study was to explore the possibilities and usability of the exergaming simulator in activating Japanese elderly people in their physical capabilities. The experiment was carried out at Sendai University in Funaoka, Japan in September 2014. Subjects' task was to explore the virtual city and play Collecting Bananas game. Some of the subjects also carried out an extra task, which included wandering in a virtual Finnish forest environment. The exergaming set up included OxyCycle 3 restorator bike for arms and legs, the legs-option was used in this study (Figure 2); PC and the exergaming software; Athene Communication Device (ACD) card to integrate the restorator bike with the exergaming software; video projector and screen; and Microsoft Kinect motion controller.



Figure 2. OxyCycle 3 rehabilitation device

Twelve (12) Japanese elderly people participated in the study (8 females and 4 males). The average age of subjects was 75.4 years, the youngest being 64 years and the oldest 87 years. The subjects were divided into two groups. Group 1 consisted of subjects ($n=5$) who either lived or visited regularly in the elderly people day care center and were provided with relatively good facilities and possibilities for daily exercising and rehabilitation. Group 2 ($n=7$) consisted of people who were living in a Tsunami shelter, in where they had rather limited space and limited possibilities to do exercising.

Three of the subjects had played computer games, mobile games (mobile phone or tablet) and/or games with game consoles (e.g. Nintendo, PlayStation) before (one reported "I have tried a couple of times", one reported "I play several times per month" and one reported "I play every week"). Two of the subjects had experienced virtual environments before. Four of the subjects had tried a restorator bike before.

None of the subjects reported having any illnesses or other constraints that prevent his/her physical activity nor any illnesses or other constraints that prevent using the restorator bike in the testing. A health check was carried out before the test (incl. blood pressure and questionnaire). After the health check, the exergaming simulator equipment and tasks were introduced to subjects.

The first task (Task 1) was designed to get the subjects familiarized with the exergaming simulator. The task was to navigate in the virtual City of Kajaani, where the route selection was made in the crossroads. The choice of the route was tracked with Kinect motion controller which recognized the body gestures. The left turn was caused by raising left hand and right turn by raising right hand. If the subject did not raise either hand, the route continued straight forward. The task lasted no more than three (3) minutes.

The second task (Task 2) was to collect as many bananas as possible within 2.5 minutes (Figure 3). Different routes were made in the virtual city of Kajaani. In each route, there

was a certain amount of bananas in the middle of the street to be collected by going through them. Each banana was worth of one point and only a limited number of routes could be chosen. Depending on the route choices and the speed, it was possible to collect more or less points. The game stopped automatically when the time limit was reached or if the subject reached the last point in the virtual map. Each subject had three trials, thus it was possible to memorize whether the route selections in the previous trials were good or not. After each trial in this task, the number of collected bananas was shown on the result screen. The distance, max speed, time, collected points and average pace (minutes/kilometers) of the subjects were measured in each trial to see if there are some changes in these values between the trials.



Figure 3. The task 2 included "Collecting Bananas game", in which the subjects were instructed to collect as many bananas as possible in 2.5 minutes.

After finishing the first two task, the subjects were interviewed about their experiment. The interviewer asked subjects' subjective evaluations of the usability of the system and how fun did the subjects consider different features. The subjects were also asked for development ideas; what kind of features would they wish to see in this kind of exergaming simulator.

After the interview, the Group 1 ($N=5$) was offered the bonus task in order to get feedback of other kind of exercise. The bonus task included a Finnish forest route (Figure 4) in which the task was just to wander in the forest where the birds were singing. The route was so called restricted route, where it was not possible to choose the direction but the character followed the route automatically. This exercise lasted for 3 minutes.

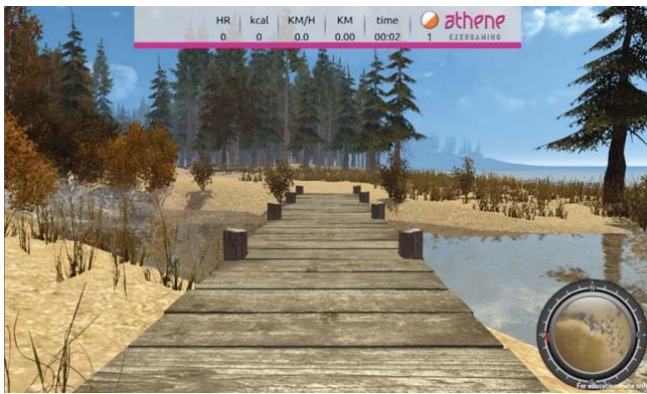


Figure 4. The Finnish forest scene (Virtual Vuokatti)

6 Results

The aim of task 1 was to get the subjects familiar with the exergaming simulator. For some subjects there were some difficulties to change the direction in the crossroads, but in most of the cases the subjects did not encounter any difficulties performing the choice of direction. After performing task 1, all the subjects knew how to use the exergaming simulator and were ready to move to task 2. The task of exploring the virtual city was considered fun by the subjects. The grade for task 1 was 4.4 points (“Exploring the virtual city was fun”; on scale 1-5, where 1 = totally disagree, 5 = totally agree; N=12).

In task 2 (Collecting Bananas game), remarkable change of speed between trials was noticed (Figure 4). All of the subjects raised their speed during the three trials. Also the number of collected points increased along new trials. The average of collected points of all subjects was 13.7 points in trial 1 and 16.3 points in trial 3. Thus, the average increase in collected points was 2.6 points (19% increase in collected points). The average distance of all subjects in trial 1 was 550 meters and in trial 3 the average distance was 610 meters. Thus, the increase in distance was in average 60 meters (11% increase in the distance). This can be well seen in the decrease in pace between the trials. The Figure 5 shows the changes in the pace (seconds/kilometers) from trial 1 to trial 3. The smallest change was three seconds per kilometer decrease in pace whereas the biggest change was with one subject, who did the last trial with 2 minutes 24 seconds per kilometer faster pace than the first trial. The average change in the pace was 43 seconds/km. The decrease in pace could be related to subjects’ better understanding how to move and change the direction (learning effect); the increase in the encourage to use device, but also in the competitive instinct that aroused in some subjects.

The task 2, Collecting Bananas game, was considered fun by the subjects. The overall grade for Collecting Bananas game was 4.6 points (“It was fun to collect the bananas”; on scale 1-5, where 1 = totally disagree, 5 = totally agree; N=12).

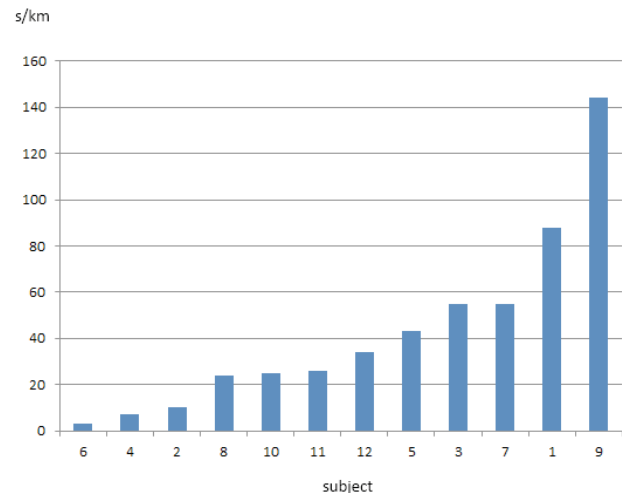


Figure 5. In Task 2, the decrease in the pace (s/km) varied between trials 1 and 3 from three seconds/km to 2 minutes and 24 seconds/km, and the average decrease in pace was 43 seconds/km; N=12.

Finally, the overall grade for wandering in Finnish forest environment (Task 3) was 5.0 points (“It was nice to move in the virtual forest”; on scale 1-5, where 1 = totally disagree, 5 = totally agree; N=5). None of the participants reported simulator sickness symptoms during or after experiments.

7 Conclusions

The Japanese seniors who participated in this experiment were in general in relatively good physical shape. They found the exergaming simulator as a motivating and appealing means to do physically activating exercises. However, in order to make better conclusions of usability and benefits of the exergaming simulator for elderly people, more research is needed. It would be interesting to add biometric measurements in the study protocol, such as heart rate monitoring, recording of muscular activity and tracking the eye movements. These biometric measurements could give a deeper insight for instance to user experience as well as physical and mental workload. It would also be very interesting to explore the differences between e.g., Japanese and Finnish cultures in the use of exergaming for seniors.

Subjects commented that they would wish to see e.g., flowers in the virtual environments. Many of the subjects mentioned in the interview, that seeing moving objects (e.g., cars, walking people, birds, squirrels and other animals) in the virtual environment would be nice. Exercising with the exergaming simulator and virtual environments was generally considered fun and motivating, and most of the subjects stated they would like to use this kind of exercising environment also in the future.

Acknowledgements

The multipurpose exergaming simulator has been developed at Kajaani University of Applied Sciences in collaboration with University of Jyväskylä and University of Oulu. Numerous companies have provided their expertise and resources for the R&D. We are grateful of all the support we have received from our partners. The research and development of the exergaming simulator has been funded by European Social Fund, European Regional Development Fund, Finnish Funding Agency for Innovation (TEKES), Joint Authority of Kainuu and numerous companies.

References

- [1] United Nations, Department of Economic and Social Affairs, Population Division. (2013). World Population Ageing 2013. ST/ESA/SER.A/348. Available online: <http://www.un.org/en/development/desa/population/publications/pdf/ageing/WorldPopulationAgeing2013.pdf><http://www.un.org/en/development/desa/population/publications/pdf/ageing/WorldPopulationAgeing2013.pdf>
- [2] Oh, Y. & Yang, S. (2010). "Defining Exergames and Exergaming." In *Proceedings of Meaningful Play*, 1-17.
- [3] Bohle, S. (2014). Opportunities, challenges, worldwide trends. Nordic Digital Business Summit 2014 conference presentation. <https://www.youtube.com/watch?v=nLV2KpplU4U>
- [4] Brox, E., Luque, F.L., Evertsen, G.J. & Hernandez, J.E.G. (2011). "Exergames for Elderly: Social Exergames to Persuade Seniors to Increase Physical Activity." Presented at the 2011 5th IEEE International Conference on Pervasive Computing Technologies for Healthcare (Pervasive Health).
- [5] Lamothe, C.J., Caljouw, S.R. & Postema, K. (2011). "Active Video Gaming to Improve Balance in the Elderly." *Stud Health Technol Inform* 167: 159-164.
- [6] Graf, D.L., Pratt, L.V., Hester, C.N. & Short, K.R. (2009). "Playing Active Video Games Increases Energy Expenditure in Children." *Pediatrics* 124 (2): 534-540.
- [7] Fogel, V.A., Miltenberger, R.G., Graves, R. & Koehler, S. (2010). "The Effects of Exergaming on Physical Activity among Inactive Children in a Physical Education Classroom." *Journal of Applied Behavior Analysis* 43 (4) : 591-600.
- [8] Smith, B.K., (2005). "Physical Fitness in Virtual Worlds." *Computer* 38 (10): 101-103.
- [9] Sanders, S. & Hansen, L. (2008). "Exergaming: New Directions for Fitness Education in Physical Education [Policy Brief]." Tampa: University of South Florida, College of Education, David C. Anchin Center.
- [10] Graves, L., Stratton, G., Ridgers, N.D. & Cable, N.T. (2007). "Comparison of Energy Expenditure in Adolescents When Playing New Generation and Sedentary Computer Games: Cross Sectional Study." *British Medical Journal* 335: 1282-1284.
- [11] Russell, W.D. & Newton, M. (2008). "Short-term Psychological Effects of Interactive Video Game Technology Exercise on Mood and Attention." *Educational Technology & Society* 11 (2): 294-308.
- [12] Daley, A.J. (2009). "Can Exergaming Contribute to Improving Physical Activity Levels and Health Outcomes in Children?" *Pediatrics* 124 (2): 763-771.
- [13] Nurkkala, V.-M., Kalermo, J. & Järvillehto, T. (2014). Development of Exergaming Simulator for Gym Training, Exercise Testing and Rehabilitation. *Journal of Communication and Computer*. David Publishing Company.

SESSION

NOVEL ALGORITHMS AND APPLICATIONS + VISUALIZATION + AUGMENTED REALITY + NUMERICAL METHODS

Chair(s)

TBA

3D Printing for Visualisation of the Complex Physical Structures of Agent-Based Simulation Models on Lattices

K.A. Hawick, L.R.F. Odiam and L.A.D. Stockwell

Computer Science, University of Hull, Cottingham Road, Hull HU6 7RX, UK.

Email: { k.a.hawick, l.odiam, l.stockwell } @hull.ac.uk

Tel: +44 01482 465181 Fax: +44 01482 466666

March 2016

ABSTRACT

Visualising the complex emergent spatial structure of large-scale agent-based models is a challenge that is only partially addressed by 3D rendering. We explore the use of 3D printing technology to construct physical artifacts from lattice-oriented models such as the Kawasaki and Potts models in 3D. We describe the problems of support structure, resolution and overhangs in constructing physical 3D print models, and describe our work using additive manufacturing technologies including both both Filament Fabrication and Powder-Bed Fabrication. We experimented with cut away approaches to reveal the complex internal structure of the emergent model configurations and we describe our techniques for generating 3D printable artifacts for models of this nature. We found that powder bed technology enabled quite crisp coloured model components, and present some photographic examples of printed complex model system configurations, showing results of quenching and annealing in stochastic lattice simulations.

KEY WORDS

3d-print; additive manufacture; modelling; simulation; agent-based model; simulations; lattice models.

1 Introduction

Simulating complex systems such as agent-based models can yield great insights into emergent behaviour, and this is enhanced if whole model systems can be visualised to identify spatial structures. Visualising models with graphical rendering technology is a powerful and now well established approach, with virtual reality technologies becoming commodity priced and accessible to wider user groups.

However, being able to examine a physical artifact can also provide significant and different insights into emergent spatial structures and growth properties [35]. In this article we report on experiments with a range of 3D printers and associated technologies to make physical 3D printed constructs from some simulation models such as Kawasaki diffusion model [24] on a lattice and multi species Potts [31] model variants [15,21] of it.

3D printing technology [19, 34] has developed considerably in

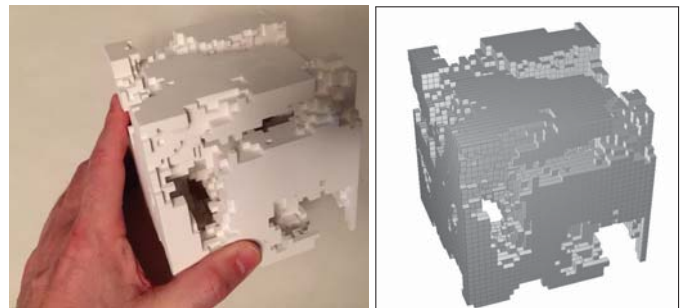


Figure 1: Kawasaki Model rendered in 3D right and photograph of 3d-printed solid model using colour inkjet powder bed fabrication (left).

recent years [3] and as well as the industrially-priced devices, there are also a range of desktop and consumer priced 3D printers available [8]. It is therefore now much more feasible for researchers to aspire to making representations of their models. We have used both low priced filament extrusion 3D printers as well as more sophisticated devices such as a layered powder bed printer [27]. We describe the relative advantages of both, particularly in the context of detailed spatial structures as can be seen in models such as that shown in Figure 1.

The Kawasaki model is essentially a lattice gas model [30] and is typically studied on a lattice of cells, where each cell is occupied by a different species of atom in for example an alloy [5]. The model is quenched from an initially random pattern and complex striated spatial structures grow during the thermal annealing of the model, following what is known as spinodal decomposition [2, 4, 12, 25].

Although the Kawasaki model “agents” are very simple ones, and their microscopic behaviour of diffusing around the system is largely governed by thermodynamics, this class of model is a good foundation for more sophisticated agent-based models. We experimented with visualising multiple species using 3D printed artifacts as a first step. Colour or embossing of the individual cells in the 3D printed artifact offers potential for making constructs that show off various microscopic properties of such models.

The key area of interest to us for making 3D printed artifacts of

models is being able to gain insights into the clusters, components [16] and spatial structures, and for 3D models this primarily involves seeing inside the complex 3D shapes and structures that arise from the models.

In addition to colouring the model cells therefore, we have experimented with omitting some species and leaving them represented as vacancies - either real vacancies or just a visual representation of a particular species. The photographs of model artifacts we have generated show how this approach lets one see inside models that one can hold in one's hand, and examine at length in a way that is still difficult to do with graphical renderings or even with virtual reality technologies.

Computer aided design software packages [32] are also widely available both as proprietary packages but as open community software packages that can help generate designs for 3D printing. However, our own use of 3D printers is with models that are not generated by design packages, but which are generated by our own software, semi-automatically from our data from our simulations [10].

The key challenge in printing complex simulation models with various physical length scales present is understanding the geometry and physicality of what is possible, given the way that 3D printers work, building up material gradually in sliced layers - and which must be supported and stable during the manufacturing process. We discuss temporary support structural issues [26] and the contributions that different 3D printer approaches can offer to this problem.

We also explore various ways of cutting away parts of the model itself to enable seeing inside the 3D physically printed structures and describe the algorithms and software we have developed to support this, taking data from the lattice cells generated from the simulation code, through to a realisable 3D printable artifact. Missing parts of the model can exacerbate the support structure problem however, and we analyse the tradeoffs that result from the different approaches.

Figure 1 shows a comparison between a Kawasaki model configuration grown in our simulation system, and rendered using 3D graphics (above) and photographed as a 3d-printed solid using inkjet powder bed fabrication (below). The figure shows the key challenge in "seeing inside" the complex emergent structure.

Our article is structured as follows: In Section 2 we give some background to models such as the Kawasaki exchange model, the Potts model, and the way we use these as representative of more complex Agent-Based models. We give some technical background on 3D printers in Section 3 and in particular describe the filament extrusion and powder bed technologies we used for the work reported in this present article.

We describe our techniques for making appropriately formatted information for driving the printers in Section 4 and present some selected photographic results in Section 5 where we also discuss the relative advantages and disadvantages of the various 3D print technologies and model cut-away approaches. We offer some conclusions and areas for further research and development in Section 6.

2 Lattice Agent-Based Growth Models

Agent-Based Models of interest to us are often simulated on a mesh or lattice, with each cell occupied by a particular agent or species of agent. Generally agents interact with the spatially localised neighbours and the Kawasaki model and its variants provide a good starting point for agent-based models involving spatial movement or rearrangement. Similar models include the Potts variant of Kawasaki (involving an arbitrary number of different agent species, and other systems including: cellular automata like the two state Game of Life model [9], or three-state variant such as the Game of Death [17]. Similar models include simulations of: health systems including disease propagation [14]; predator-prey systems [18]; social segregation systems [13]; materials propagation in gas and oil wells [11].

The Kawasaki model has been described in depth elsewhere, but for completeness we give a brief summary of its properties for 3D print-ability. We consider a spatial (cubic) mesh of length L so that the $N = L^3$ sites are all occupied by one of Q different states of agent. The simple Kawasaki model has $Q = 2$ and we typically represent these two states as material or vacancy in the 3D printed artifacts.

The model is initialised with a random mixture of the agent species, and since the diffusion model has not a direct physical energy equation to drive it, a stochastic model [29] is used to provide an effective dynamics scheme that drives the model through a quench to a finite temperature followed by an annealing process at that temperature. In practice, the algorithm involves:

- pick a random neighbouring pair of cells;
- compute energy consequences of swapping them;
- Boltzmann probability determines probability of the swap;
- repeat, above.

The pair-wise site swaps emulate an atomic diffusion process very effectively and drives the model from an initial random mixture to a phase separated structure where like species have congregated together [22].

In this present article, we do not study wide temperature variations in this present paper by simple fix on models that have been quenched to half of the characteristic critical temperature of the Kawasaki model. This means we obtain steady growth and large scale complex spinodal structures that are challenging to visualise and understand.

3 3D Print Technologies

There are a range of different 3D print technologies now available including filament extrusion; resin photopolymerisation; powder-based binder jetting; material jetting; and laser sintering. We focused on just two for the work reported in this present paper - filament based extrusion and powder bed ink binder jetting.

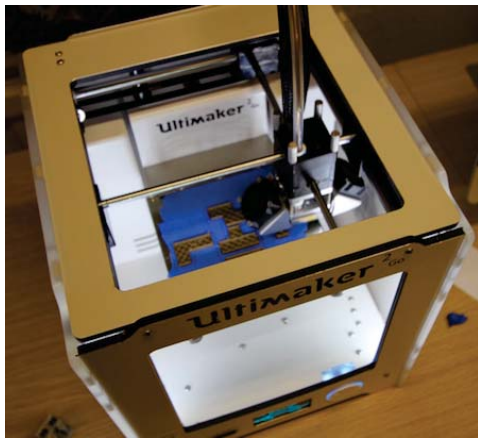


Figure 2: Ultimaker Go Filament-based portable printer showing the horizontal axes of extrusion head movement and the raise-able blue build plate.

The Ultimaker shown in Figure 2 is one of many widely available filament extrusion based 3D-printers. This heats and deposits melted filament that is most commonly made from either the starch based polylactic acid (PLA) or acrylonitrile butadiene styrene (ABS) plastic. It layers melted filament into a heated glass build plate upon which the model is formed and fans on either side of the heated print heads cool and set the filament. Filament printers are good for prototyping work and they are cheap to run but the materials used are relatively brittle and in the case of PLA, due to the starch based nature of the material, over time it can decompose in the air [28]. The Ultimaker devices we used can only print one colour or sort of material at a time. While they can make cheap prototypes of the structure, it is difficult to introduce any realisation of multiple species of our models.

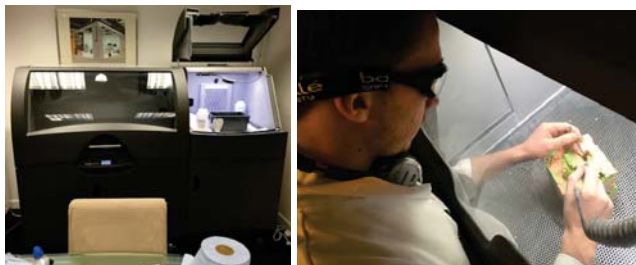


Figure 3: 3D Systems ProJet 660 Powder Bed Printer; Removing excess powder from printed Kawasaki model cube.

The 3D Systems ProJet 660 seen in Figure 3 uses a gypsum based powder with polymer binding agent as a printing medium, and it spreads a thin layer of this powder across its print bed and then binds it with jetted ink. Layers are built up as it lowers the print bed and repeats. When the print is complete the printer then heats up the entire print area drying out the powder and the printed model, which remains quite structurally weak and which can easily crumble without cautious handling. The model is transferred to the cleaning section where excess powder is blown off of the model (Figure 3 - right), and which is

subsequently submerged in a cyan-acrylate based setting substance. This reacts exothermically and cures the polymer binding agent producing a robust model that can be readily handled safely. The printer uses standard print heads and due to the setting of the powder being done by a fluid the device is able to print models in high resolution and full colour.

All of the 3D-print platforms require data to be fed to them in a particular format, and we discuss this as part of the process of practical model generation.

4 Cellular Model Data Generation

A crucial aspect of facilitating the 3D model artifacts is to interface the simulation code and its data formats with the STL format files used to drive the 3d-printers [20].

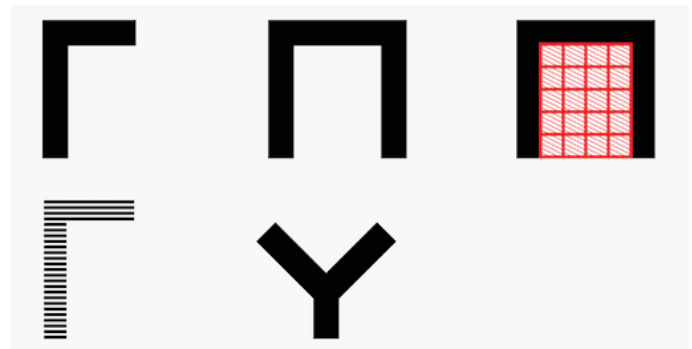


Figure 4: Geometric Support Issues: cantilever or hollow shapes require physical support during print material deposition although temporary support material (shown as red hatched) could be added. The structure is actually being made with horizontal layers or material, and 45 degree slopes are generally feasible, but not appropriate for our models.

Figure 4 illustrates the geometric support issues that are key to being able to feasibly manufacture a particular structure using the various 3D print technologies.

The interface language specifications [23, 33] for modern 3D-printers have evolved over several decades, the most common format is the STereoLithography or Spatial Tessellation Language or STL [6]. This was developed from commercial formats [1] but have similar properties to 3D computer aided design files and with the widespread exchange of graphical object files for games characters and other digital assets, these formats have converged in recent years to a relatively open *de facto* standard [7]. STL files are used by 3d-printers [7] such as the Ultimaker and Form1 to describe the models being loaded on to them. The printer itself generally cannot read STL files directly, and instead the file is first loaded into a slicer program such as Cura. This software will translate the file into something that the printer can read and use telling it how to move the print head this format depends on the brand of printer, the most common is *gcode*. We describe our model data formats and software to convert them to STL in [20].

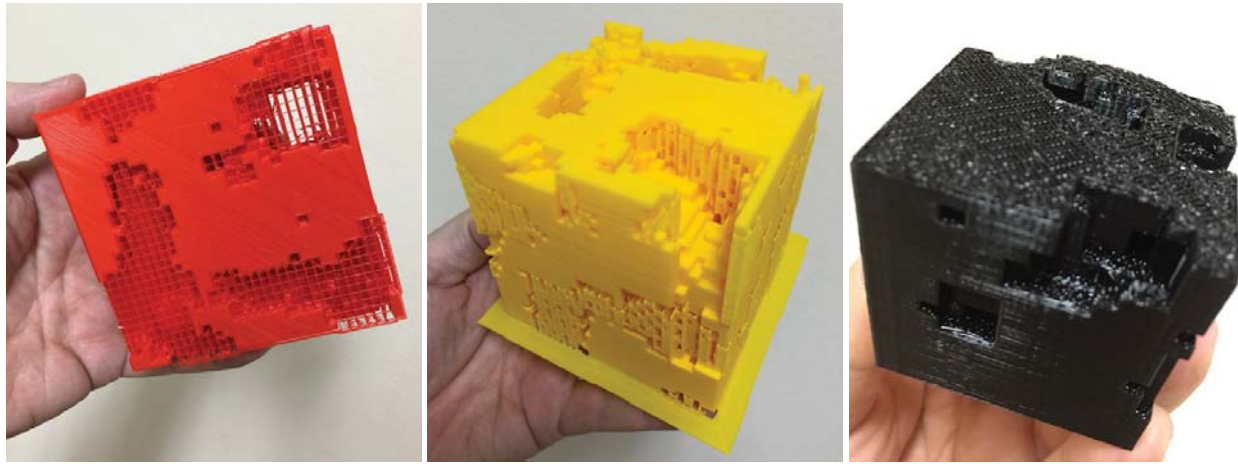


Figure 5: Cube constructs using mono-colour PLA filament and with support structure partially cut away.

5 Selected Results

The first of the Kawasaki models that we attempted to print was that of a simple two state Kawasaki system. The first attempt was done on an “Ultimaker 2 Extended” model and this can be seen in Figure 5, the only way that the print could handle printing the model as with the addition of considerable PLA support structure. This caused problems as it was very difficult to remove all of this structure due to the complexity of the model and the cavernous nature of it, it would not be possible to get into the model to remove this support structure with any practical tools.

Figure 6 shows our attempts at making hollowed out boxes. This uses less material, and is quite rapid on the filament printer - although it can only show one colour at present. We are investigating using an embossing of physical texturing approach to make bas relief surfaces that convey the different species on a filament print. The powder bed manufactured hollow boxes are feasible, although we had to experiment with different box thicknesses in terms of ABM cells, to avoid the physical box structure being too fragile. This approach of making two half-boxes should be useful for making a sequence of solid cube models for illustrating the time evolution, but is not useful for our central goal of seeing inside the physical 3D structures.

To circumvent the support structure problems, we used the powder based 3D printer (ProJet 660) because of the nature of the supporting structure being that of unbounded gypsum based powder it means the removal of the support structure was very easy only requiring the use of an air gun even in the complicated cavernous like internal structure of the Kawasaki model. It should also be noted that when the supporting structure was removed unlike with the filament models, caused no damage to the look of the model. An example of the Kawasaki model print on the ProJet 660 is shown in Figure 7 and as can be seen, the model has numerous overhangs and a highly complicated internal structure - which is of course the central challenge for visualisation and physical realisation.

The filament style printers were only capable of representing bi-state models with the 2 state being filled or empty. This meant

that model with the need to represent more data states would not be possible on these particular printers. But the project 660 with its capability to print in full colour with the use of standard commercially available print heads could realise much more complex models and an example of this can be seen in Figure 8 which shows a three state Kawasaki model with the state being that of green, red, and empty. This allows for a much wider range of these complex physical structures to be realised and with the addition of its very easy to remove support structure it even allows for very minimal damage to the model itself.

One highly valuable approach with 3D visualised agent-based models is the ability to look inside the model at the internal structure, due to the opaque nature of the model being rendered this requires the application of slicing to octant removal to allow for these otherwise unseen aspects of the model to be brought to the light of day and be analysed effectively, giving a much more complete understanding of the model itself. This means that in order to avoid the obfuscation of such aspects of the simulation model when realising the model into a 3D print it is important to be able to print off these slices. An example of this can be seen in Figure 9 which shows both of these figures showing the same 4 state Kawasaki model sliced in different ways allowing for the interaction that has occurred inside of the model to be seen.

Figure 10 shows another approach we developed for seeing inside the structure. We remove a whole octant of the cube, so that a cross sectional slide can be inspected, with exposure all the way to the core centre of the simulated model system. This also shows two-dimensional cross-sections through the model, which can of course also be generated as free standing slices.

One solution to the excess supporting structure that we thought of was to slice the model into one cube thick sheets that would allow for them to be printed off of the printing with and overhangs at all and an example of this type of sheet can be seen in Figure 11 (left). When doing this we encountered a problem when the sheet grew to a considerable size, that being that as that model would print the corners of it would become detached from the heated glassed build plate and would begin to curl upwards, meaning that they would not be able to be attached to one

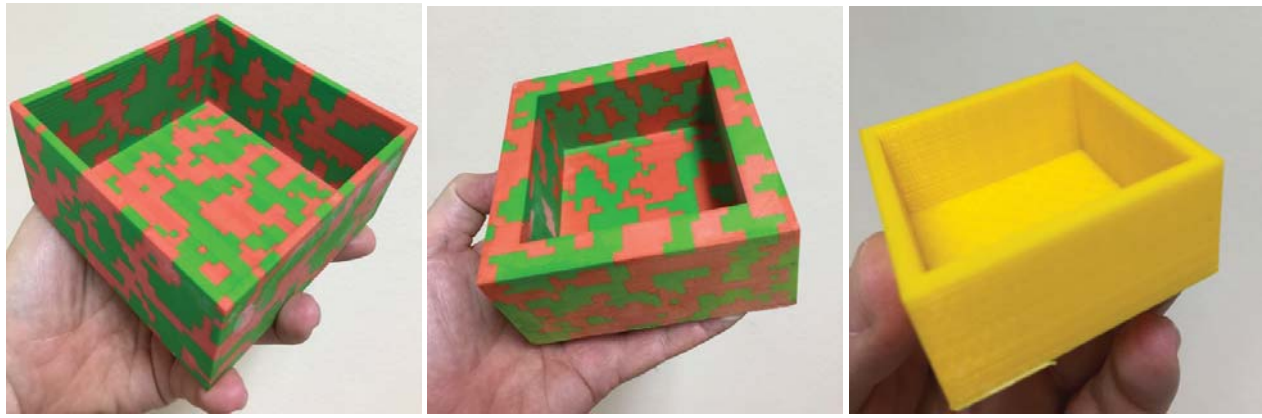


Figure 6: Various hollowed out half box cube constructions of single or four cells in thickness. From left: using powder 1 cell thick and 4 thick and using yellow PLA, 1 thick.

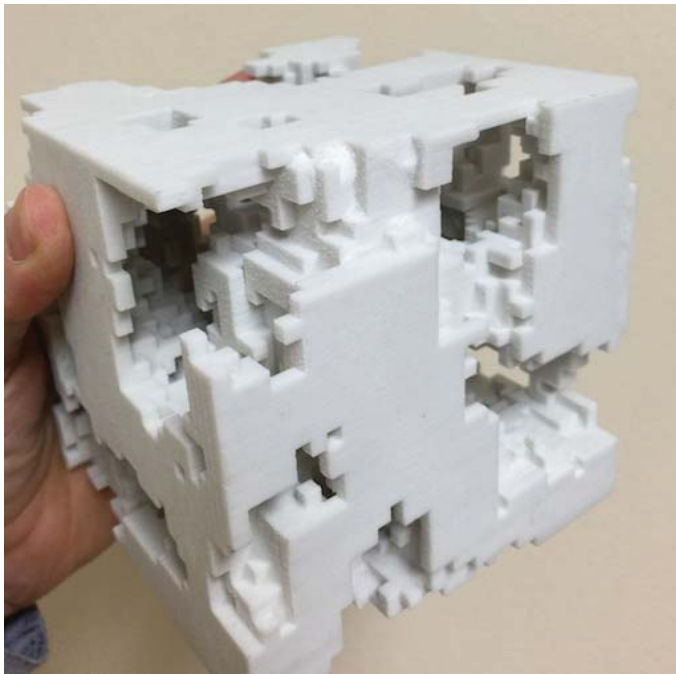


Figure 7: white-q2-sand

another once printed. Figure 11 (right) shows some free standing slices through the model, with just 4 layers of cells rendered. Note that in this particular print, no ink was used to colour the normally hidden faces, so white gypsum powder is exposed on one side.

6 Conclusion

We have described how we manufactured physical manifestations of the Kawasaki diffusion model using various 3D printing technologies and approaches to see inside the complex emergent structure of the model. Filament based extrusion printers are cheap and adequate for structural prototyping but at the time of writing can not yet effectively print in different colours with in

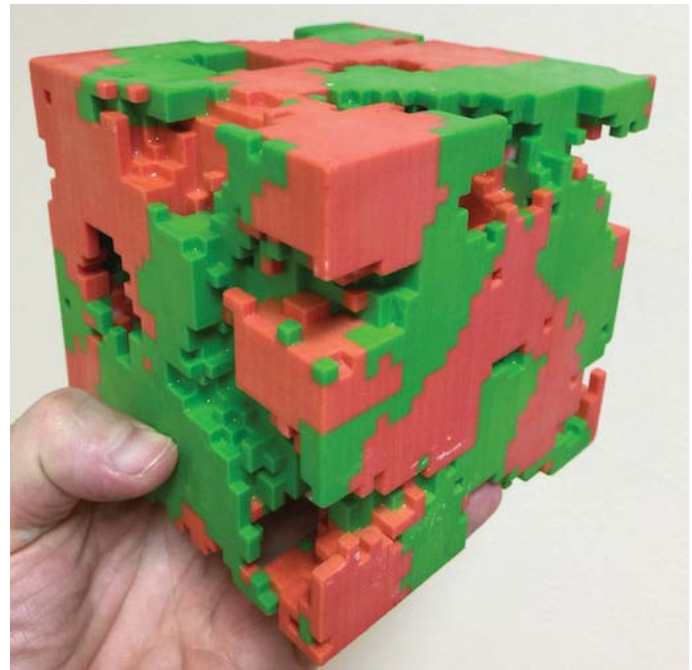


Figure 8: 3-Species Kawasaki model on 32^3 lattice with vacancy species removed as cavities allowing see-through into the internal structure.

a complex print model, and tend to require the addition of considerable material support structures to make a complex model into a feasible print.

Removing excess support material post-print is a difficult task and typically leads to damage of the near fractal physical structure of some models. Powder-bed 3D print technology with self support from the powder itself is a more feasible approach for complex models with the excess dry powder being relatively simple to blow out of the model's complex interior. There are still limitations on the relative weight the uncured model can self support when removing excess powder.

We have shown how using vacancies, hollowed cores; diagonal

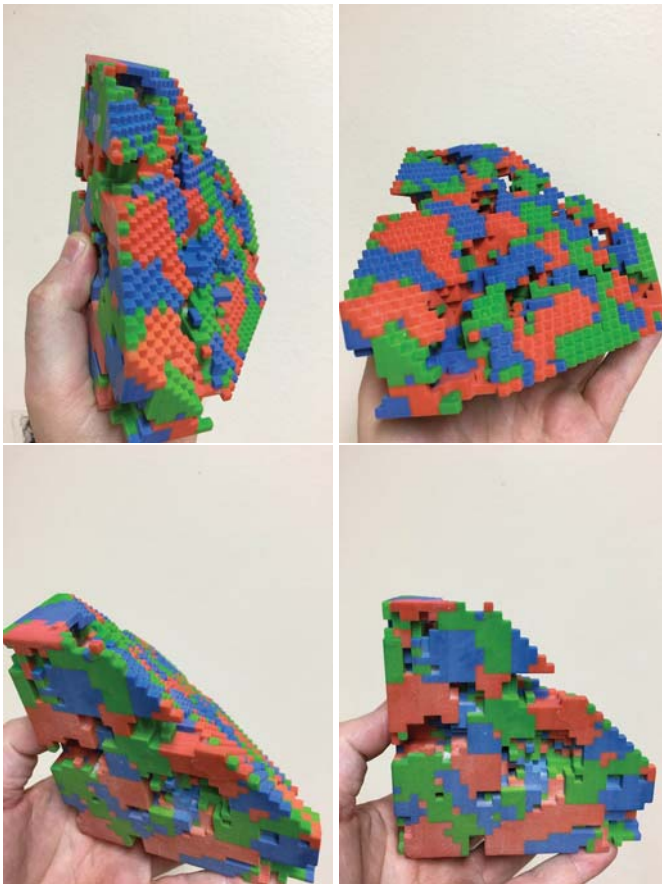


Figure 9: Diagonal Cut-away Views of 4-state model showing interior structure of spinodal interfaces.



Figure 10: Octant Cut-away View of 4 state model with unprinted vacancies used for the fourth species.



Figure 11: Sheet approach - manufactured by filament extrusion and 1 model layer thick (left) and with powder bed and 4 layer thickness (right).

and octant cut-aways and sheet slicing all give different insights into the interior structure of these 3D models, as well as reducing the amount of physical material used. . We have used simple but inefficient construction techniques that do not economise on the printed materials used for our simulation model artifacts. This is adequate for one-off print runs of scientific models, but there are a number of techniques to make better use of the printer capacity. Managing the cost effectiveness and practical logistics of manufacturing these sort of models is an important area for further research.

The colour capability of the powder bed ink jetted printer opens up considerable possibilities for multi state models with different species. We found that the powder printed models with internal vacancies used for one of the model species was an adequate approach, giving good insights into the complex interior structure of the spinodals formed by the Kawasaki model under annealing.

We believe this approach could be usefully deployed for other and more complex agent-based simulation models. Identifying isolated cluster components of agents and removing them or printing them separately is likely to be useful however to simply the print process and avoid damage to partially cured models.

There is scope for investigation of other 3D manufacturing techniques for making physical artifact realisations. Laser cutting of layered material such as cardboard or wood and plastic and resin printing using photo-polymerisation techniques may also lend themselves well to making models of this sort.

In summary, 3D printing offers a different experience to graphical rendering of complex agent based models. 3D printed artifacts are enduring, can be handled, and viewed from different angles, as well as offering a much more tactile experience, perhaps giving different insights into complex structure formation. We envisage various educational and model demonstration uses for the printed cluster or artifacts from such simulation models. We believe 3d-printing complements 3D graphical rendering for such model visualisation, and that commodity priced 3d-printers will have a significant role to play in analysing such models of complex systems.

References

- [1] 3D Systems Inc.: Stereolithography Interface Specification (STL) (October 1989)
- [2] Ball, R., Essery, R.: Spinodal decomposition and pattern formation near surfaces. *J. Phys. Condensed Matter* 2, 10303–10320 (1990)
- [3] Barnatt, C.: 3D Printing: The Next Industrial Revolution. ExplainingTheFuture.com (2013)
- [4] Binder, K.: Mechanisms for the Dynamics of Phase Transformations. In: Lovesey, S.W., Scherm, R. (eds.) *Condensed Matter Research using Neutrons Today and Tomorrow*. pp. 1–38. NATO ASI, Plenum Press (1984)
- [5] Brown, J.E., Smith, G.D.W.: Atom probe studies of spinodal processes in duplex stainless steels and single- and dual-phase Fe-Cr-Ni alloys (Oct 1990), *proc. 37 Int. Field Emission Symp.* Albuquerque, 1990, to appear in *Surf. Sci.* 1991
- [6] Burns, M.: *Automated Fabrication: Improving Productivity in Manufacturing*. Prentice Hall, Ennex Fabrication Technologies (1993), ISBN:0-13-119462-3
- [7] Fabbers: Historical resource on 3d printing - stl (October 1989), http://www.fabbers.com/tech/STL_Format
- [8] Frauenfeld, M.: Make: Ultimate guide to 3d printing. *Magazine Special Issue* (2014), makezine.com
- [9] Gilman, A., Hawick, K.: Field programmable gate arrays for computational acceleration of lattice-oriented simulation models. In: *Proc. Int. Conf. on Computer Design (CDES'12)*. pp. 91–97. CSREA, Las Vegas, USA (16-19 July 2012)
- [10] Hawick, K.A.: 3d visualisation of simulation model voxel hyperbricks and the cubes program. *Tech. Rep. CSTN-082*, Computer Science, Massey University, Albany, North Shore 102-904, Auckland, New Zealand (October 2010), <http://www.massey.ac.nz/~kahawick/cstn/082/cstn-082.pdf>
- [11] Hawick, K.A.: Gravitational and barrier effects in d-dimensional invasion percolation reservoir models. In: *Proc. Int. Conf. on Power and Energy Systems and Applications (PESA 2011)*. pp. 259–266. No. CSTN-134, IASTED, Pittsburgh, USA (7-9 November 2011)
- [12] Hawick, K.A.: Analysing spinodal decomposition using image morphology with thinning, edge detection and graph methods. In: *Proc. IASTED International Conference on Signal and Image Processing (SIP 2013)*. pp. 804–840. No. CSTN-176, IASTED, Banff, Canada (17-19 July 2013)
- [13] Hawick, K.A.: Multiple species phase transitions in agent-based simulations of the schelling segregation model. *CSI 0001*, Department of Computer Science, University of Hull, Robert Blackburn Building, Cottingham Road, Hull, UK (December 2013), <http://www.hull.ac.uk/php/466990/csi/reports/0001/csi-0001.html>
- [14] Hawick, K.A.: Role of connectivity and clusters in spatial cyclic sirs epidemic dynamics. In: *Proc. IASTED International Conference on Health Informatics*. pp. 1–8. IASTED, Gabarone, Botswana (1-3 September 2014), <http://www.hull.ac.uk/php/466990/csi/reports/0003/csi-0003.html>
- [15] Hawick, K.A., Husselmann, A.V.: Photo-penetration depth growth dependence in an agent-based photobioreactor model. In: *Proc. 14th International Conference on Bioinformatics and Computational Biology (BIOCOMP'13)*. p. BIC4051. No. CSTN-204, WorldComp, Las Vegas, USA (22-25 July 2013)
- [16] Hawick, K.A., Leist, A., Playne, D.P.: Parallel Graph Component Labelling with GPUs and CUDA. *Parallel Computing* 36(12), 655–678 (December 2010), www.elsevier.com/locate/parco
- [17] Hawick, K.A., Scogings, C.J.: Cycles, transients, and complexity in the game of death spatial automaton. In: *Proc. International Conference on Scientific Computing (CSC'11)*. pp. 241–247. No. CSC4040, CSREA, Las Vegas, USA (18-21 July 2011)
- [18] Hawick, K.A., Scogings, C.J., James, H.A.: Defensive spiral emergence in a predator-prey model. *Complexity International* 12(msid37), 1–10 (October 2008), <http://www.complexity.org.au/ci/vol12/msid37>, ISSN 1320-0682
- [19] Hawick, K.A., Stockwell, L.A.D., Odiam, L.R.F.: A review of 3d printer technologies. *Tech. Rep. CSI-0030*, Computer Science, University of Hull, Cottingham Road, Hull, UK (November 2015)
- [20] Hawick, K.A., Stockwell, L.A.D., Odiam, L.R.F.: 3d print technology for cellular agent-based growth models. In: *Proc. Int. Conf. on Modelling Identification and Control*. pp. 1–8. No. 830-037, IASTED, ACTA press, Innsbruck, Austria (15-16 February 2016)
- [21] Hawick, K.: Visualising multi-phase lattice gas fluid layering simulations. In: *Proc. International Conference on Modeling, Simulation and Visualization Methods (MSV'11)*. pp. 3–9. CSREA, Las Vegas, USA (18-21 July 2011)
- [22] Hawick, K.A.: *Domain Growth in Alloys*. Ph.D. thesis, Edinburgh University (1991)
- [23] Kai, C.C., Jacob, G.G.K., Mei, T.: Interface between cad and rapid prototyping systems. part 2" lmi an improved interface. *Int. J. Adv. Manuf. Technol.* 13, 571–576 (1997)
- [24] Kawasaki, K.: Diffusion constants near the critical point for time dependent Ising model I. *Phys. Rev.* 145(1), 224–230 (1966)
- [25] Lebowitz, J.L., Marro, J., Kalos, M.H.: Kinetics of Phase Segregation: A Review of Some Recent Results. *Comments Solid State Physics* 10(6), 201–217 (1983)
- [26] Lee, M., Dunn, J.C.Y., Wu, B.M.: Scaffold fabrication by indirect three-dimensional printing. *Biomaterials* (2005)
- [27] Liu, H., Maekawa, T., Patrikalakis, N.M., Sachs, E.M., Cho, W.: Methods for feature-based design of heterogeneous solids. *Computer-Aided Design* 36, 1141–1159 (2004)
- [28] Lu, D.R., Xiao, C.M., Xu, S.J.: Starch-based completely biodegradable polymer materials. *Express Polymer Letters* 3(6), 366–375 (2009)
- [29] Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equation of state calculations by fast computing machines. *J. Chem. Phys.* 21(6), 1087–1092 (Jun 1953)
- [30] Penrose, O., Buhagiar, A.: Kinetics of nucleation in a lattice gas model: Microscopic theory and simulation compared. *J. Stat. Phys.* 30(1), 219–241 (1983)
- [31] Potts, R.B.: Some generalised order-disorder transformations. *Proc. Roy. Soc.* pp. 106–109 (1951), received July
- [32] Ritland, M.: *3D Printing with Sketchup*. Packt Publishing (May 2014), ISBN 978-1-78328-457-3
- [33] Rock, S.J., Wozny, M.J.: A flexible file format for solid freeform fabrication. In: *Proc. Solid Freeform Fabrication Symposium*. pp. 1–12. Austin, Texas, UAS (12-14 August 1991)
- [34] Sangani, K.: Make it to fake it. *Engineering and Technology* 8(9), 38–41 (October 2013)
- [35] Thompson, D.W.: *On Growth and Form*. Cambridge University Press (1942)

Comprehensive 3D Visualization of Simulated Processes in Virtual Factories

S. Masik¹, T. Schulze², M. Raab¹, and M. Lemessi³

¹ Fraunhofer Institute for Factory Operation and Automation IFF, Magdeburg, Germany

² School of Computer Science, Otto von Guericke University, Magdeburg, Germany

³ Mannheim Regional Center, John Deere GmbH & Co. KG, Mannheim, Germany

Abstract - Industrial virtual reality (IVR) facilitates the development and operation of factories and plants by enhancing communication among all the stakeholders involved and providing a visualization combining the different planning data with simulation data generated during development or real-time data acquired during operation. The major benefit derived from combining these heterogeneous data sources is the capability to evaluate and review a complete system's performance and consistency. Experience gained in numerous industry projects indicates that the acceptance and practicability of this technology is largely contingent on the labor required to procure, preprocess and synchronize data as well as the maintainability of the IVR models. This paper describes the workflows and methodologies used to integrate industrial data in interactive IVR semi-automatically to plan, review and operate factory systems and to train executives, technicians and workers.

Keywords: virtual reality, assembly simulation, Industry 4.0, digital factory

1 Introduction

Visualization is an indispensable tool for the validation of simulation models and the presentation of simulation results [1]. Simulation models of assembly operations based on manufacturing specifications frequently have a high level of detail (LOD) in response to the increased variety of models and complexity of assembly systems. Currently widespread and often schematic 2D visualizations are no longer able to meet these heightened demands.

One option is to use 3D visualizations based on industrial virtual reality (IVR) incorporating existing 3D assembly layouts, 3D product models, and process flows generated from simulation models. Rather than simulating assembly in detail for ease of product assembly, the goal here is to visualize the simulations at the stations in conjunction with a realistic representation of the stations, the stage of assembly of products including the components and tools to be provided, and assemblers' changes of location. This form of visualization permits a good evaluation of the simulated sequences, which goes beyond

established standard 2D visualizations. 3D visualizations of assembly operations can be generated automatically by integrating the generated simulation data with the existing 3D data of layouts and products.

This paper sets forth the benefits of 3D visualizations of simulated assembly operations and presents a methodology for generating 3D visualizations of simulated systems and operations. It includes a discussion of the requirements imposed on the data provided, particularly interoperability. Prototype systems implemented at a manufacturer's facilities are presented. It concludes with a look at future development studies.

2 Industrial Virtual Reality (IVR)

Dynamic visualization of simulated operations is a typical format for representing discrete simulations. Usually, the representation changes over time. Given its benefits, many commercial simulation software systems include dynamic visualization of results. 2D layouts were still used for presentation in the early days of development. Once 3D geometric models of simulated systems became available, simulation results were also visualized in so-called 4D systems, the fourth dimension being the component of time.

Virtual reality is the representation and simultaneous perception of interactive virtual environments. The artificial environments are generated by computers in real time while allowing for physical properties. Virtual reality systems are used in many domains, e.g. flight simulation, building design and urban planning. Virtual realities generally require a higher level of immersion, i.e. the boundaries between the real and the virtual are blurred in the user's perception (imagination). In particular, self-contained stereoscopic projection systems or head mounted displays produce such a state.

Immersion only plays a secondary role in industrial virtual reality (IVR) applications. Instead, the focus is on interactivity and real-time compatibility, interoperability and dynamics of the integrated data, and the realistic representation of the objects and their properties, which are invisible in reality.

Virtual factories are industrial collaborative environments intended to represent a factory in virtual reality [2]. The underlying IVR model links existing partial models or model components of virtual factories [3], each of which reproduces parts of industrial reality. Among others, these include the factory, product, process and simulation models. The use of IVR is intended to increase the knowledge users acquire from interactive analysis of interconnected model components and their changes over time and from additional information generated by interacting model components.

Interactive, functional 3D visualizations expedite industrial planning significantly. Delivering universally understandable representations of complex issues, IVR is used in virtual manufacturing as a tool for interdepartmental collaboration and communication, e.g. in industrial product development and process engineering [4]. This entails linking heterogeneous data sets from different planning and simulation systems and integrating them in a single IVR model.

At present, IVR models are frequently not used until the final third of industrial planning since required raw data is unavailable in a suitable form beforehand, interfaces are inadequate or generation and updating are too time-consuming. IVR models can be used after planning, e.g. for marketing, documentation or training, and can additionally be used and upgraded during operative planning or later replanning. Moreover, IVR models combined with simulation models can be turned into cyber-physical systems.

The acceptance of IVR models is heavily linked to the time and money required to acquire data and to generate, maintain and synchronize models, thus making it essential to use automatic or semi-automatic methods of generation and, generally, to revert to existing data sets and to integrate them in maintenance as well [5].

3 Related Work

This paper concentrates on two major aspects of IVR models, specifically efficient modeling and use cases. Efficient modeling requires being able to link heterogeneous model data automatically. Every component of an IVR model is an abstraction of a real or hypothetical system and has its own perspective within the modeled system. When IVR models are created, data from every component included has to be integrated. Every model component has its own proprietary specification. Standard exchange formats can be used to link or render any component interoperable. Integration is frequently easier when all of the data come from system software components, e.g. Siemens PLM, but this is not an option for IVR models that consist of diverse components. Standard open or proprietary interfaces have to be used instead. Reliable standards for model components from CAD systems are already state-of-the-art. The situation is more difficult for simulation components.

The simulation community addressed this problem by developing Core Manufacturing Simulation Data (CMSD), a data interface standardized for simulation applications and other software systems [6]. The Simulation Data Exchange (SDX) interface has the same goal.

Szalay [7] also describes the need to visualize simulation results, which support decision making or verify repudiate hypotheses, quickly and easily. Eilers et al. [8] determine that, when extending the functions of IVR models, it is expedient and possible for visualizations of simulated systems to represent data and simulation results in real time as well as information that would be easily visible and evaluable in the real system.

Dynamic IVR models are used in industry as, among others, collaborative planning support systems in different stages of the factory life cycle [4], as virtual laboratories [7] and as platforms for decision making [9] and training [10].

4 Advantages of IVR Models for Assembly Simulation

An assembly simulation model generally consists of the objects of station, work-in-process, worker, assembled product, required components and tools, and the times to complete assembly. Typical results of such simulation models are throughput per unit of time and allocated times at the individual stations.

2D visualization is supported by simulation systems employed in factory simulation such as Plant Simulation, Simio and FlexSim. 2D visualizations, which have been being used effectively for years to display material flows and to detect bottlenecks, are very well suited for early stages of factory planning.

In later stages of planning with higher levels of detail and availability of greater quantities of data, shortcomings of simulation models and 2D visualization manifest themselves in the domain of assembly in particular. For instance, workers' changes in location to and from racks storing required components may not be represented in the visualization.

IVR models of simulated assembly workflows are used in factory planning for extended validation of integrated simulation models, scalable presentation, interdepartmental process improvement, and evaluation of aspects of planning, something which could otherwise only be done by completely linking and dynamizing the different static base models.

4.1 Extended Validation of Simulation Models

Necessary operations including the technological sequence at a station are assigned in the simulation model to the product being assembled according to the process specification. These operations have to be executed by the worker and, what is more, the respective conditions for execution, e.g. availability of a component, a module or a

tool, are known. These operations are frequently aggregated into one work package in assembly simulation models and the individual conditions for the execution of the operations are combined into one condition for the execution of the work package. This simplification is permissible in order to achieve the classic objectives of assembly simulation and shortens the execution time of the simulation model.

Systematically controlled elimination of such aggregation and the transfer of the specific operations to an IVR model significantly improve the representation of the stage of assembly based on operations. This visualization of operations including changes made on the assembled product facilitates extended validation of simulated assembly procedures, particularly when a wide range of models is assembled. Correct mapping of operations in the simulation model is crucial to the correctness and acceptance of the results.

4.2 Presentation and Process Improvement

3D visualization with IVR models is very important for presenting the results of planning [11]. The views generated by IVR not only provide a “real” image of the planned (virtual) environment but also allow the introduction of additional information that cannot be found in a “real” environment. This may include dashboards and scoreboards that display throughputs, tags on assembly objects with object information, and color coding of overloaded units. A collective analysis of the virtual environment by several parties is conducive to identifying process improvements. Modifications of the virtual layout are made interactively and incorporated quickly once the simulation has been restarted.

Depending on their complexity and the visualization methods used, IVR models impose medium to very high requirements on the hardware used for visualization. Care must be taken with the complexity and rendering technology to ensure that the IVR models generated for visualization are scalable and thus usable on lower performance computer systems or mobile systems with limited resources.

The scalability of the visualization system used is an important foundation of IVR. On the one hand, 3D visualizations can be displayed in special display facilities equipped with efficient, usually distributed visualization and simulation hardware, e.g. a CAVE, a Powerwall or a 360° projection system such as the Elbe Dom in Magdeburg [12].

On the other hand, presentations may be accessible on site, e.g. in a real factory. Tablet computers can be used, which, depending on their performance, either generate images directly or stream them from a server by remote rendering. The advantage of onsite analysis outweighs such hardware’s incapability to create deep immersion.

4.3 Evaluation by Linking Different Data Models

IVR models use dynamic pathfinding to determine, evaluate and compare the distances covered by workers during a simulated assembly operation with the specifications from process planning. Different product series and models require different operations and thus also different paths taken by workers during assembly. Information on the location of racks, required components and assembled objects is used to determine path distances. Physiological indicators of work are derived from these cumulative values. The path analysis in Figure 1 reveals the benefits of linking different data models in an IVR model.

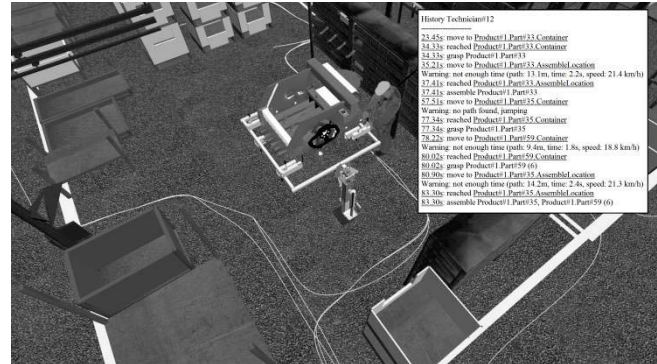


Figure 1: Path analysis in an IVR model

Integrated use of 3D geometries of manufacturing equipment and the product and their linkage with process flows facilitate this analysis of collisions between dynamic and static entities. Depending on the IVR model’s level of detail, different product models and configurations and different stages of production can be analyzed in the context of every pertinent manufacturing environment. Figure 2 shows the visualization of a collision check in an IVR model.



Figure 2: Collision check in an IVR model

5 Components of IVR Models

The components of an IVR model are the factory, the product, the process and the simulation model. Every model has its own view of the modeled production system. The factory model specifies a factory’s architecture and the systems and equipment inside. The product model reproduces the geometry of the manufactured product. Both

of these model components are static and are typically created using standard CAD systems and can be exported into IVR models in good quality by means of common exchange formats. The process model specifies the time sequence of operations and the conditions needed to manufacture a product. The simulation models generate an instance for the modeled sequences of other components. Simulation models control and affect the changes in model entities and their properties over time.

In industrial settings, the data needed to create IVR models are typically specified in different software systems. Since the software used is heterogeneous and interfaces are thus lacking, inadequate or proprietary, the integration of data from components frequently entails manual labor.

The integration and fusion of heterogeneous data necessitates linking the properties of a single physical entity over and above all data sets. Linking different data sources is complex since ubiquitous, unique identifiers frequently do not exist and appropriate mapping and data mining strategies have to be applied.

Only a subset of the potential raw data is available, depending on the state of planning of the simulated system. Nevertheless, dynamic IVR models of factory systems with maximum detail have to be generated each time. A multitude of widely different data on the product and production process is produced during the different stages of the product and factory life cycle. These planning data are typically compiled by different units and stored and managed with the aid of internal IT infrastructures.

Although the data needed for IVR models are usually available digitally in IT systems and could be used as the basis for further and combined analyses or directly for the visualization and discussion of important issues, they are frequently not used directly as input for simulation analyses or visualizations in downstream phases of planning. Typical reasons for this are problems with access, interfaces or integration that result in substantial additional manual labor to generate and maintain IVR models.

This makes it necessary throughout the factory planning process to identify the combinations of available data sets relevant to planners and to provide automated systems and workflows that generate and progressively detail IVR models. Relevant data must be extracted and linked automatically in order to be able, for instance, to project simulation results seamlessly in a factory setting and evaluate them integrally.

6 Component Integration

The Fraunhofer IFF in Magdeburg developed the framework Review3D to create heterogeneous IVR models. One goal while developing Review3D was to integrate heterogeneous components with proprietary and standard interfaces. Import functions of Review3D import data on components. The integration of individual data models is highly automated since this task is very complex and in

order to minimize or eliminate manual creation of IVR models and thus save time and money.

Combining simple geometries with simulation results is often sufficient for non-detailed animations of processes simulated in early stages of planning. Since neither the simulation model nor the geometry data contains the necessary information, the generation of detailed IVR models of manual or automated assembly operations additionally requires the data models already described. The simulation model is often not detailed enough to include the assembly of individual parts or the required tools, for instance. Conversely, the product model does not contain any information on linking part geometries and individual procedures.

Imported process specifications are used to close these information gaps and link simulation results with product and manufacturing equipment geometries. Information from the factory model additionally delivers the requisite position data on places referenced in the simulation model, e.g. stations, storage facilities and racks. Figure 3 presents an overview of links between IVR model components.

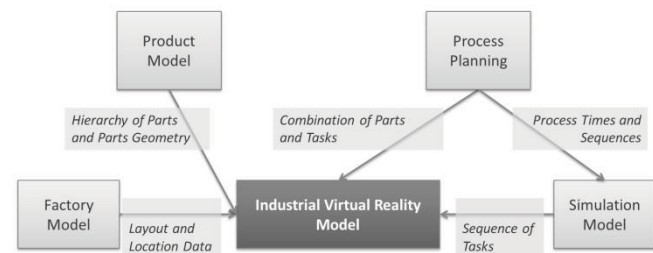


Figure 3: Simplified overview of links between IVR model components

Data needed to create a dynamic IVR model, which is still lacking after the available data sources have been imported, usually has to be generated automatically from the data on hand. These could include trajectories of dynamic entities and kinematics of virtual humans and robots as well as realistic descriptions of material or lighting.

Objects that change location in IVR models, e.g. products, manufacturing equipment and workers, move on object paths. These individual paths are combined into a static or dynamic network. Static networks are generated from available data sources such as CAX systems and simulation models or created manually. These networks do not change during simulation or visualization. Since every path in a static network is defined before visualization, such networks are not suitable for varying layouts and process specifications.

Dynamic networks employ methods of automatic pathfinding to identify paths [13]. Figure 4 presents an example of a generated navigation mesh [14] for dynamic pathfinding at an assembly station. The higher computational complexity and the real-time demands of IVR models make it essential, however, to strike a balance

between detailing and efficiency. Particularly in multi-layered environments [15], automatic pathfinding requires contextual information in addition to the classic factory layout in order to identify floors, stairwells, elevators, obstacles and the like clearly [16].

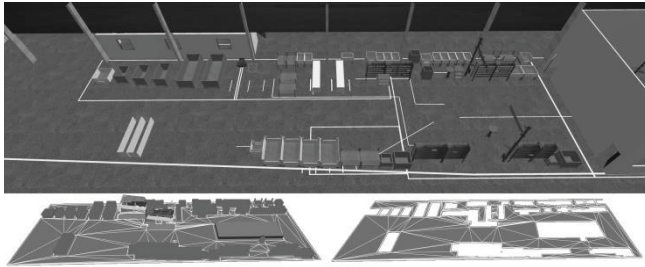


Figure 4: Navigation mesh generated for dynamic pathfinding (top: original factory layout of an assembly station, bottom left: detected obstacles, bottom right: navigation mesh)

Workers' operations can either be captured empirically with motion capturing systems or modeled with software tools. Atomic components of typical manual workflows and motion sequences are stored in libraries and combined into complex animations as needed. The stored motions can range from simple runs to complex assembly animations. Realistic animations of human models require a skeleton specified with bones, transformations and weightings at different times [17]. Skinning methods, e.g. dual quaternion blending [18], are employed to create such dynamic geometry models.

Figure 5 presents an example of an IVR model of an assembly line, which was generated automatically from raw data.



Figure 5: IVR model of an assembly line

7 Prototype Implementation

IVR models of work and production systems must be capable of integrating the vast quantities of heterogeneous data produced during planning and operation. Data on geometries, processes and simulation results are the most important. Prototype implementation of 3D visualization of heterogeneous data in IVR models is presented below with an example from the automotive industry.

7.1 Factory and Product Models

Geometry data predominantly come from CAD systems and are imported into IVR models using common exchange formats. Normally, they specify the geometry and hierarchical structure of products, work and production systems, machinery and plants, and the factory's architecture. Once the data has been imported into the IVR model, other steps are needed to generate high performance and visually representative visualizations from them.

Realism is an important goal of IVR models. Visualizations of objects created with CAD systems frequently lack any relation to their real appearance. Colors are predominantly used in CAD systems to code information on certain properties or a specific object's relationships. These artificial materials can be replaced by realistic representations by semi-automatically assigning colors, textures or enhanced shader materials from a material and resource database to reflective, anisotropic, variable or liquid objects. Specific criteria, designations or meta elements are used to categorize materials automatically [19]. Figure 6 presents a factory model before and after preprocessing.

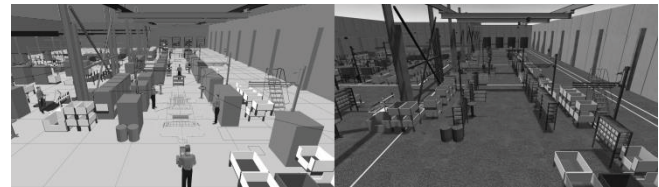


Figure 6: Factory model before and after preprocessing

Since they are generated during product development and factory planning and imported from the pertinent CAD systems, the geometries and hierarchies of products and the factory layout are normally very detailed. The conversion of native CAD geometries consisting of BREP (description of objects by their surfaces) or NURBS (non-uniform rational B-spline) definitions additionally produces very complex, polygonal geometries. Demands for interactivity and real-time compatibility make it essential to reduce geometric and hierarchical complexity [20].

Methods based on certain error metrics or complexity parameters can be used to simplify geometry [21]. Such algorithms can be applied semi-automatically by basing their parameterization on particular object information, e.g. size, location, designation, meta information and type. Additional techniques such as level of detail generation and hidden surface removal can also be applied automatically when preprocessing data.

Reducing complexity and optimizing the object hierarchy is particularly important for temporary dynamic objects since they are often instantiated several times. For instance, fifty product instances with different geometries required by the variety of models and different stages of assembly have to be represented on an assembly line simultaneously.

The quality of the illumination model displayed by IVR models is extremely important not only for realism but also as the basis for the analysis of the illumination in factory buildings and at workstations. Unlike classic methods of illumination, global illumination incorporates the interactions of rays of light with different surfaces in a virtual scene [22]. Whereas a few light sources suffice for local illumination, global illumination requires a real light setup, which can either be specified manually or imported from the planning systems automatically [23]. Figure 7 presents a detail of an IVR model with classic and global illumination.

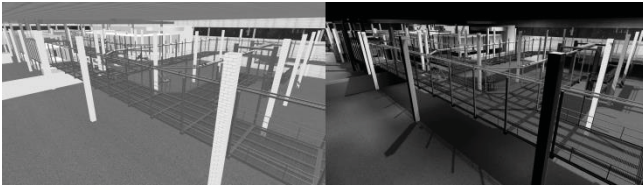


Figure 7: IVR model with classic illumination (left) and global illumination (right)

7.2 Process Models

Process models represent the process specifications from computer-aided process planning (CAPP). Among other things, these specifications include information on process times, pertinent components and modules, required tools and manufacturing equipment, workers and robots involved, associated stations or cells, and group and model relationships. They may also include written descriptions of steps of manual work. Individual operations are uniquely referenceable. The data sets needed are normally stored in relational database management systems and can be exchanged using database interfaces or database queries exported as tables.

7.3 Simulation Models

Simulation model data and results from simulation runs provide the basis for the dynamics in IVR models. When supported by both the simulation and the IVR software system, standard formats such as SDX or CMSD can be used to exchange simulation data and results, but specialized proprietary exchange formats are usually used to do this. The simulation data can come from distributed, online and offline simulations.

Discrete event simulations normally export a chronologically sorted specification of every event that occurs during the simulation run. Such a trace is an important data source for the visualization of dynamic assembly operations.

7.4 Case Study

The methods and workflows described have been tested and refined in different projects. A pilot project with Deere & Company serves to elucidate the approach. The objective of the project was to create a prototype IVR model of an existing agricultural machinery assembly line,

solely using existing (planning) data and automating as many procedures as possible. The project partner delivered raw data consisting of 3D CAD data of the factory and the product, tabulated process data and the simulation model.

The integration steps were performed automatically and required a minimum of manual revision, e.g. whenever identifiers could not be used consistently with the different systems. The workflow described has been incorporated in an assistance system that integrates the aforementioned data and can be used in situations with similar data to generate other IVR models automatically.

8 Summary and Outlook

This paper discussed the visualization of simulated assembly systems in industrial virtual reality with little labor in parallel with industrial planning. The main focus was placed on the analysis of potential data sources aside from those already covered by commercial VR applications and on the development of semi-automatic workflows and techniques that are crucial to the effortless integration of different kinds of planning data. The approaches described in this article have been tested successfully in multiple pilot scenarios for production and manufacturing applications. The lack of suitable standard exchange interfaces remains the major challenge and needs to be addressed by future research projects.

On the one hand, ongoing research studies are further refining the IVR model by integrating detailed assembly process animations from ergonomics and process modeling tools as well as the requisite tools, resources and means of conveyance.

On the other hand, there is a need to develop virtual environment display systems that deliver suitable visualizations, allow intuitive interaction with IVR models and are particularly suited for and meet the demands of collaboration, interconnectivity, computing capacity, interactivity and immersion. The Fraunhofer IFF is therefore currently developing a large-scale, multi-user virtual environment display system that consists of a cylindrical and hemispherical screen (6.5 m in height and 16 m in diameter), two dozen high resolution stereoscopic projectors, a high-performance rendering cluster and interfaces for a number of different interaction devices.

9 References

- [1] M. Schmitz, S. Wenzel. "Using 3D Visualization in the Context of Discrete-Event Simulation - Significance and Development Trends"; Proceedings of Simulation in Produktion und Logistik, 2013.
- [2] S. Jain, G. Shao. "Virtual factory revisited for manufacturing data analytics."; Proceedings of the 2014 Winter Simulation Conference, 2014.

- [3] VDI Association of German Engineers. "VDI Guideline 4499: Digital factory - Digital Factory Operation". Beuth Verlag Berlin, 2011.
- [4] N. Menck, C. Weidig, J. Aurich. "Virtual Reality as a Collaboration Tool for Factory Planning based on Scenario Technique."; Proceedings of Forty Sixth CIRP Conference on Manufacturing Systems, 2013.
- [5] VDI. Association of German Engineers. "VDI Guideline 3633 Part 11: Simulation of systems in logistics, materials handling and production - Simulation and visualization". Beuth Verlag Berlin, 2009.
- [6] SISO Simulation Interoperability Standards Organization. "SISO-STD-008-01-2012: Standard for Core Manufacturing Simulation Data – XML Representation". 2012.
- [7] A. S. Szalay. "From simulations to interactive numerical laboratories"; Proceedings of the 2014 Winter Simulation Conference, 2014.
- [8] K. Eilers, J. Rossmann. "Modelling an AGV based facility logistics system to measure and visualize performance availability in a VR environment"; Proceedings of the 2014 Winter Simulation Conference, 2014.
- [9] A. J. Collins, D. K. Ball, J. Romberger. "Simulation visualization issues for users and customers"; Proceedings of the 2014 Winter Simulation Conference, 2014.
- [10] T. Haase, N. Weisenburger, W. Termath, U. Frosch, D. Bergmann, M. Dick. "The Didactical Design of Virtual Reality Based Learning Environments for Maintenance Technicians"; Virtual, Augmented and Mixed Reality. Applications of Virtual and Augmented Reality, 27-38, Springer, 2014.
- [11] S. Choi, H. Jo, S. Boehm, S. Do Noh. "An Integrated System for One-Stop Virtual Design Review."; Proceedings of Concurrent Engineering, 2010.
- [12] W. Schoor, S. Masik, M. Hofmann, R. Mecke, G. Müller. "Elbe Dom: 360 Degree Full Immersive Laser Projection System"; Proceedings of Virtual Environments IPT-EGVE, 2007.
- [13] G. Snook. "Simplified 3D Movement and Pathfinding Using Navigation Meshes"; Game Programming Gems, Charles River Media, 2000.
- [14] P. Tozour, I. S. Austin. "Building a Near-Optimal Navigation Mesh"; AI Game Programming Wisdom, Cengage Learning, 2002.
- [15] W. Toll, F. A. Cook IV, R. Geraerts. "Navigation Meshes for Realistic Multi-Layered Environments"; Proceedings of International Conference on Intelligent Robots and Systems, 2011.
- [16] M. Fischer, H. Renken, C. Laroque, G. Schaumann, W. Dangelmaier. "Automated 3D-motion planning for ramps and stairs in intra-logistics material flow simulations"; Proceedings of the 2010 Winter Simulation Conference, 2010.
- [17] D. Terzopoulos. "Simulating Humans and Lower Animals"; Proceedings of the 2010 Motion in Games Conference, 2010.
- [18] V. Kavan, S. Collins, C. O'Sullivan. "Dual Quaternions for Rigid Transformation Blending"; Technical Report TCD-CS-2006-46, 2006.
- [19] A. Schilling, S. Kim, D. Weissmann, Z. Tang, S. Choi. "CAD-VR geometry and meta data synchronization for design review applications"; J. Zhejiang Univ.-Sci. Journal, 2006.
- [20] H. Hoppe, T. Deroose, T. Duchamp, J. McDonald. "Mesh Optimization"; Proceedings of the 20th annual conference on Computer graphics and interactive techniques SIGGRAPH, 1993.
- [21] P. Heckbert, M. Garland. "Optimal triangulation and quadric-based surface simplification"; Computational Geometry, Vol. 14 (1999), No. 1-3, 49-65, 1999.
- [22] I. Radax. "Instant Radiosity for Real-Time Global Illumination"; Institute of Computer Graphics and Algorithms, Vienna University of Technology. <https://old.cg.tuwien.ac.at/research/publications/2008/radax-2008-ir/radax-2008-ir-paper.pdf> [Accessed February 2016], 2008.
- [23] G. Papaioannou. "Real-time diffuse global illumination using radiance hints"; Proceedings of the ACM SIGGRAPH Symposium on High Performance Graphics, 2011.

A Visualization Framework to Eliminate Cluster Overlap

A. Robert Marceau¹, B. Karen Daniels², and C. Georges Grinstein²

¹Computer Science Department, Rivier University, Nashua, NH, USA

²Computer Science Department, University of Massachusetts Lowell, Lowell, MA, USA

Abstract—Overlapping clusters of data points in a multi-dimensional space may be difficult to visualize. We introduce the visualization "conflict" graph, representing overlaps between pairs of clusters in the visualization space. To clearly differentiate clusters from each other, we separate the set of clusters into a limited number of partitions, each containing no overlapping clusters. We show that, with some assumptions, minimizing the number of partitions is NP-complete. We apply vertex-coloring heuristics to partition the visualization space graph, resulting in multiple visualization views, each without overlapping clusters. Our approach is broadly applicable, allowing different methods for different visualizations (scatterplot, RadViz, parallel coordinates ...), clustering quality assessment, conflict detection and graph coloring methods. We illustrate it for clusters of real and artificial data in scatterplots, with two different clustering quality metrics, two conflict detection methods, and two graph coloring heuristics.

Keywords: Visualization, Clustering, Graph Theory

1. Introduction

1.1 Problem Statement, Motivation, Approach

Clustering multi-dimensional data may reveal significant relationships. These clusters, especially when occurring in high dimensional data, may be difficult to visualize due to overlaps. For example, clusters may become merged in a lower dimensional visualization space, leading to difficulty in distinguishing one cluster from another. Figure 1 illustrates a problematic 3-D data set containing 15 elongated clusters. Scatterplots (and variants) are one of the most common visual encoding techniques for dimensionality reduction [1]. There are numerous methods for generating a useful scatterplot from high-dimensional clustered data. However, none guarantee that clusters in the original data space will not overlap in the visualization space. This motivated our multiple view approach.

We introduce the visualization "conflict" graph, which represents overlaps between pairs of clusters. This can help a visualization analyst explore overlapping relationships among clusters. To clearly differentiate clusters from each other, the set of clusters can be separated in the visualization space into a limited number of partitions, each without overlapping clusters. We show that, with some assumptions, minimizing the number of partitions (views) is NP-complete, so we apply

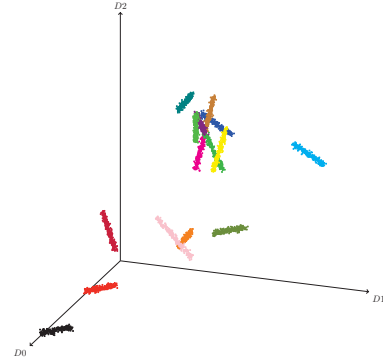


Figure 1: [2] Artificial data set, 3 dimensions, 15 clusters each containing 256 points (elongated data set 8, Section 3.1)

vertex-coloring heuristics to partition the graph. While some existing methods have similar goals, our approach is unique since it guarantees no overlap of clusters within the same view and limits the number of views using strong graph-theoretic heuristics. Moreover, our approach works with different methods for visualization (scatterplot, RadViz, parallel coordinates ...), clustering quality assessment, conflict detection and graph coloring methods. We illustrate it for clusters of real and artificial data, where all the visualization views of a given data set use the same 2-D scatterplot projection, with two different quality metrics, two conflict detection methods, and two graph coloring heuristics.

1.2 Related Work

1.2.1 Clustering

a) Cluster Formation: Clustering of data points may reveal significant interactions between various attributes of the data. Given a set of n points in \mathbb{R}^d , (d attributes), clustering separates the points into a set of k distinct clusters: $C = \{C_1, C_2, \dots, C_k\}$, where $|C_i| = n_i$ and $\sum_{i=1}^k n_i = n$. We assume that our input data is already clustered, so here we briefly mention a few of the approaches from the rich clustering literature and refer the reader to the comprehensive clustering overviews by Jain [3][4] and the categorization and discussion of clustering issues by Estivill-Castro [5]. The popular k -means partitioning algorithm seeks to minimize a squared error function [6][4]. Other approaches include incremental [7], randomized [6], graph-based [8], hierarchical [6], constructing cluster boundaries [9], and identifying irregularly shaped clusters [10].

b) Clustering Quality: Clusters are subjective and many automatic measures assess the “quality” of a clustering [11][12]. In this paper, clustering quality acts as a “sanity check” for our approach as we place overlapping clusters into separate views. Since indices can also suggest how intrinsically difficult a data set is, we want indices that work in both the data and visualization spaces. (Visualization-dependent quality is discussed in Section 1.2.1(c).) To illustrate our approach we use real and artificial data. Clusters in our artificial data tend to be convex, dense, and pairwise separable by a hyper-plane. The Dunn index [12] is effective at measuring dense and well-separated clusters. A high Dunn index indicates good clustering. The Davies-Bouldin index extension [12] uses the scatter within each cluster. A good (low) Davies-Bouldin value is associated with lower cluster scatter and higher cluster centroid distances.

c) Cluster Visualization: The visualization literature contains a wide variety of approaches for visualizing multi-dimensional data that may be classified or clustered (e.g. scatterplots [13], Radviz [14], parallel coordinates [15]). Methods exist to select a particular visualization, such as Projection Pursuit [16] and xGOBI [17]. We illustrate our conflict graph-based approach using 2-*D* scatterplots [13] with orthographic projection. As stated above, scatterplots (and variants) are a common visual encoding technique for dimensionality reduction [1]. Figure 2 shows an orthogonal scatterplot of the data from Figure 1 in two dimensions.

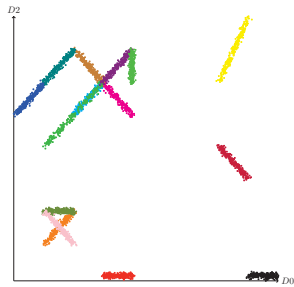


Figure 2: [2] Scatterplot for data set in Figure 1 projected orthographically onto the D_0D_2 plane

Of particular interest are methods for selecting scatterplot projections that preserve clustering structure well, especially those validated by users. Successful methods address challenges inherent in the usage of high-dimensional data and scatterplots [18][19][20]. Tatu *et al.* [18] describe a method to limit the number of “interesting” lower dimensional visualizations. Two cluster evaluation metrics for scatterplots are discussed: 1) the pixel-based Class Density Measure (CDM) and 2) the Histogram Density Measure (HDM). Sips *et al.* [21] select a pair of axes for a scatterplot of classified data using a Class Consistency Measure (CCM). Two CCM methods are: 1) Centroid Distance, leading to

Distance Consistency, and 2) Distribution Consistency. Tatu *et al.* [22] describe visual quality metrics and their relation to human perception. They study CDM and HDM from [18], as well as CCM from [21]. For a small user study, they report that the best measure for the UCI wine data set [23] is 2-*D* HDM, followed by CCM. Sedlmair, Tatu and Tory [1] discuss several dimensional reduction techniques (including t-SNE [20] and PCA) in addition to Distance and Distribution Consistency [21] and the 2-*D* form of HDM [18]. These, together with an extensive user study, result in a comprehensive taxonomy of visual cluster separation factors.

1.2.2 Multiple Visualization Views. The IEEE conference on Coordinated and Multiple Views in Exploratory Visualization (CMV) addresses many aspects of multiple views. Roberts [24] captures some of this state-of-the-art in 2007. Earlier work by Baldonado, *et al.* [25] offers guidelines for when multiple views of a data set are appropriate. Selecting different subsets of attributes for different views is discussed. In contrast, our partitioning process creates partitions that each have the same attributes but different clusters. Two other guidelines from [25], in opposing directions, are also relevant for our work: decomposition and parsimony. We strive for balance: decompose data to place overlapping clusters in separate views, and reduce the number of views via graph coloring methods.

1.2.3 Conflict Graph. We create a *conflict graph* for our clusters, where each vertex represents a cluster and an edge connects two vertices if their clusters “overlap” (see Section 2.2). Conflict graphs are not new, but have been used in a different way in existing literature. For example, the computation of a convex hull in \mathbb{R}^3 described by de Berg uses bipartite conflict graphs [26].

1.2.4 Graph Coloring. We color the aforementioned conflict graph to place conflicting (overlapping) clusters into different “conflict-free” partitions. Coloring a graph using the minimum number of colors [27] is applicable. The chromatic number $\chi(G)$ of a graph G is the minimum number of colors required. The associated decision problem is NP-complete [27]. Clusters corresponding with like colored vertices do not conflict and may be in the same partition. Because finding $\chi(G)$ is NP-complete, heuristics are often used to color a graph, such as MAXIMAL-DEGREE-COLORING [28] (colors the highest degree, currently uncolored, vertex with the lowest allowable color), and COLORING-BACKTRACKING [29] (backtracks to attempt to color the graph with fewer colors).

Upper and lower bounds on $\chi(G)$ that we use are:

- Given a degree sequence $\langle \deg v_1, \deg v_2, \dots, \deg v_n \rangle$ for a graph $G = (V, E)$, $v_i \in V$, and $\deg v_i \leq \deg v_{i+1}$ for all i from 1 to $n-1$, we have $\chi(G) \leq 1 + \max_i \min\{\deg v_i, i-1\}$ [28].
- $\chi(G) \geq \omega(G)$, where $\omega(G)$ is the clique number for G ,

the maximum size of a set of pairwise adjacent vertices of G [30]. (Finding $\omega(G)$ is also NP-complete [27].)

1.3 Overview

Section 2 describes our flexible visualization framework for generating multiple conflict-free views; within each view no clusters overlap and each cluster appears in exactly one view. It allows choice of visualization type (not just scatterplots), clustering quality indices, conflict detection, and graph coloring methods. We illustrate our approach for clusters of artificial data and real UCI machine learning repository data [23], where clustering quality is assessed using the two indices from Section 1.2.1(b). All our views of a data set use the same 2- D scatterplot projection. We present two conflict detection methods applicable in a visualization space. We apply the framework, with these two conflict detection methods and the two graph coloring heuristics from Section 1.2.4. Heuristics are appropriate because, even for our two simple conflict detection methods with a simplified distance measure, we show that producing a minimal number of conflict-free partitions for clustered data is NP-complete. Section 3 presents our results for scatterplots.

An earlier version of this work appeared in [2]. Here our conflict detection methods from [2] are generalized beyond the 2- D context. Adjustments are made to the NP-hardness proof. An additional UCI data set (wine) is included. A 2- D scatterplot for this data set, judged to be "good" by Sips *et al.* [21] (with Distance Consistency measure), is a starting point for wine data in our framework. This guarantees conflict-free views, demonstrating a possible synergy between scatterplot selection and our framework. Section 4 draws conclusions and suggests avenues for future work.

2. Methodology

Section 2.1 describes our framework for partitioning data into multiple conflict-free visualization views. Section 2.2 defines our conflict graph. Section 2.3 presents two conflict detection methods. We use heuristics to color our conflict graph since we establish the intractability of a version of the conflict-free partitioning problem in Section 2.4.

2.1 Visualization Framework (adapted from [2])

Figure 3 shows our framework for creating multiple conflict-free views of clustered data. Shading indicates visualization-dependent steps. Clustered input data is first normalized so that each dimensional value is in the range $[0, 1]$. Next, a visualization method is chosen. Conflict detection is then performed (Section 2.3) to create a conflict graph (Section 2.2). A threshold value τ , reflecting the amount of overlap between 2 clusters, is used to decide whether or not the clusters will have an edge in the conflict graph.

The estimate of χ , the chromatic number (Section 1.2.4) of the conflict graph, gives an estimate of the number of partitions required. We calculate an upper bound for this

estimate, using the degree sequence for the graph [28] (Section 1.2.4). If it exceeds a user's threshold (different from τ), then a further transformation of the data may be needed, or visualization selection might need to be done again (e.g. choose a different scatterplot projection). The conflict graph is colored, putting overlapping clusters into separate partitions, each with the same data attributes. The graph can be visualized and then τ may be adjusted if desired. The Partition Data Set box uses the color class information (the dashed line) to separate data for the various views, using any applicable visualization parameters.

The 2- D orthographic scatterplot is one example of a visualization that may be used within our framework, and we illustrate our process using scatterplots in Section 3. This same process may be applied to other visualizations, such as radial visualizations and parallel coordinates (see Section 4).

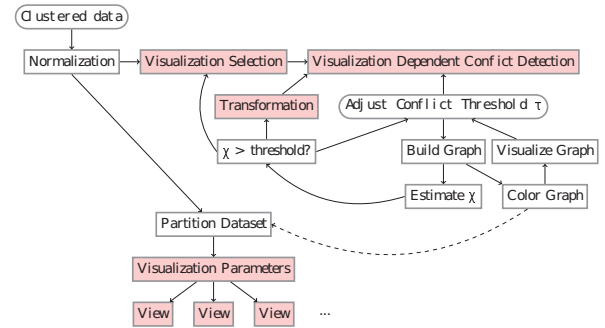


Figure 3: Framework for Multiple Conflict-Free Views

2.2 Conflict Graph ([2])

A conflict graph, $G = (V, E)$, is defined as an ordered pair of two sets. The non-empty set V is the vertex set and the (potentially empty) edge set E consists of sets of unordered pairs of distinct elements of V . Our vertex set V has k vertices, one for each cluster. E has an edge $\{i, j\}$ for each pair of distinct clusters C_i and C_j that may overlap. Clusters that "may" overlap are considered to be in conflict (see two conflict detection methods in Section 2.3). An edge is added to the conflict graph for each such pair of clusters. Each potential conflict is given a non-negative weight. At the time the graph is colored, only edges with a weight greater than threshold τ are considered to exist. An example of a conflict graph is in Figure 6 for the data from Figure 1.

2.3 Conflict Detection (generalized from [2])

Our two methods are chosen because they scale well in higher dimensions and are efficient to compute (linear in time). When we refer to clusters here, we assume that we are working in the visualization space. Distance may be calculated using one of several distance measures (L_1 (Manhattan), L_2 (Euclidean), L_∞). In both methods, an edge

between clusters C_i and C_j will be in the conflict graph if the weight $0 \leq w_{ij} \leq 1$ (defined below) $> \tau$.

2.3.1 Bounding Box Method. In our first conflict detection method two clusters C_i and C_j conflict if their bounding boxes intersect. An example of this is in Figure 4 (left) for a 2- D scatterplot. Here the bounding boxes for the cyan and magenta clusters do intersect. Here we define the weight, w_{ij} , as 0 if the bounding boxes of C_i and C_j do not intersect; otherwise w_{ij} is the ratio of the volume of the overlap between the two bounding boxes over the minimum of the volume of C_i 's bounding box and the volume of C_j 's bounding box. For 2- D , bounding boxes are axis-aligned rectangles. Bounding boxes appear to work well for elongated clusters (see Sections 3.1 and 3.2).

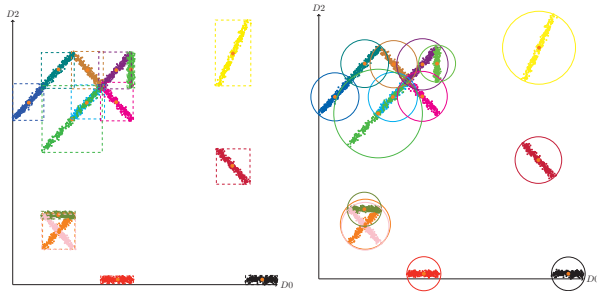


Figure 4: [2] D0D2 Scatterplot with centroids (orange), bounding box conflict detection (left), and centroid-radius conflict detection (right) for each cluster in Figure 2

2.3.2 Centroid-Radius Method. Another method is our *centroid-radius* approach, which yields a shape centered at the centroid of the cluster. For 2- D with L_1 distance this constructs a diamond shaped region (square rotated by 45 degrees). For 2- D with L_2 distance, we obtain a circular region. For 2- D with L_∞ distance, we have an axis-aligned square. The centroid-radius (r'_i) of cluster C_i is defined as the maximal distance between the cluster's centroid b'_i , and all points in the cluster. We consider two clusters C_i and C_j , with centroids b'_i and b'_j , are not in conflict if $r'_i + r'_j < D_{ij}$, where $D_{ij} = \|b'_i b'_j\|$. We therefore define the weight, w_{ij} , as 0 if $r'_i + r'_j < D_{ij}$, and $((r'_i + r'_j)/D_{ij}) - 1$ otherwise. Figure 4 (right) is a projection of a 3-dimensional, 15-cluster data set onto the D0D2 plane. Here we see that a conflict between the cyan and magenta clusters has been detected, using L_2 distance. We also see a “false positive” conflict between the blue and cyan clusters. This conflict detection method appears to be more appropriate for globularly shaped clusters (see Sections 3.1 and 3.2).

2.4 Partition Data Set (adapted from [2])

Given a conflict graph for a data set, we then apply a graph coloring method to the conflict graph. Based on the coloring results we partition the data set so that within each

partition all of the elements correspond to vertices of the same color. The number of partitions (visualization views) is therefore the number of colors in the colored graph. Even for simple conflict detection methods like those in Section 2.3, but with a polynomial-time distance measure such as L_1 or L_∞ (avoiding irrational numbers), conflict-free partitioning is NP-complete, as we show below. In such cases, graph coloring heuristics are appropriate (Section 1.2.4).

Lemma 2.1: Separating the clusters into the minimal number of conflict-free partitions is in NP when polynomial-time cluster overlap detection is used.

Proof: Posed as a decision question: “Does there exist a partitioning of the clusters into at most n sets such that the clusters do not overlap within any of the sets?”. Given a family of sets, it may be verified to include all of the clusters in their union in polynomial time. Within each set, we assume that it may be verified in polynomial time that all of the clusters are pair-wise without overlap. With this caveat a proposed solution to the partitioning problem can be verified in polynomial time. ■

Lemma 2.2: Partitioning the clusters into the minimal number of conflict-free partitions is NP-hard for our bounding box and centroid-radius conflict detection methods, using L_1 or L_∞ distance.

Proof: A polynomial time reduction from the well-known NP-complete graph chromatic number problem (see Section 1.2.4) [27] is shown below. Given an undirected graph $G = (V, E)$, where $|V| = d$, construct a data set in d -dimensional space as follows: For each vertex v_i , a reference point (not part of the cluster) is defined as $(0, \dots, 0, 1, 0, \dots, 0)$, where all components except the i^{th} component are zero. A cluster of data points is constructed by placing a point $\pm \frac{3}{8}$ from the reference point in all d dimensions. For each vertex v_j , $j > i$, that is adjacent to v_i , 7 additional points are placed along the line segment connecting the reference points for cluster C_i and cluster C_j . These 7 points are members of cluster C_i and are $\frac{1}{8}, \frac{2}{8}, \dots, \frac{7}{8}$ of the way along the line segment connecting the reference points of C_i and C_j . Let p denote the point of C_i closest to C_j . Using bounding box conflict detection and either L_1 or L_∞ distance, p is on an edge of C_i 's bounding box, and is inside the bounding box of C_j , causing C_i to conflict with C_j . Using centroid-radius conflict detection and either L_1 or L_∞ distance, C_i also conflicts with C_j . For example, in 2- D with L_1 distance, $r'_i + r'_j > D_{ij}$, yielding conflict. By this construction, clusters C_i and C_j conflict if, and only if, there was an edge adjacent to vertices v_i and v_j . This polynomial time reduction places our problem in the class NP-hard. ■

The two possible arrangements for a graph of order (cardinality of the vertex set) 2 are shown in Figure 5. The left shows the two clusters created if the two vertices are not adjacent (in the graph E_2). The right image shows the cluster constructed when the two vertices are adjacent (in the graph P_2). In each image, the reference points are shown

as open circles. Each division of the grid is $\frac{1}{16}$ of a unit.

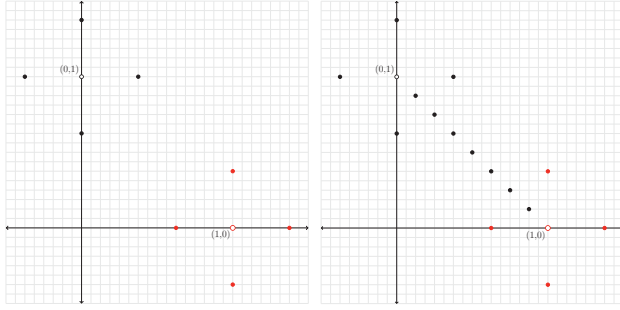


Figure 5: [2] Clusters created for E_2 , right for P_2 .

Following immediately from Lemmas 2.1 and 2.2:

Theorem 2.1: Partitioning the clusters into the minimal number of conflict-free partitions is NP-complete for our bounding box and centroid-radius conflict detection methods, using L_1 or L_∞ distance.

3. Results

3.1 Data Sets

Our 23 data sets' dimensionality varies from 3 to 13. Sixteen of our 23 data sets were generated artificially by adding points normally distributed about either: 1) a central point to form 8 globularly shaped data sets ([31][2]) with 8 to 12 dimensions, 5 to 8 clusters, and 600 to 10^6 points (to illustrate scalability), or 2) a line segment to form 8 elongated data sets [2]. The artificial clusters tend to be convex and pairwise separable by a hyper-plane. Additionally, 7 data sets were selected from the UCI Machine Learning Repository [23] and are summarized in Table 1.

3.2 Partitioning Results

In our experiments 2- D scatterplots were generated using orthographic projection with D_0D_2 axes. As expected, clustering quality in the unpartitioned visualization space is almost always at most as good as that of the data space, and separating overlapping clusters in the visualization space into multiple views almost never negatively affects quality. The two graph coloring heuristics are sufficient, in all but one of our test cases, to optimally partition the clusters into multiple views without additional transformations and/or visualization selection. Below we summarize results for our artificial data sets; for more detail see [2]. We do give detailed results here for the UCI data sets (Tables 1, 2).

For the elongated data sets with bounding box conflict detection method, experiments were performed using $\tau = 0.1, 0.01$ and 0.001 [2]. The value 0.01 was selected since the other values produced either (subjectively) too many or too few visualizations. Since $\min w_{ij} \leq \tau \leq \max w_{ij}$ (where i, j range over the cluster indices), it may be useful to try several values for τ within this interval to produce an acceptable

number of visualizations. Figure 4 shows centroids and bounding boxes for the 15 clusters for elongated data set 8 (from Figures 1 and 2). Note that several of the bounding boxes have a slight overlap. Using $\tau = 0.01$, we obtain the conflict graph in Figure 6. Figure 7 shows the 4 resulting views that show all clusters without overlap. The conflict graph contains a 4-clique: {brown, magenta, cyan, lime}. Since there are no cliques of size greater than 4, the lower bound on χ of 4 matches the number of views and therefore the number of views here is optimal. For all of our elongated data sets the number of views is optimal. The partitioning process shows an improvement in both cluster quality indices for all data sets. Very low Dunn index values are consistent with cluster overlap in the visualization space. Each of these data sets required at least two views.

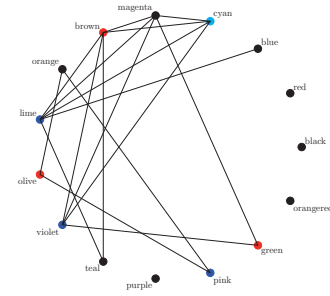


Figure 6: [2] Conflict graph for 4 colors for elongated data set 8, $\tau = 0.01$, from Figure 4

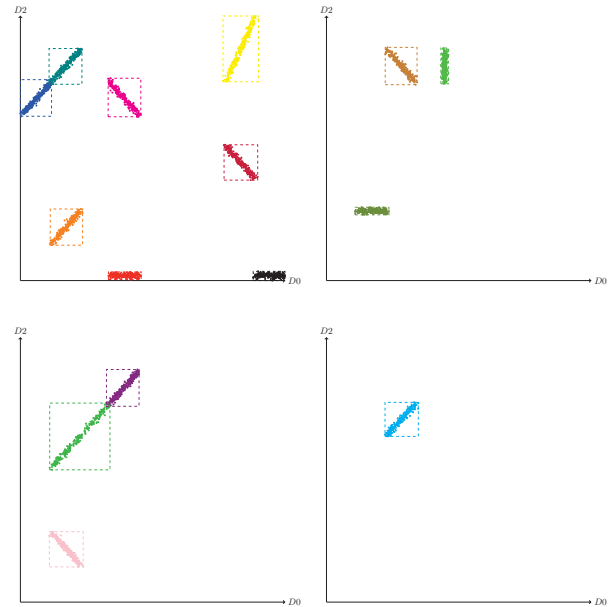


Figure 7: [2] The 4 visualization views required to show all clusters in elongated data set 8 without overlap

For the globular data, centroid-radius conflict detection is used. Clustering quality for the data space has high Dunn index and low Davies-Bouldin index values [2], suggesting well-separated clusters. There is an improvement in both indices, except for data sets 2 and 6, where the clusters were already well separated and one color was sufficient to color the graph. For all of the globular data sets the graph coloring heuristics produced an optimal number of views because the lower bound on χ was achieved. This holds even for globular data set 8, which contains 10^6 data points, illustrating the scalability of our approach.

For the UCI data, Table 1 shows clustering quality for unpartitioned data in the visualization space, whereas Table 2 presents results for the partitioned data. Table 2 gives the number of partitions (colors) using bounding box conflict detection, number of colors using, from Section 1.2.4, MAXIMAL-DEGREE-COLORING (Δ) or COLORING-BACKTRACKING (Backtrack), and clustering quality for the data partitioned into multiple views. Here the "Mean" column is the average of each index over all resulting views where the index can be calculated. The abalone UCI Machine Learning dataset (29 clusters) showed the greatest quality improvement when partitioned into 19 separate views. For the glass data set the 6 clusters were partitioned into 4 views consisting of 2 views of 2 clusters each, with the remaining 2 clusters in separate views. The seeds and stones data sets did not require any partitioning, as one color was sufficient. For the abalone data set, it was determined by inspection that the clique number is at least 3. For all of the other UCI data sets, the clique number was determined and the graph coloring heuristic produced an optimal number of views because the lower bound on χ was achieved.

Table 1: UCI Data Sets [23][2] - $D0D2$ scatterplots, unpartitioned data, Dunn and Davies-Bouldin (D-B) indices (extended from [2] to include wine data set)

#	Data space			Visualization space			
	d	k	n	Dunn	D-B	Dunn	D-B
Abalone	9	29	4178	0.018	8.716	0.000	27.663
Flea	6	3	74	0.236	0.895	0.045	1.132
Glass	9	6	214	0.022	4.421	0.000	4.326
Olive	9	9	572	0.036	2.474	0.000	3.250
Seeds	7	3	210	0.082	0.909	0.000	0.568
Stones	8	4	73	0.100	1.404	0.000	1.510
Wine	13	3	178	0.189	1.328	0.005	1.749

In Section 1.2.1(c) we observed that some measures of cluster separation in scatterplots, although they might not completely separate clusters, can be good starting points in our conflict-free partitioning framework. As an example, in Figure 8 we use a 2- D scatterplot for the wine UCI dataset, with dimensions alcohol ($D0$) and flavanoids ($D6$). According to Sips *et al.* [21], this projection is well-rated using the Distance Consistency measure. This projection is particularly interesting because Sips *et al.* [21] related good Distance Consistency ratings to good views selected by

Table 2: UCI Data Sets [23][2] - $D0D2$ scatterplots, partitioned data, bounding box conflict detection, lower and upper bounds on chromatic number χ , number of colors from the 2 heuristics, Dunn and Davies-Bouldin (D-B) indices (extended from [2] to include wine data set)

#	k	Number of colors				Mean	
		$\chi \geq$	$\chi \leq$	Δ	Backtrack	Dunn	D-B
Abalone	29	≥ 3	19	19	19	0.124	0.743
Flea	3	2	2	2	2	0.260	0.956
Glass	6	4	4	4	4	0.244	0.601
Olive	9	7	7	7	7	0.137	0.416
Seeds	3	1	1	1	1	0.000	0.568
Stones	4	1	1	1	1	0.000	1.510
Wine	3	3	3	3	3	N/A	N/A

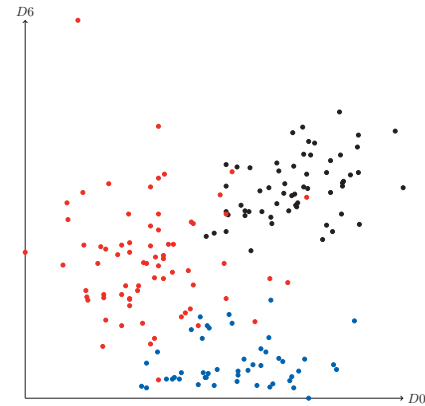


Figure 8: Scatterplot of Wine data with Alcohol ($D0$) and Flavanoids ($D6$) dimensions.

humans to provide some user validation. The black and blue clusters are well separated (zero overlap) using bounding box conflict detection. The red cluster overlaps with both the black and the blue clusters, but a human can see that the red cluster is mostly in the lower left. For $\tau = 0.01$ two views are produced. Using $\tau = 0.65$, one color is sufficient (and matches the lower bound).

4. Conclusions

We presented a general visualization framework for partitioning clustered data into multiple visualization views. The process creates a conflict graph for the clusters, then invokes graph coloring to partition the clusters into multiple views. Our framework is flexible, allowing different methods for different visualizations, clustering quality assessments, conflict detections and graph colorings. We illustrated the use of the framework via 2- D orthographic scatterplots, the Dunn and Davies-Bouldin cluster quality metrics, bounding box and centroid-radius conflict detection, and two heuristics from the graph coloring literature. In particular, we demonstrated potentially powerful synergy where good scatterplot selection from the literature may be a starting point for further improvement within our framework.

Our experiments use 23 data sets. Some are from the UCI machine learning repository [23] and others are artificially generated. One contains 10^6 data points to illustrate the scalability of our approach. The dimensionality varies from 3 to 13. The graph coloring heuristics we use are sufficient, in all but one of our test cases, to optimally partition the clusters into multiple views. This is despite the NP-completeness of a version of the multiple visualization problem, whose intractability we establish. Changes in cluster quality values suggest that the partitioning is operating as expected.

A few areas of future work are suggested by these findings. For the wine UCI data set, an alternative method of conflict detection might replace ratios of areas with the ratio of the cardinality of data points in the intersection over the cardinality of the cluster. The graph coloring heuristics tend to place as many vertices as possible in the early color classes (Figure 7). The use of an equitable coloring [32] method may alleviate this behavior at the expense of additional color classes. Another alternative may be to redistribute some of the clusters from one view to another. Processing the data set in the data space and identifying a clique cover [33] in the conflict graph may allow the choice of a different visualization for each clique. Although the paper uses 2-D orthogonal scatterplots, the framework can be used with 3-D scatterplots and several other types of visualizations. In [2] conflict detection methods for radial visualization and parallel coordinates were introduced and used within the framework; further development along these lines is another avenue for future work.

References

- [1] M. Sedlmair, A. Tatu, T. Munzner, and M. Tory, "A Taxonomy of Visual Cluster Separation Factors," *Computer Graphics Forum*, vol. 31, pp. 1335–1344, 2012.
- [2] R. Marceau, "Partitioning Data to Minimize Cluster Overlap using Multiple Visualization Views," Ph.D. dissertation, University of Massachusetts, Computer Science Department, 2015.
- [3] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, Sept. 1999. [Online]. Available: <http://portal.acm.org/citation.cfm?doid=331499.331504>
- [4] A. K. Jain and E. Lansing, "Data Clustering : 50 Years Beyond K-Means," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.
- [5] V. Estivill-Castro, "Why so many clustering algorithms: A position paper," *SIGKDD Explor. Newsl.*, vol. 4, no. 1, pp. 65–75, June 2002. [Online]. Available: <http://doi.acm.org/10.1145/568574.568575>
- [6] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (Pt.1)*. Wiley-Interscience, 2000.
- [7] L. O'Callaghan, N. Mishra, A. Meyerson, S. Guha, and R. Motwani, "Streaming-data algorithms for high-quality clustering," in *Proceedings of 18th International Conference on Data Engineering*, February 2002, pp. 685–694.
- [8] E. Hartuv and R. Shamir, "A clustering algorithm based on graph connectivity," *Information Processing Letters*, vol. 76, no. 200, pp. 175–181, 2000.
- [9] V. Estivill-Castro and I. Lee, "Automatic clustering via boundary extraction for mining massive point-data sets," in *Proceedings of the 5th International Conference on Geocomputation*, 2000.
- [10] A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. Vapnik, "Support Vector Clustering," *Journal of Machine Learning Research*, vol. 2, pp. 125–137, 2001.
- [11] E. Bertini, A. Tatu, and D. Keim, "Quality Metrics in High-Dimensional Data Visualization: An Overview and Systematization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, December 2011.
- [12] B. Desgraupes, "Clustering Indices," pp. 1–34, April 2013, from R documentation: <http://cran.r-project.org/web/packages/clusterCrit/vignettes/clusterCrit.pdf>.
- [13] J. M. Chambers, W. S. Cleveland, P. A. Tukey, and B. Kleiner, *Graphical Methods for Data Analysis (Wadsworth & Brooks/Cole Statistics/Probability Series)*. Duxbury Press, 1983.
- [14] K. Daniels, G. Grinstein, A. Russell, and M. Glidden, "Properties of Normalized Radial Visualizations," *Information Visualization*, vol. 11, no. 4, pp. 273–300, October 2012.
- [15] A. Inselberg, *Parallel Coordinates: Visual Multidimensional Geometry and Its Applications*. New York: Springer, 2009.
- [16] J. Friedman, W. Stuetzle, and S. U. P. Orion, "Projection Pursuit Methods for Data Analysis," June 1981. [Online]. Available: <http://www.dtic.mil/dtic/tr/fulltext/u2/a119824.pdf>
- [17] D. F. Swayne, D. Cook, and A. Buja, "XGobi: Interactive Dynamic Data Visualization in the X Window System," 1998. [Online]. Available: <http://lib.stat.cmu.edu/general/XGobi/papers/xgobi98.pdf>
- [18] A. Tatu, G. Albuquerque, M. Eisemann, J. Schneidewind, H. Theisel, M. Magnork, and D. Keim, "Combining automated analysis and visualization techniques for effective exploration of high-dimensional data," in *VAST 09 - IEEE Symposium on Visual Analytics Science and Technology, Proceedings*, 2009, pp. 59–66.
- [19] E. J. Wegman, "Hyperdimensional data analysis using parallel coordinates," *Journal of the American Statistical Association*, vol. 85, no. 411, pp. 664–675, September 1990.
- [20] L. V. D. Maaten and G. Hinton, "Visualizing Data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
- [21] M. Sips, B. Neubert, J. P. Lewis, and P. Hanrahan, "Selecting good views of high-dimensional data using class consistency," *Computer Graphics Forum*, vol. 28, pp. 831–838, 2009.
- [22] A. Tatu, P. Bak, E. Bertini, and D. Keim, "Visual quality metrics and human perception: an initial study on 2D projections of large multidimensional data," *Advanced Visual Interfaces (AVI)*, pp. 49–56, 2010.
- [23] A. Frank and A. Asuncion, "UCI machine learning repository," 2010, <http://archive.ics.uci.edu/ml>.
- [24] J. C. Roberts, "State of the art: Coordinated & multiple views in exploratory visualization," in *Coordinated and Multiple Views in Exploratory Visualization, 2007. CMV'07. Fifth International Conference on*. IEEE, 2007, pp. 61–71.
- [25] M. Q. Wang Baldonado, A. Woodruff, and A. Kuchinsky, "Guidelines for using multiple views in information visualization," in *Proceedings of the Working Conference on Advanced Visual Interfaces*, ser. AVI '00. New York, NY, USA: ACM, 2000, pp. 110–119. [Online]. Available: <http://doi.acm.org/10.1145/345513.345271>
- [26] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*. Springer, 2008.
- [27] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1979.
- [28] R. Gould, *Graph Theory (Dover Books on Mathematics)*. Dover Publications, 2012.
- [29] H. T. Lau, *A Java Library of Graph Algorithms and Optimization (Discrete Mathematics and Its Applications)*. Chapman and Hall/CRC, 2006.
- [30] G. Chartrand and P. Zhang, *A First Course in Graph Theory (Dover Books on Mathematics)*. Dover Publications, 2012.
- [31] A. Russell, R. Marceau, F. Kamayou, K. Daniels, and G. Grinstein, "Clustered Data Separation via Barycentric Radial Visualization," in *Proceedings of the 2014 International Conference on Modeling, Simulation and Visualization Methods (MSV)*, 2014, pp. 101–107.
- [32] H. Kierstead, A. Kostochka, M. Mydlarz, and E. Szemerédi, "A Fast Algorithm for Equitable Coloring," *Combinatorica*, vol. 30, no. 2, pp. 217–224, 2010.
- [33] D. B. West, *Introduction to Graph Theory (2nd Edition)*. Prentice Hall, 2000.

Real life augmented reality for maintenance

John Ahmet Erkoyuncu¹, Mosab Alrashed¹, Michela Dalle Mura², Rajkumar Roy¹, Gino Dini²

¹Cranfield Manufacturing, School of Aerospace, Transport and Manufacturing, Cranfield University, United Kingdom

²Department of Civil and Industrial Engineering, University of Pisa, Italy

Abstract - *A major challenge for Augmented Reality (AR) in real life maintenance is varying lighting conditions. This research developed a novel registration technique to use AR effectively in real life lighting conditions, where registration is the accurate alignment of real and virtual images. The study has demonstrated that the registration technique can register shiny samples and implements image enhancements on dim samples within a Non-Destructive Testing environment. The experimental set-up included recognition efficiency testing on shiny and dim samples. A detailed aerospace maintenance case study has been used to validate the registration technique. The results show the duration of registration reduced and the accuracy improved for both shiny and dim samples.*

Keywords: *Augmented reality, registration, maintenance, machine learning,*

1 Introduction

Maintenance involves a range of preventive and corrective actions taken to sustain and enhance the use of equipment [1]. These actions are typically dependent on pre-defined procedures according to the maintenance task. In this process, maintainers can have training requirements and may need assistance during maintenance interventions [2]. Reducing the cost and duration of maintenance are commonly referred targets [1; 3]. There are a number of characteristics of maintenance that promotes the use of AR, including [4]:

- Use of standardized procedures.
- Maintenance refers to data usually available on bulky manuals (e.g. AR allows an easy access to technical documentation without using paper manuals).
- Maintenance is often carried out “in the field”.

In terms of AR applications good potential has been realised in areas such as: a) user interfaces, which can be rendered in a ubiquitous manner that can train maintainers and deskill maintainer tasks, and b) maintenance data logging and management that allows remote collaboration [5]. Within the

remote maintenance context, AR allows combining the maintainers real vision with an ‘expert’ located at distance and overlaying useful information to assist with providing warning signs, interact with 3D models, diagnosing faults, assist with performing unfamiliar maintenance tasks and even giving maintenance task instructions [6]. The remote maintenance can be facilitated with graphics, video, and audio.

Application of AR is largely still at the prototype stage and has typically not achieved wide adoption in industry [6]. There are a number of challenges faced in real life application of AR including registration, latency, calibration and human factors [7]. Registration refers to the accurate alignment of real and virtual images when the user moves his/her head or viewpoint [7]. If we assume the position of the sensor relative to the display is fixed, registration may be split into two parts: accurately calibrating each eye’s display relative to the sensor, and accurately tracking the sensor’s position. In the case of tracking there is a reliance on various sensors integrated in the registration system. Therefore, registration includes calibration and tracking. A potential drawback for some sensor-based methods for real life maintenance is that the equipment may be attached to the user at all times [6]. In contrast, the vision-based approaches can take a sensor-less approach for registration [7]. Though, these methods commonly face a lack of robustness [10]. Some of the key drivers for the registration challenges for vision-based systems include: varying lighting conditions, reflections, shiny surface, shadows, dust, dirt, rust, etc [10]. Accordingly, this paper focuses on offering an innovative registration approach for shiny and dim surfaces, applied in a system for delivering real life maintenance.

1.1 Structure of paper

The paper is structured as follows: Section 2 provides an overview of the proposed registration system. Subsequently, Section 3 covers the experimental set-up for the system. Section 4 presents the experimental results for registration approach. Section 5 provides the conclusions and future work.

2 The proposed registration system

The registration system that is presented in this paper relies on machine learning algorithms that aim to enhance the ability to register different types of material with varying surface characteristics (e.g. shiny vs dim).

2.1 AR assisted inspection for aircraft maintenance

An AR assisted inspection system was developed to guide with maintenance activities for aircraft engine. The unique feature of the system is the adaptive AR registration capability for shiny and dim samples in Non-Destructive Testing (NDT) processes. These characteristics are significant drivers in NDT [11]. The system can also be controlled by an expert or skilled team that can be remotely connected to the cloud. Two stories were created with regards to AR for the aircraft maintenance case study:

- Checking the jet engine fan blade from a remote distance using a tablet. This logic was illustrated through a storyboard as represented in Figure 1. The figure shows that the use case relies on checking damage size and type using 3D and infrared to report an outcome. The registration technique is applied to enhance the ability to align the virtual and real objects for different sample types.
- Using wearable glasses to lead and supervise the repair process.

Figure 1 presents the scenario where an expert is working on a mobile tablet from a remote distance. The steps include:

- Step 1: The mobile tablet connects to the robot in a particular area for the aircraft that needs to be tested.
- Step 2: After the robot finishes registering the degradations of interest, a real view of the engine appears enhanced by the number of things in need of repair, either as warnings or recommendations on the specific component from past knowledge.
- Step 3: Clicking on any issue gives important data associated with that issue, e.g. damage type and size.
- Step 4: For more details, like the 3D view, there is an ability to show required data at specific locations.
- Step 5: An additional function displays the damage type in detail including primary and secondary damage.
- Step 6: Option to have a full technical report sent after manual maintenance, automatic maintenance or both.

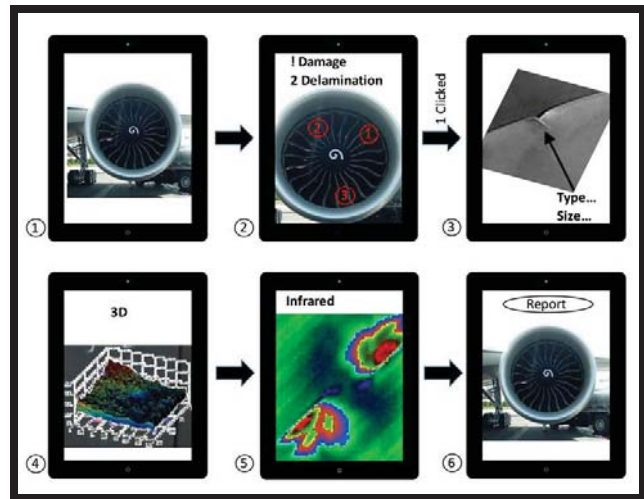


Figure 1. Storyboard AR assisted remote maintenance

The data related to the registration system is presented through a sequence of steps followed as listed below:

- The connecting infrared camera takes a shot that consist of significant images during a period of time in RAW file. The infrared camera needs to connect and perform simultaneously with the flash to provide the required heat.
- Using image analyser methods the 2D and 3D images are generated with some technical details like damage size and type.
- Develop a database for the materials, registered for future machine learning that can be used for AR.

2.2 The registration system

A mobile robot was utilised to help the inspection of aircraft engine blades in a faster and more effective manner. The registration system is demonstrated in Figure 2, (Double Robot – telepresence robot) consists of a motherboard (CPU), batteries, moving motor and wheels. Furthermore, the following hardware was integrated into the robot to enable the registration and AR function: flash device, infrared camera, RGB Camera, and two open slots. The flash device provides heat to the surface so that the infrared and RGB cameras recognise the damages (e.g. cracks, air gapes and delamination on the blade surface) and provide the live streaming used for the registration and AR technology. There are two open slots that could be used for ultrasound laser inspection and/or a snake camera (RGB) depending on the quality of the result needed in the inspection.

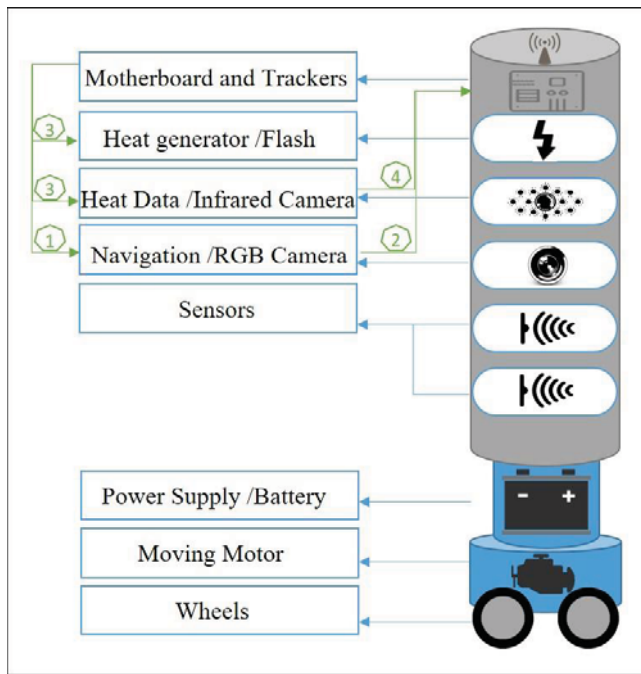


Figure 2. Mobile registration system

Developing the adaptive registration technique required robust capture of images and powerful development in computer vision in order to understand what happens in front of the camera. A significant number of software applications were utilised for this project; where the main algorithms are open-source software, called IDEA, developed in Matlab. This focuses on analysing the data coming from infrared and RGB cameras. Some of the software, scripts and libraries are OpenCV, images, object recognition libraries and external support libraries for Matlab. OpenCV was considered as it offers one of the best programming libraries to provide a real-time analysis and recognition of the environment surrounding the camera [14]. Also, TeamViewer was used for remote maintenance. As it is open-source, anyone can use and upgrade it in a manner appropriate to the maintenance field. Figure 3 demonstrates a real life picture of the architecture of the registration system using the Double Robot.

The process for how the data flows is as follows: first the data enters the maintenance cloud to be analysed, processed, stored, disseminated and backed up; then it flows to the operations platform, which is designed to handle the amount of data required to display it on the data presenter. Moreover, to have a full AR control there will be an interaction between the operations platform and data controller as the data moves through the data presenter. If needed, if one must customise McRobot with regard to gathering data, there is a function on the controller that works as a bridge between the operation platform and the maintenance cloud. This controller manages the position of taking images and helps to concentrate in a particular area.

As McRobot is an open source robot any plugin added to the robot can be controlled by the controller as well. For example, if a camera plugged into McRobot, the movement and area of inspection for the camera can be monitored remotely through this controller.

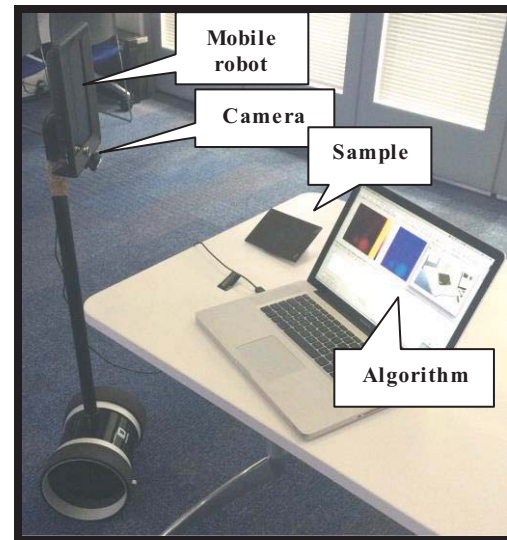


Figure 3. Architecture of the registration system

A markerless 3D registration approach was adopted, which is one of the most advanced optical registration approaches. This helps to determine the “real” measures in relation to the “visuals” by detecting the spatial properties of objects (including location in relation to camera and other objects). This involved building a map of 3D distinctive features (e.g. point descriptors). In this process the authors ensured that the 3D object had sufficient visual features. The process for the overall registration is demonstrated in Figure 4.

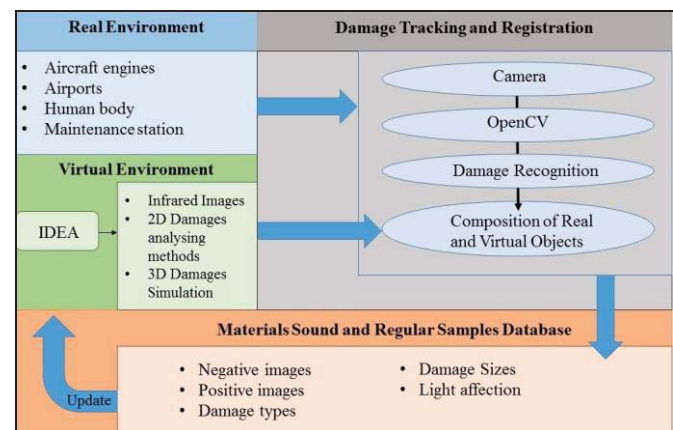


Figure 4. Overview of the process for registration

The algorithm developed in Matlab is used for the registration process. The algorithm was based on the Cascade Classifier developed by Intel [10]. This included:

- Train cascade object detector model
- Image category classification using bag of features
- Vision cascade object detector system object

The developed algorithm focuses on supporting the registration process with shiny and dim real-world samples. This applies machine learning and relies on the changes in the sample surface by collating a large number of samples without damage including dissimilar angles and various lighting conditions. The same happens for damaged images, so a database was built with the classified location of damages. Using that database, the algorithm helps to learn from the past and analyses the current damages. The algorithm helps with improving: the time it takes for registration and the range of coverage. Furthermore, a timer was designed for the inspection to automate registration.

2.3 Overview of proposed innovation

The hardware solution of McRobot was designed according to the aerospace requirement to provide all NDT tools in one device that could be automated to do the inspection and be customised as needed. Solving the problem of big data and data processing, the cloud handled central processing, storing, managing and connecting the whole software life cycle. Open-source inspection software was updated and tested with AR to satisfy the aim of this paper. Remote connection to stream augmented information was tested and evaluated, even though there are some disadvantages to distant networks, like lags, speed, and security. Nevertheless, these drawbacks can be avoided through private cloud in future.

3 Experimental set up

The developed registration capability was tested on shiny and non-shiny/dim fan blade composites. The shiny sample was made of carbon fibres as shown in Figure 5a. The dim sample is cross-woven carbon composite laminate, as shown in Figure 5b.

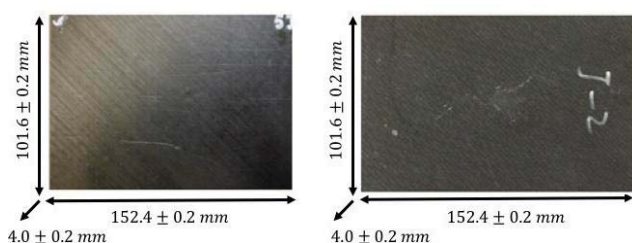


Figure 5. a) Shiny sample; b) Non--shiny sample.

A set of tests were designed to measure the efficiency of the registration method across a number of aspects for both the shiny and non-shiny samples. These tests involved counting the number of correct and flawed recognitions over a period of time while the camera is moved in a pre-defined manner. All data surrounding and including the experiments was recorded to make sure these experiments are repeatable in similar conditions. The following presents an overview of the tests conducted:

1. The demand experiment counted the number of seconds and damages simultaneously to compare the performance of the algorithms and functions used.
2. The diagnostic testing focused on the size and location of the damages, which was recorded as part of this experiment to be analysed, compared, and validated.
3. The objective of the registration test was to determine where the camera for image recognition needs to be located and to test the relationship between the machine learning distances. This evaluation was done with both fixed and stable camera locations that would allow an observer to have a full view of the plant.

4 Experimental results

This section presents an overview of the experimental results by covering the registration effectiveness and the machine learning capability.

4.1 Registration effectiveness

In order to conduct the experiments, the first step involved registering the location and calculating the size of the objects to be enhanced. Subsequently, the locations were learned or recognised from multiple tests using OpenCV libraries. Then the size of the damage to enhance at the surface of samples was calculated to align the real world and the virtual object. The final step involved tracking the points detected (e.g. location of damages) and sizes calculated for the composite surfaces.

4.2 Machine learning investigation results

A library of images was developed to learn from existing shiny and dim samples and to build comparisons. As part of the experiment, the developed software compiled 250 images from samples with damages for the damage area from different angles and light positions and 200 images from non-damage samples from diverse angles and bright positions. There is no limit as to how many images can be collected, though the specified amounts were considered to be feasible. Furthermore, a larger number of images for

damaged samples were collected because they required gathering more images from the surface.

Figure 6a, b, and c show the damage investigation on a test of 60 seconds by moving the two shiny samples in front of the camera. This sequence of the experiment included: 1) test damaged sample, if yes, the damage was detected and counted, 2) test non-damaged sample, if yes then no damage was detected, 3) validate that it can still detect damage on original sample, if yes, whether it still correctly detects damage. The tests were conducted in the same lighting conditions and the figure demonstrates the shiny and dim nature of the materials. There was no reflection from the flash device. In the first instance, the sound sample (with damage) 'S', and the first damage was detected and highlighted the number of damages found. Then, moving to the non-sound, 'NS', sample no damages were detected on it. Finally, testing on a sound sample 'S' with more than one damage was calculated (The test was done at the AR-Lab at Cranfield University, 11:30 p.m., 21°C and at 50 % of humidity). In Figure 6 all the green dots are measurement points overlaid on to the real sample and the large yellow square presents the identified number of damages (e.g. in Figure 6a the damaged sample contains one damage compared to 2 in Figure 6c).

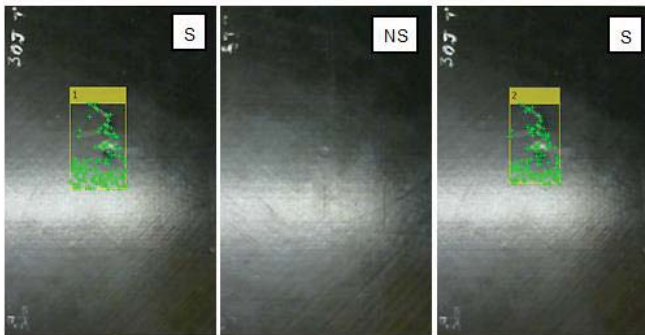


Figure 6. a) Sample with one damage; b) Sample with no damage; c) Sample with two damages.

4.3 Outcomes of the proposed process

A test was developed to measure the efficiency of the registration method. The tests were done four times for each sample type, and every test lasted 60 seconds with the camera moving at 90° angle base to simulate a real investigation scenario. Tests were conducted in relation to the registration duration, and registration distance.

Table 1 demonstrates the results across the shiny and dim samples. Overall across the 4 tests the shiny sample was on average 1.6 seconds quicker to register than the dim sample. Also, the registration durations were reduced for both samples from the first to the fourth test, which implies that the registration system is learning. All samples were taken from a height of 297 mm with the same light and

environmental conditions. The results in test 4 show that the dim sample provided greater distance (between 48 and 248 mm) than the shiny sample.

Table 1 Tests for registration duration and distance

Test	Registration duration results		Registration distance	
	Shiny	Dim	Shiny	Dim
	seconds	Seconds	Mm	Mm
1	1.51	3.29	594 ± 50	742.5 ± 50
	1 Damage	1 Damage		
2	1.97	3.37	592 ± 50	741 ± 50
	2 Damages	1 Damage		
3	1.72	3.32	594 ± 50	742 ± 50
	1 Damage	2 Damages		
4	1.46	2.90	593 ± 50	742.4 ± 50
	2 Damages	2 Damages		

An examination test was developed to measure the accuracy of the machine learning investigation method by counting the number of correct and noise registration for both shiny and dim samples; as represented in Table 2. Noise was considered when the registration was inaccurate. The tests were conducted four times for each sample type, and every test lasted 60 seconds with the camera moving at 90° angle base to simulate a real investigation scenario. Across the tests the number of noise and correct registration were observed, where the number of 'noise' items decreased for both samples. This demonstrates the machine learning process increased as the tests progressed.

Table 2 Tests for registration accuracy

Test	Shiny		Dim	
	Correct	Noise	Correct	Noise
1	2	2	3	1
2	2	1	3	0
3	3	1	5	0
4	2	0	6	0

After developing the ability to register the damage on dim and shiny samples, the capability to enhance infrared images became easier and more practical. The only new method was calculating the size of the enhanced image related to the distance from the sample surface. The infrared images were generated from the same software (IDEA) after analysing the raw files. Figure 7a shows the enhancement result for the infrared image on a dim sample.

The infrared image is only a demo image, not the actual damage image, and represents the demonstration of AR.

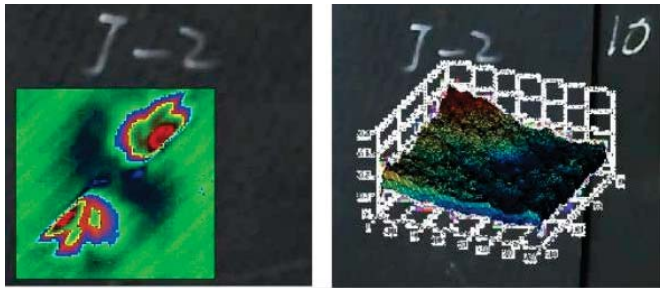


Figure 7. a) Image enhancement on dim samples; b) 3D enhancement on dim samples

The same technique of using image enhancement was used in 3D enhancement. These images were generated from the IDEA software by analysing the infrared images and the raw files together (as shown in Figure 7b) and the AR was utilised on each sample. In this process the alignment with a virtual object was also a target. The same method was used to calculate the size based on the distance moving on the sample surface. The developed software solution allows sharing these images with a remote partner as covered in Figure 1.

5 Conclusions and future work

An open-source hardware and software system was developed to assist with registration challenges in AR within a real life remote maintenance context. The novelty of the paper is two folds: a) a learning based adaptive registration technique for AR has been developed for shiny and dim samples, and b) the designed mobile system for inspection of real life maintenance. The developed registration approach was tested in a real life aerospace maintenance context on shiny and dim samples of fan blades. Relevant NDT tools have been used in an integrated mobile robot. The tests for the novel learning based registration technique demonstrate:

- Light reflection on shiny materials may give the RGB camera a shorter timeframe for AR registration of damages than dim materials.
- The distance for damage registration depends on the distance from the initial learning point and the light reflected angles.
- The dim materials are more recognisable than shiny materials from further distance.
- The registration of shiny sample can be quicker than the dim case.
- Image enhancements and 3D enhancements offer opportunities for better inspection times and

understanding of the maintenance damage type, location and size.

As registration algorithms improve there is an expectation that inspection processes can be automated to assist remote real life maintenance.

Acknowledgements

This research was partially supported by the Engineering and Physical Sciences Research Council (EPSRC) Centre for Innovative Manufacturing – Through-life Engineering Services. Grant number EP/1033246/1 and it was a cooperative work between Cranfield and Pisa Universities.

References

- [1] Ong, S. K., & Zhu, J., 2013, A novel maintenance system for equipment serviceability improvement, *Annals of the CIRP*, 62/1: 39–42.
- [2] Koch, C., Neges, M., König, M., & Abramovici, M., 2014, Natural markers for augmented reality-based indoor navigation and facility maintenance. *Automation in Construction*, 48:18–30.
- [3] Manuri, F., Sanna, A., Paravati, G., Pezzolla, P., & Montuschi, P., 2014, Challenges, Opportunities, and Future Trends of Emerging Techniques for Augmented Reality-Based Maintenance. *IEEE Transactions on Emerging Topics in Computing*, 2/4: 411–421.
- [4] Dini, G., Dalle Mura, M., 2015, Applications of augmented reality techniques in through-life engineering services, *Proceedings of the 4th International Conference on Through-life Engineering Services*, 38: 14–23.
- [5] Langlotz, T., Mooslechner, S., Zollmann, S., Degendorfer, C., Reitmayr, G., and Schmalstieg, D., 2012, Sketching up the world: in situ authoring for mobile Augmented Reality, *Personal and Ubiquitous Computing*, 16/6: 623–630.
- [6] Nee, A. Y. C., Ong, S. K., Chrysosouris, G., and Mourtzis, D., 2012, Augmented reality applications in design and manufacturing, *Annals of the CIRP*, 61/2:657–679.
- [7] Zhu J., Ong S.K. and Nee, A.Y.C., 2015, A context-aware augmented reality assisted maintenance system, *International Journal of Computer Integrated Manufacturing*, 28/2: 213–225.
- [8] Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., MacIntyre, B., 2001, Recent Advances in Augmented Reality. *IEEE Computer Graphics and Applications*, 21(November): 1–15.
- [9] Ong, S. K. , Pang, Y. Nee, A. Y. C., 2007, Augmented Reality Aided Assembly Design and Planning. *Annals of the CIRP*, 56/1: 49–52.
- [10] Pulli, K., Baksheev, A., Korniyakov, K., Eruhimov, V., 2012, Real-time computer vision with OpenCV, *Communications of the ACM*, 55/6: .61-69.
- [11] Mehnen, J., Tinsley, L., Roy, R., 2014, Automated in-service damage identification, *Annals of the CIRP*, 63/1: 33–36.

Reconstruction of Uniform Sampling from Nonuniform Sampling Using Discrete Cosine Transform

Sung-won Park

Department of Electrical Engineering and Computer Science, Texas A&M University-Kingsville
Kingsville, TX, USA

Abstract — Reconstruction of uniform sampling from nonuniform sampling using the DCT is described. It was shown by experiment that the reconstruction using the DFT of the symmetrically extended sequence improved the performance greatly as the symmetric extension avoids discontinuity that adds high-frequency content [1]. The DCT of a sequence is equivalent to the DFT of the symmetrically extended sequence. In this paper, the relationship between the DCT of a uniformly sampled sequence and the DCT of a nonuniformly sampled sequence is obtained. Using the relationship an algorithm to reconstruct uniform sampling from nonuniform sampling has been developed.

Keywords — uniform sampling, nonuniform sampling, DFT, DCT

I. INTRODUCTION

Reconstruction of a uniformly sampled sequence from a non-uniformly sampled sequence using the DCT is described in this paper. Computing the DFT of a signal is the same as computing the Fourier series coefficients of the periodically extended signal. If the extended signal has discontinuity at the junction of extension, it will unduly add high frequency content and increase the bandwidth of the extended signal especially when the signal is very short.

To prevent such discontinuity, the symmetric extension has been considered [1]. It has been shown by experiment that the reconstruction using the half-sample symmetric extension performed the best. Instead of using the DFT of the symmetrically extended signal we propose to use the DCT for the reconstruction. Because DCT does not need doubling of the sequence length and complex operations, the new method requires less computational complexity and would be more desirable. Out of four possible DCT techniques we used the DCT-2 [2]. The DCT-2 of a sequence is identical to the DFT of the half-sample symmetrically extended sequence.

The paper is organized as follows. In section II, the relationship between the DFT of a uniformly sampled sequence and the DFT of a nonuniformly sampled sequence is obtained and the perfect reconstruction condition is explained. In section III, the relationship between the DCT of a uniformly sampled sequence and the DCT of a nonuniformly sampled sequence is obtained. From the relationship, the formula to reconstruct the DCT of the uniformly sampled sequence from

the DCT of the nonuniformly sampled sequence is derived when the nonuniform sampling ratios are known. In section IV, reconstruction experiments are presented. Finally, a conclusion is made in section V.

II. RELATIONSHIP BETWEEN UNIFORM SAMPLING AND NONUNIFORM SAMPLING

Uniform sampling means that a continuous-time signal, $x(t)$, is sampled uniformly at $t = 0, T, 2T, \dots, (N-1)T$ where the sampling interval or period, T , is constant. Nonuniform sampling means that the sampling interval is not constant as shown in Fig. 1. Throughout this paper we assume that the number of samples taken between 0 and NT [s] is N for nonuniform sampling so that the average sampling interval is T . The nonuniform sampling ratios, α_n , are assumed to be known parameters.

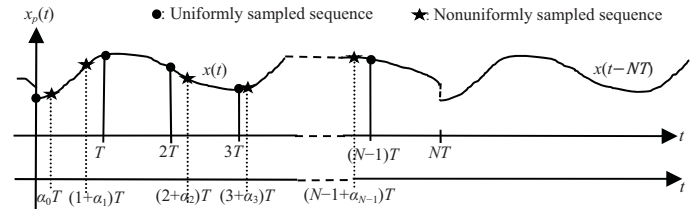


Fig. 1. Uniform and nonuniform sampling. T is the sampling interval for uniform sampling, α_n is the nonuniform sampling ratio, and N is the number of samples. The periodic signal $x_p(t)$ is obtained by extending $x(t)$ periodically with the period of NT .

Let $x_p(t)$ be the periodic signal that is obtained by periodically extending $x(t)$ with the period of NT [s] where N is an odd integer. When N is even, a similar procedure can be followed. The periodic signal, $x_p(t)$, will have its Fourier series with the fundamental radian frequency of $2\pi/NT$ [rad/s] as shown in equation (1) if the periodically extended signal has no harmonics greater than $(N-1)/2$:

$$x_p(t) = \sum_{k=-\frac{N-1}{2}}^{\frac{N-1}{2}} F_k e^{jk \frac{2\pi}{NT} t} \quad (1)$$

where F_k are the Fourier series coefficients and $j = \sqrt{-1}$.

If N samples of $x_p(t)$ are taken uniformly between 0 and NT with the sampling interval, T , then equation (1) becomes

$$x_p(nT) = \sum_{k=-\frac{N-1}{2}}^{\frac{N-1}{2}} F_k e^{jk \frac{2\pi}{NT} nT} \text{ for } n = 0, 1, 2, \dots, N-1. \quad (2)$$

Because $x_p(t)$ is identical to $x(t)$ for $0 \leq t < NT$, equation (2) can be rewritten as a sequence

$$x(n) = \sum_{k=-\frac{N-1}{2}}^{\frac{N-1}{2}} F_k e^{jk \frac{2\pi}{N} n} \text{ for } n = 0, 1, 2, \dots, N-1. \quad (3)$$

Now equation (3) can be rewritten as

$$x(n) = \sum_{k=0}^{N-1} \frac{X(k)}{N} e^{j \frac{2\pi}{N} kn} \text{ for } n = 0, 1, 2, \dots, N-1. \quad (4)$$

where

$$\frac{X(k)}{N} = \begin{cases} F_k & \text{for } 0 \leq k \leq \frac{N-1}{2} \\ F_{k-N} & \text{for } \frac{N-1}{2} + 1 \leq k \leq N-1 \end{cases} \quad (5)$$

Let us define the vectors as follows.

$$\mathbf{x} = [x(0), x(1), x(2), \dots, x(N-1)]^T \quad (6)$$

$$\mathbf{w}_k = \left[1, e^{j \frac{2\pi}{N} k}, e^{j \frac{2\pi}{N} 2k}, \dots, e^{j \frac{2\pi}{N} (N-1)k} \right]^T \quad (7)$$

From equation (4), one can show that \mathbf{x} can be expressed in terms of a linear combination of \mathbf{w}_k 's so that

$$\mathbf{x} = \sum_{k=0}^{N-1} \frac{X(k)}{N} \mathbf{w}_k \quad (8)$$

Now $X(k)/N$ in equation (8) is the \mathbf{w}_k component of \mathbf{x} . The $X(k)/N$ can be computed by the projection of \mathbf{x} onto \mathbf{w}_k so that

$$\frac{X(k)}{N} = \frac{\mathbf{w}_k^* \mathbf{x}}{\mathbf{w}_k^* \mathbf{w}_k} = \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j \frac{2\pi}{N} nk} \quad (9)$$

where the superscript $*$ denotes the complex conjugation transpose. Equation (9) is known as the DFT formula:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j \frac{2\pi}{N} kn} \text{ for } k = 0, 1, 2, \dots, N-1. \quad (10)$$

By plugging $x(n)$ of equation (4) into equation (10), one obtains

$$X(k) = \sum_{n=0}^{N-1} \left[\frac{1}{N} \sum_{m=0}^{N-1} X(m) e^{j \frac{2\pi}{N} mn} \right] e^{-j \frac{2\pi}{N} kn} \quad (11)$$

If N samples are taken nonuniformly between 0 and NT , the expression of the nonuniformly sampled sequence becomes

$$\tilde{x}(n) = x_p((n + \alpha_n)T) = \sum_{k=-\frac{N-1}{2}}^{\frac{N-1}{2}} F_k e^{jk \frac{2\pi}{NT} (n + \alpha_n)T} \quad (12)$$

where α_n are termed the nonuniform sampling ratios. Equation (12) can be rewritten as

$$\tilde{x}(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j \frac{2\pi}{N} k(n + \alpha_n)} \quad (13)$$

By replacing n in the exponent inside the brackets in equation (11) with $n + \alpha_n$ or by plugging equation (13) into equation (10), the DFT of the nonuniformly sampled sequence is expressed as follows.

$$\tilde{X}(k) = \sum_{n=0}^{N-1} \left[\frac{1}{N} \sum_{m=0}^{N-1} X(m) e^{j \frac{2\pi}{N} m(n + \alpha_n)} \right] e^{-j \frac{2\pi}{N} kn} \quad (14)$$

The order of the summations in equation (14) can be reversed so that

$$\tilde{X}(k) = \sum_{m=0}^{N-1} \left[\frac{1}{N} \sum_{n=0}^{N-1} e^{j \frac{2\pi}{N} m \alpha_n} e^{-j \frac{2\pi}{N} (k-m)n} \right] X(m) \quad (15)$$

Reconstruction of uniform sampling from nonuniform sampling using the DFT is as follows. First, find the DFT, $\tilde{X}(k)$, of the nonuniformly sampled signal, $\tilde{x}(n)$. Second, estimate the DFT, $X(k)$, of the uniformly sampled signal using the relationship in equation (15). Finally, reconstruct $x(n)$ by taking the IDFT of the estimation of $X(k)$ [1].

When N is odd, for perfect reconstruction the highest harmonic of $x_p(t)$ should not be greater than $(N-1)/2$. When N is even, for perfect reconstruction the highest harmonic of $x_p(t)$ should not be greater than $N/2$. In other words, for perfect reconstruction using the DFT the bandwidth of $x_p(t)$ should be less than π/T [rad/s] (which is $N/2$ multiplied by $2\pi/NT$).

III. RECONSTRUCTION USING DCT

The following continuous-time signal is considered in this and the next sections.

$$x(t) = e^{-0.1t} \cos(0.2\pi t)u(t) \quad (16)$$

where $u(t)$ is the unit step function. The sampling interval $T = 1$ [sec] and the number of samples, $N = 8$.

As shown in the previous section, computation of the DFT of a sequence is in fact the same as computing the Fourier series coefficients of the periodically extended signal. The periodic extension is shown in Fig. 2 (a). Note that there is discontinuity at the junction of extension. This discontinuity or sudden jump unduly adds substantial high-frequency contents and hence increases the bandwidth of the extended signal.

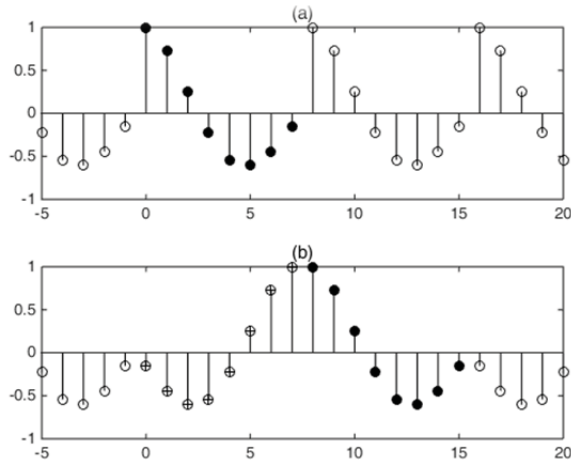


Fig. 2. Extension of $x(n) = e^{-0.1n} \cos(0.2\pi n)$ for $0 \leq n \leq 7$ using (a) periodic extension (b) half-sample symmetric extension followed by periodic extension.

To prevent such discontinuity, symmetric extension is used. It was shown that the half-sample symmetric extension performed the best for the reconstruction using the DFT [1]. The half-sample symmetric extension is shown in Fig. 2 (b). Instead of using the DFT of the symmetrically extended signal we propose to use the DCT for the reconstruction. The half-sample symmetric extension corresponds to the DCT-2 [2]. Suppose a continuous-time signal, $x(t)$, is sampled at $t = 0, T, 2T, \dots, (N-1)T$ where T is the sampling interval. The DCT of the uniformly sampled sequence, $x(n)$, for $n = 0, 1, 2, \dots, N-1$, is given by

$$X_{dct}(k) = \beta(k) \sum_{n=0}^{N-1} x(n) \cos\left(\frac{\pi(2n+1)k}{2N}\right) \quad (17)$$

where $\beta(0) = \sqrt{1/N}$ and $\beta(k) = \sqrt{2/N}$ for $k = 1, 2, \dots, N-1$.

The IDCT is given by

$$x(n) = \sum_{k=0}^{N-1} \beta(k) X_{dct}(k) \cos\left(\frac{\pi(2n+1)k}{2N}\right) \quad (18)$$

By plugging $x(n)$ of (18) into (17), one obtains

$$X_{dct}(k) = \beta(k) \sum_{n=0}^{N-1} \left[\sum_{m=0}^{N-1} \beta(m) X_{dct}(m) \cos\left(\frac{\pi(2n+1)m}{2N}\right) \right] \cos\left(\frac{\pi(2n+1)k}{2N}\right) \quad (19)$$

Suppose the signal is nonuniformly sampled so that the nonuniformly sampled sequence is given by (as in (12))

$$\tilde{x}(n) = x((n + \alpha_n)T) \text{ for } n = 0, 1, 2, \dots, N-1. \quad (20)$$

By replacing n inside the brackets in (19) with $n + \alpha_n$, the DCT of the nonuniformly sampled sequence is expressed as follows:

$$\tilde{X}_{dct}(k) = \beta(k) \sum_{n=0}^{N-1} \left[\sum_{m=0}^{N-1} \beta(m) X_{dct}(m) \cos\left(\frac{\pi(2n + 2\alpha_n + 1)m}{2N}\right) \right] \cos\left(\frac{\pi(2n+1)k}{2N}\right) \quad (21)$$

The order of the summations in (21) can be reversed so that

$$\tilde{X}_{dct}(k) = \sum_{m=0}^{N-1} \beta(m) \left[\beta(k) \sum_{n=0}^{N-1} \cos\left(\frac{\pi(2n + 2\alpha_n + 1)m}{2N}\right) \cos\left(\frac{\pi(2n+1)k}{2N}\right) \right] X_{dct}(m) \quad (22)$$

Let

$$r(n, m) = \cos\left(\frac{\pi(2n + 2\alpha_n + 1)m}{2N}\right) \cos\left(\frac{\pi(2n+1)k}{2N}\right) \text{ for } n = 0, 1, \dots, N-1, \quad (23)$$

then equation (22) becomes

$$\tilde{X}_{dct}(k) = \sum_{m=0}^{N-1} \beta(m) R(k, m) X_{dct}(m) \quad (24)$$

where the sequence $\{R(0, m), R(1, m), \dots, R(N-1, m)\}$ is the DCT of the sequence $\{r(0, m), r(1, m), \dots, r(N-1, m)\}$ for $m = 0, 1, \dots, N-1$.

In matrix form, equation (24) becomes

$$\tilde{\mathbf{X}}_{dct} = \mathbf{R} \mathbf{D} \mathbf{X}_{dct} \quad (25)$$

where

$$\mathbf{X}_{dct} = \begin{bmatrix} X_{dct}(0) \\ X_{dct}(1) \\ \vdots \\ X_{dct}(N-1) \end{bmatrix}, \quad \tilde{\mathbf{X}}_{dct} = \begin{bmatrix} \tilde{X}_{dct}(0) \\ \tilde{X}_{dct}(1) \\ \vdots \\ \tilde{X}_{dct}(N-1) \end{bmatrix},$$

$$\mathbf{R} = \begin{bmatrix} R(0,0) & R(0,1) & \cdots & R(0,N-1) \\ R(1,0) & R(1,1) & \cdots & R(1,N-1) \\ \vdots & \vdots & \ddots & \vdots \\ R(N-1,0) & R(N-1,1) & \cdots & R(N-1,N-1) \end{bmatrix}, \text{ and}$$

$$\mathbf{D} = \text{diag}(\beta(0), \beta(1), \dots, \beta(N-1)).$$

By performing the following matrix computation, the DCT of the uniformly sampled sequence can be reconstructed from the DCT of the non-uniformly sampled sequence.

$$\mathbf{X}_{dct} = (\mathbf{R}\mathbf{D})^{-1} \tilde{\mathbf{X}}_{dct} \quad (26)$$

IV. EXPERIMENTAL RESULTS

For our experiment, the signal of equation (16) is sampled nonuniformly. Statistically independent zero-mean Gaussian random numbers were used for nonuniform sampling ratios, α_n . The standard deviations were chosen as 0.01, 0.02, 0.04, 0.08, 0.16 and 0.32.

(i) Reconstruction using DFT without symmetric extension

The DFT of the nonuniformly sampled sequence is computed and the DFT of the uniformly sampled sequence is estimated using equation (15) [1]. The IDFT of the estimated DFT is computed for reconstruction of the uniformly sampled sequence. 5,000 trials were performed at each standard deviation. The average signal to noise ratio is computed using the following method.

$$\text{average SNR (in dB)} = 10 \log_{10} \frac{\text{signal power}}{\frac{1}{5000} \sum_{m=1}^{5000} [\text{noise power in each trial}]} \quad (27)$$

where signal power = $\frac{1}{N} \sum_{n=0}^{N-1} x^2(n)$ and

noise power in each trial = $\frac{1}{N} \sum_{n=0}^{N-1} [x(n) - \hat{x}(n)]^2$.

where $\hat{x}(n)$ is the reconstructed uniformly sampled sequence by taking the IDFT of the estimated DFT, $X(k)$, obtained according to the reconstruction algorithm [1] in each trial.

(ii) Reconstruction using DCT

The DCT of the nonuniformly sampled sequence is computed and the DCT of the uniformly sampled sequence is estimated using equation (26).

TABLE I

Comparison of performance between the DFT method without symmetric extension and the DCT method. 5000 trials were performed at each standard deviation.

Standard deviation of nonuniform sampling ratios	Average SNR [dB]	
	DFT method (without extension)	DCT method
0.01	44.0675 dB	64.4223 dB
0.02	37.9453 dB	58.5197 dB
0.04	31.8171 dB	52.3820 dB
0.08	25.4370 dB	46.2497 dB
0.16	17.7753 dB	39.7371 dB
0.32	too much error	31.6782 dB

The IDCT of the estimated DCT is computed for the reconstruction of the uniformly sampled sequence. The DCT method performed much better as shown in Table I.

There is at least 20-dB advantage in SNR with the DCT method over the DFT method without symmetric extension.

V. CONCLUSION

In this paper reconstruction of a uniformly sampled sequence from a nonuniformly sampled transient sequence using the DCT is described. The DCT method that does not need doubling of the sequence length and complex operations requires less computational complexity and is more desirable than the method based on DFT with symmetric extension.

VI. REFERENCES

- [1] S. Park, W. Hao and C. S. Leung, "Reconstruction of uniformly sampled sequence from nonuniformly sampled transient sequence using symmetric extension," *IEEE Trans. on Signal Processing*, vol. 60, no. 3, pp.1498-1451, March 2012.
- [2] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing Third Edition*, Pearson, Upper Saddle River, NJ, 2010.

Numerical Comparison of the Performance of Submerged Entry Nozzles for Slab Continuous Casting

Carlos A. Hernandez¹, Raul Miranda², Miguel A. Barron¹

¹Departamento de Materiales, Universidad Autonoma Metropolitana Azcapotzalco, Mexico City, Mexico

Abstract - Two new geometrical designs for submerged entry nozzles for slab continuous casting of molten steel are presented here. Their performances are compared with that of a conventional cylindrical nozzle using Computational Fluid Dynamics software. Topography of the free surface, velocity vectors and turbulence intensity were employed as variables to evaluate their performance in terms of chance of powder entrapment. Numerical results show that the conventional submerged entry nozzle exhibits the poor performance.

Keywords: Computational fluid dynamics, continuous casting, fluid flow, mold flow, submerged entry nozzle.

1 Introduction

More than 95% of the world raw steel production is currently cast by means of the continuous casting process [1]. Fluid flow in the continuous casting mold greatly influences the quality of the slabs of cast steel [2], and among several factors, the geometry of the submerged entry nozzle (SEN) is the most influential one [3,4]. In the continuous casting mold, mold powder is added in order to avoid heat losses, provide lubrication and protect the molten steel from oxidation with the surrounding air. Unfortunately, the mold powder sometimes is entrapped into the molten steel and such entrapment alters in a significant way the quality of steel [5].

An improper design of the SEN promotes the formation of Karman's vortices on the meniscus of the mold, and these vortices may cause powder entrapment [5-7]. Then, design of SEN is focused to minimize instabilities of the free surface to prevent the above phenomenon. In [8] a numerical design of an elliptic SEN is proposed which supposedly reduces the pressure difference around the nozzle avoiding powder entrapment. In [9] the dynamic behavior of the flow inside a bifurcated SEN is physically and numerically studied. The authors report that the flow pattern inside the SEN is periodic, and its frequencies are determined from the power spectrum of the generated time series. Effect of the configuration parameters of SEN in the flow field in a slab mold is analyzed in [10] using of water modeling experiments. It is reported that by increasing the SEN outlet area a proper flow field is obtained. In [11] the design of a swirling-flow SEN is proposed, and its performance is evaluated in a water model.

This SEN increases the casting speed and improves the surface

quality of steel. Flow optimization in the mold by port design improvement in a SEN is analyzed in [12].

In this work two new geometrical designs of SEN are proposed. Their performances are evaluated through mathematical modeling using transient 3D numerical simulations using Computational Fluid Dynamics (CFD) software. Besides, the performance of the new proposed devices is compared with that of a conventional SEN.

2. Design of the Submerged Entry Nozzles

Fig. 1 shows the geometries of the considered two-port SEN. The first one (Fig. 1a) corresponds to a conventional cylindrical SEN (hereinafter called simple), the second one (Fig. 1b) corresponds to a SEN with an oval plate located in its upper part (hereinafter called plate), and finally, an anchor-shaped SEN (hereinafter called anchor). The simple and the anchor SEN are new, and their performances will be compared with that of the simple SEN. The SEN dimensions are: 0.5 m high, 0.06 m port diameter, 0.04 m port diameter, plate thickness 0.02 m.

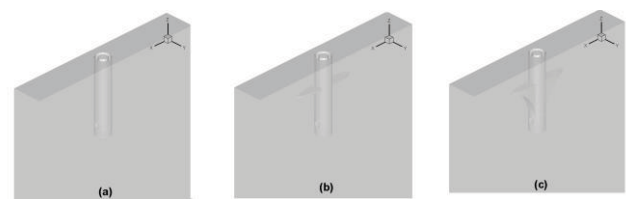


Fig. 1. The three SEN considered.

3. Mathematical Model and Numerical Solution

The flow of an isothermal incompressible Newtonian fluid and the mass conservation are represented by the Navier-Stokes equations and the continuity equation, respectively [13]. Turbulence in the mold is simulated by means of the classical two equations K- ϵ model [14, 15] given that this model yields more numerical stability during the integration for long times. The multiphase nature of the mold flow is simulated by means of the Volume of Fluid (VOF) model

[16], which considers that all the present phases share the same flow field. The mass conservation principle forces that the whole of the phase volume fractions sums the unity.

The mold and SEN mathematical model was solved using commercial CFD software, using the mesh shown in Fig.2, which consists of 464 000 tetrahedral/hybrid cells.

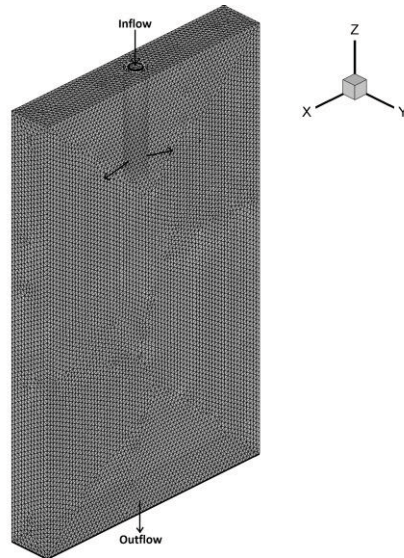


Fig. 2. The continuous casting mold and its computational mesh.

Transient 3-D two-phase (air, molten steel) isothermal computer simulations were carried out using time steps of 0.001 s. Runs of 90 s were made to guarantee a well developed flow in the mold. The mold dimensions were 2 m high, 0.6 m wide and 0.2 m thick. The initial level of molten steel was 1.8 m, the remaining was air. The physical properties of molten steel were: density 7100 kg/m³, viscosity 00067 kg/(m.s). Boundary conditions were as follows: velocity inlet 2.83 m/s, which corresponds to a casting speed of 1.5 m/min; turbulent kinetic energy 0.08 m²/s², turbulent dissipation rate 0.755 m²/s³. The PISO (pressure implicit with split operator) algorithm was employed for pressure-velocity coupling.

4. Results and Discussion

Results of computer simulations are presented for 90 s of elapsed time since the beginning of the casting. In the left side of Fig. 3 are shown the phase distribution and the free surface for the simple, plate and anchor SEN, respectively, whereas in the right side of the same Figure are depicted the velocity vectors for the $y=0$ vertical plane. The simple SEN exhibits the deepest depression of the free surface and the largest velocity vectors in the upper section of the mold.

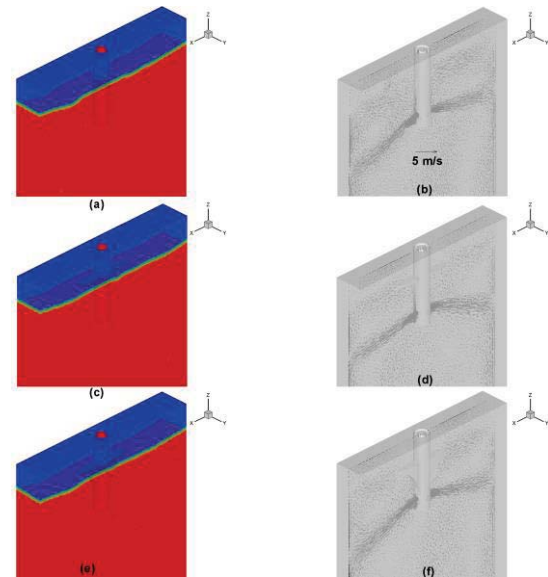


Fig. 3. Free surface (left) and velocity vectors (right) for the (a,b) simple, (c,d) plate and (e,f) anchor SEN, respectively.

In Fig. 4 are shown the velocity vectors in two planes of the three considered SEN. Figs. 4 (a,c,d) depict the plane $x=0.08$ for the simple, plate and anchor SEN, respectively, and Figs. 4 (b,d,e) depict the plane $z=0.8$ for the same nozzles. These figures show that the velocity vectors near the free surface in the vicinity of the SEN are greater for the simple SEN than that corresponding to the plate and anchor SEN. Given that the velocity field near the upper sections of the SEN determines the powder entrapment, these results suggest that the simple SEN yields the largest powder entrapment among the SEN considered.

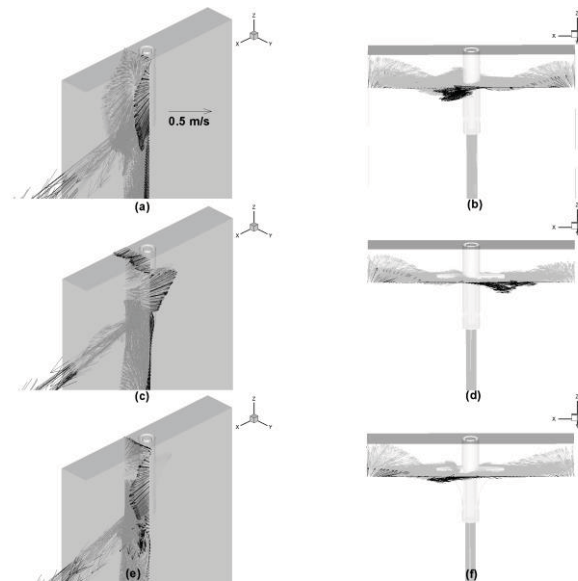


Fig. 4. Velocity vectors for the $x=0.08$ plane (left) and the $z=0.8$ plane (right) for the (a,b) simple, (c,d) plate and (e,f) anchor SEN, respectively.

Finally, in Fig. 5(a-c) are shown the turbulence intensity along a line in the plane $z=0.8$, $y=0$, located 0.02 m below the free surface. Plate SEN presents the smallest values of the turbulence intensity; this means that the chance for powder entrapment becomes small for this SEN.

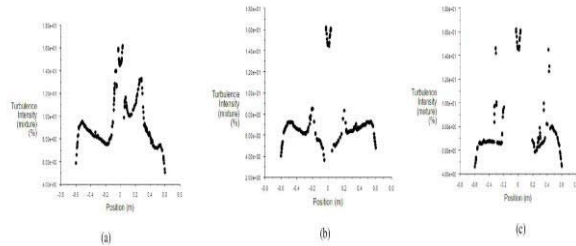


Fig. 5. Turbulence intensity along a line in the plane $z=0.8$, $y=0$, for (a) simple, (b) plate and (c) anchor SEN.

5. Conclusions

Performances of three submerged entry nozzles with different geometrical designs were numerically compared in terms of the velocity vectors and turbulence intensity distribution near the free surface. From the presented results, the following conclusions arise:

- (i) The simple SEN exhibits the deepest depression of the free surface and the largest velocity vectors in the upper section of the mold.
- (ii) The velocity vectors near the free surface in the vicinity of the SEN are greater for the simple SEN than that corresponding to the plate and anchor SEN.
- (iii) Plate SEN presents the smallest values of the turbulence intensity; this means that the chance for powder entrapment becomes small for this SEN.

6. References

- [1]. W. Slezak, M. Korolczuk, P. Migas. High temperature rheometric measurements of mold powder. *Archives of Metallurgy and Materials* 60 (2015) 289-294.
- [2]. A.W. Cramb. *The Making, Shaping and Treating of Steel. Continuous Casting Volume*. AISE Steel Foundation, Pittsburgh, PA, 2003.
- [3]. B.G. Thomas. Modeling of continuous-casting defects related to mold fluid flow. *3rd International Congress on Science & Technology of Steelmaking*, AIST, Charlotte, NC, 2005.
- [4]. S. Garcia-Hernandez, R.D. Morales, J.J. Barreto, K. Morales-Higa. Numerical optimization of nozzle ports to improve the fluidynamics by controlling backflow in a continuous casting slab mold. *ISIJ International* 53 (2013) 1794-1802.
- [5]. M. Iguchi, J. Yoshida, T. Shimizu, Y. Mizuno. Model study on the entrapment of mold powder into molten steel. *ISIJ International*, 40 (2000) 685-691.
- [6]. M. Iguchi, O.J. Ilegbusi. *Modeling Multiphase Materials Processes: Gas Liquid Systems*. Springer, New York, NY, 2011.
- [7]. K. Tsutsumi, K. Watanabe, M. Suzuki, M. Nakada, T. Shiomi. Effect of properties of mold powder entrapped in molten steel in a continuous casting process. *VII International Conference on Molten Slags Fluxes and Salts*, The South African Institute of Mining and Metallurgy, Johannesburg, South Africa, 2004.
- [8]. Y. Ueda, T. Kida, M. Iguchi. Unsteady pressure coefficient around an elliptic immersion nozzle. *ISIJ International* 44 (2004) 1403-1409.
- [9]. C. Real, R. Miranda, C. Vilchis, M.A. Barron, L. Hoyos, J. Gonzalez. Transient internal flow characterization of a bifurcated submerged entry nozzle. *ISIJ International* 46 (2006) 1183-1191.
- [10]. Z. Liang, N. Wang, Z. Zou, A. Yu. Optimization of submerged entry nozzle of continuous casting slab. *Proceedings of the 2009 TMS Annual Meeting and Exhibition*, San Francisco, CA, February 16- February 19, 2009, pp. 569-574.
- [11]. Y. Tsukaguchi, H. Hayashi, H. Kurimoto, S. Yokoya, K. Marukawa, T. Tanaka. Development of swirling-flow submerged entry nozzle for slab casting. *ISIJ International* 50 (2010) 721-729.
- [12]. T. Kuroda, A. Mizobe, J. Kurisu. Flow optimization in the mould by port design improvement of submerged entry nozzle. *Proceedings UNITECR 2011 Congress: 12th Biennial Worldwide Conference on Refractories - Refractories-Technology to Sustain the Global Environment*. The American Ceramic Society, October 30 - November 2, 2011, Kyoto, Japan, pp. 434-437.
- [13]. R.B. Bird, W.E. Stewart, E.N. Lightfoot. *Transport Phenomena*. Wiley, Second Edition, New York, NY, 2002.
- [14]. Thomas, B., Yuan, Q., Sivaramakhrisnan, S., Shi, T., Vanka, S.P., Assar, M.B. Comparison of four methods to evaluate fluid velocities in a continuous slab casting mold. *ISIJ International*, 41 (2001) 1262-1271.
- [15]. Solorio-Díaz, G., Morales, R.D., Palafox-Ramos, J., García-Demedices, L., Ramos-Banderas, A. Analysis of fluid flow turbulence in tundishes fed by a swirling ladle shroud. *ISIJ International*, 44 (2004) 1024-1032.

Fluent 6.1 User's Guide. Lebanon, NH, 2003.

[16. C.W]. Hirt, B.D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39 (1981) 201-225.

SESSION
MODELING AND NOVEL APPLICATIONS

Chair(s)

TBA

The Effect of Changing Search Patterns in an Agent-Based Model

D. Q. Quach*, D. P. Playne and C. J. Scogings

Institute of Natural and Mathematical Sciences, Massey University

Albany, North Shore 102-904, Auckland, New Zealand

Email: dara.quach@gmail.com, {d.p.playne, c.scogings} @massey.ac.nz

Tel: +64 9 414 0800 Fax: +64 9 441 8181

Regular Research Paper

ABSTRACT

Many agent-based models employ various mechanisms for agents to propagate. We explore different searching patterns influenced by distance and resource abundance. We find that when higher density of food sources is more influential than distance, the system moves towards higher sustainable population levels for both the predator and prey species. The resource influenced search patterns produce less defined defensive spiral patterned structures than distance driven searching.

KEYWORDS

Agent-Based Model; Animat; Search Patterns; Population Dynamics

1 Introduction

Computational agent-based models are developed to gain insight into emergence of complex behaviours from multiple interacting individuals following simple rules. These models have been used in a range of domains such as Economics [1, 2], History [3], Geography [4], Political sciences [5] and Ecology [6]. Extensively applied in the latter field, models have been implemented to examine complex emergent collective behaviours such as flocking [7, 8] and animat communication [9, 10]. Artificial animals termed animats have a range of publications [11–15] in the field of artificial life furthering establishment and providing new insight into spatial agent-based models.

The extended model for this work is based on a well-established predator-prey implementation [16] and reproduces the population dynamics of boom-bust cycles shown in predator prey models such as the Lotka-Volterra equations [17]. The model builds upon simple predator-prey survival behaviours where both species breed to increase their population and grow by grazing resources or hunting prey.

Prey are prone to being predated and both species are susceptible to starvation and old age.

The spatial aspect of the animat system is based on prioritised movement rules of the model such as flocking where species swarm towards food or diverge away from threats [18].

Other research applies defined approaches of simple individual-based movements [19, 20] and generative mechanisms [20]. Factors such as state and environment can play a role in animats movements [21, 22]. These factors can also be prioritised or combined differently depending on the purpose of the model.

In this work we introduce various searching approaches specific to our model for animats to decide upon target or threat agents required to move towards or away from. The approaches are based upon the factors of distance and abundance of resources. Two groups of search patterns are developed that give priority to each of these different factors.

An overview for the animat model this work is based upon is presented in Section 2. The implementations for distance driven search patterns A, B and C and patterns D and E where the prioritised factor is resource abundance are described in Section 3. Section 4 presents results of the different approaches and a discussion is provided in Section 5. Conclusions and future work is offered in Section 6.

2 Predator-Prey Animat Model

The well established predator-prey model used in this work is based on individual animats that adhere to spatial state machine mechanics [23]. The predator/prey agents or animats carry simulation stepped state information including: health, age and co-ordinate locations on a 2-dimensional grid. Earlier work shows using a grid combined with

Table 1: Priority rule-set

Predator Rules:	Condition:	Prey Rules:	Condition:
seek mate	health greater than 50%	breed	health greater than 30% and mate adjacent
seek prey	health less than 50%	graze grass	health less than 70%
consume prey	health less than 50% and prey adjacent	move randomly	50% chance success
breed	health greater than 50% and mate adjacent	seek mate	health greater than 50%
randomly move	50% chance success	move away from predators	predator adjacent

a square bounded grass location for animats to occupy [24] does not affect the emergent behaviours of the model [25]. This bounding approach enables the population to be manageable while exhibiting higher frequencies of repetitive boom bust cycles which are well-known from Lotka-Volterra models [17].

The animat model uses a two-phase system-update approach. In the first phase all animats are updated in a random sequence in which an arbitrary agent executes a specific behaviour based on the current state of itself and the system. The second phase involves updating data for agents that are changed from the first phase.

Table 1 lists the rules for each species in order from top to bottom. The animat traverses through each rule in order and attempts to execute the associated action based mainly on its current health. The animat executes no other actions if it has been already successful in performing another action.

The health of an animat determines its survival to the next system state, thus seeking prey, eating and grazing all contribute towards a healthier state for the animat. The grass grazed by prey has a fixed nutrition value that increases health towards a capped maximum value which cannot be exceeded.

Each species of animats has a radial vision area of the system based on their position. This surrounding information can be used to seek prey or a mate, the former applies to predators only as the food source of the prey is abundant throughout the system.

3 Search Patterns

Several rules from the animats rule-set require information from their spatial surroundings [26]. The *seek mate* and *seek prey* rules both require an animat to search the surrounding environment to identify animats of a specific

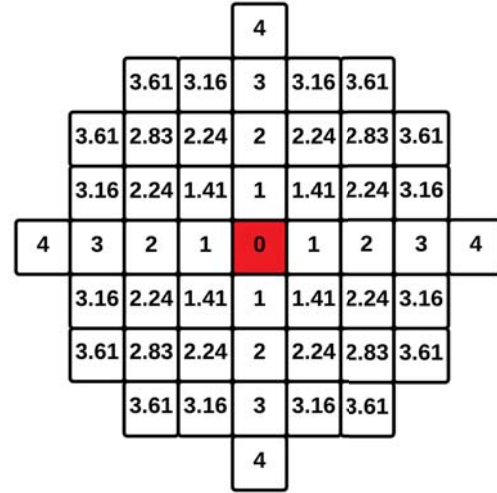


Figure 1: Equal distance search cells from centre to 4 units away

species.

Each species of animat has a specific range of vision that determines which cells can be seen. In the standard animat model, the animats will search for targets by examining surrounding cells in order of euclidian distance from closest to furthest (see Figure 1). The animats model utilises a radial look-up table that stores spatial cells sorted by distance, allowing optimal traversal to find the “nearest neighbour” [16]. The look up table can be created using a two step process.

1. Process coordinates in range of radius with mid point as 0.
2. Sort coordinates with closest coordinates first to furthest last.

Inspecting each traversed cell of the look up table in turn allows the closest target animat to be located. This can be achieved by adding the lookup table offsets (dx, dy) to the animats current location (x, y) and inspecting the grid at the calculated position.

Alternate search patterns are investigated to explore various ways an animat might search for targets or threats to move towards or away from. Previous studies of the animat models utilises a search for nearest target strategy [27], allowing animats to move in the direction of the closest target at each simulation step.

The approach of identifying the nearest target means examining each cell in a specific order derived from euclidian distances. On the two dimensional environment of the animats model exists groups of cells that are equal in euclidian distance from the animat’s cell. The possibility that valid targets occupy more than 1 cell in an equal distance group. Each of these targets are equally valid which means that in

a simulation step an animat may have a choice of which direction to move in.

The first three search patterns in this work examine the different order of cells an animat explores within an equal distance group. In search pattern A the animats have a fixed favoured pattern in which they search for their nearest targets, the order is fixed as left, top, bottom then right.

In search pattern B the order is random within equal distance groups. In this modification if there is more than one occupied cell in the equal distance group, the occupying targets or threats will be chosen randomly. Pattern B can be implemented using Algorithm 1.

Algorithm 1 Algorithm for random search patterns with equal distance groups (Pattern B)

```

for nearest cells to cells at the end of agent vision do
  get all cells equal in distance to current cell;
  if more than one cell with one or more valid target agents then
    randomly choose a cell with one or more valid target agents
    get cell information and exit loop;
  end if
  if only one cell with one or more valid target agents then
    pick the only cell with valid target agent or agents;
    exit loop;
  else
    continue to the next group of equal distance cells;
  end if
end for

```

In search pattern C, the animats are set to traverse through each equal distance group and compare the density of occupying target agents for each cell. The direction of the cell with the most targets or threat agents is chosen to move towards or away from. Algorithm 2 shows how this pattern can be implemented.

Algorithm 2 Algorithm to search for the nearest cell with the most target agents (Pattern C)

```

for nearest cells to cells at the end of agent vision do
  get all cells equal in distance to current cell;
  if no cells contain valid target agent or agents then
    continue loop;
  else
    choose the cell with the most valid target agents;
    get cell information and exit loop;
  end if
end for

```

As agents retain information of their surroundings, observed cells can be examined to determine propagation directions

based on the number of occupying target animats. Algorithm 2 uses the greatest number of target agents as the condition to choose a propagation direction. In the event of a tie where multiple cells have the same (greatest) number of maximum agents, one cell will be selected at random. This enables the animat to move towards the direction of the closest cell with the greatest number of target agents.

In the following two approaches, an animat will take into consideration all cells within its vision range. This effectively allows an agent to explore information for distant cells and choose a target cell based on a certain condition.

For search pattern D, an animat will disregard distance to explore all cells that are within the range of its vision and choose a cell to move towards or away from. As the agent traverses through each valid cell for information regarding the target or threat cell, it keeps track of the location of the cell with the most threatening or pursuable agents. The possibility of multiple cells with an equal number of a specific species can be solved by randomly selecting one of these cells.

Algorithm 3 shows how to determine the movement direction by finding the cell with the highest target density.

Algorithm 3 Algorithm to search for the cell with the most target agents within vision (Pattern D)

```

for all cells in vision of animat do
  keep track of cell or cells with the most valid target agents;
end for
if more than one cell with the most valid agents then
  randomly choose a cell and get cell information;
end if
if only one cell with the most valid target agents; then
  get cell information;
else
  no cells found with valid target agents;
end if

```

In search pattern E, a direction for the animat to move is randomly chosen based on the distribution of probabilities for each cell in range of vision. The probability distribution for each cell can be determined based on the number of occupying animats relative to the distance to that cell (see Equation 1).

$$P_k = (N_k/R_k) / \sum_{i=0}^n (N_i/R_i) \quad (1)$$

where N_k is the number of agents at cell k and R_k is the 2-dimensional euclidian distance to cell k . The probabilities calculated (P_k) will be in range of [0.0 - 1.0] and a random

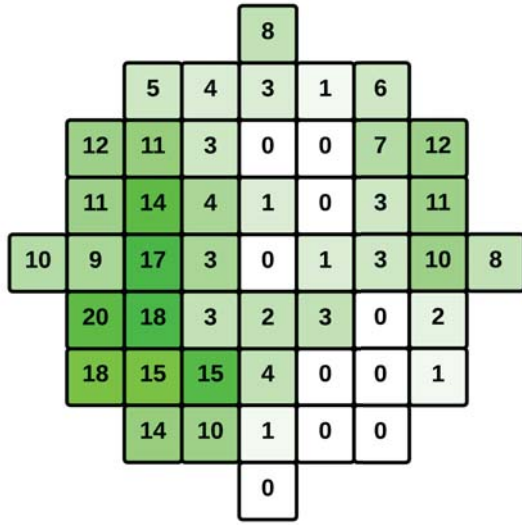


Figure 2: Example of search pattern E, where higher tints of green represent a higher probability of being selected and the value indicate the number of occupying agents

decimal number is used to choose a cell.

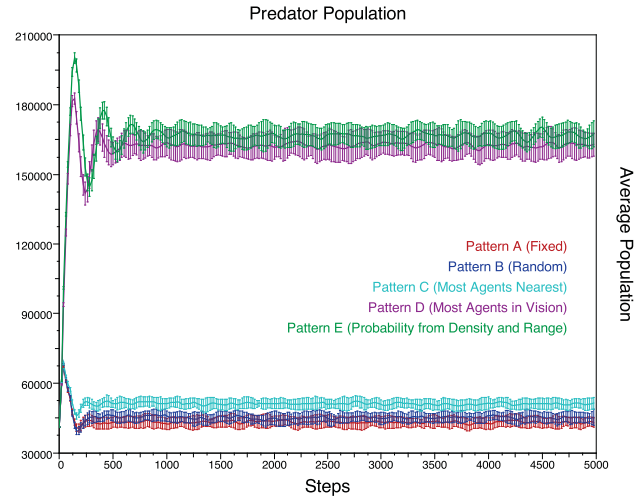
An animat may give higher importance to the number of target animats relative to distance by scouting all possible cells in vision occupied by target agents. The formula above is an approach to determine the propagation direction based on different probabilities formulated from the number of occupying targets and the distance to that cell. Algorithm 4 shows how we can use the density and distance of an examined cell to calculate probabilities to choose a cell for determining movements and an example is provided in Figure 2.

Algorithm 4 Algorithm to choose a cell based on probabilities (Pattern E)

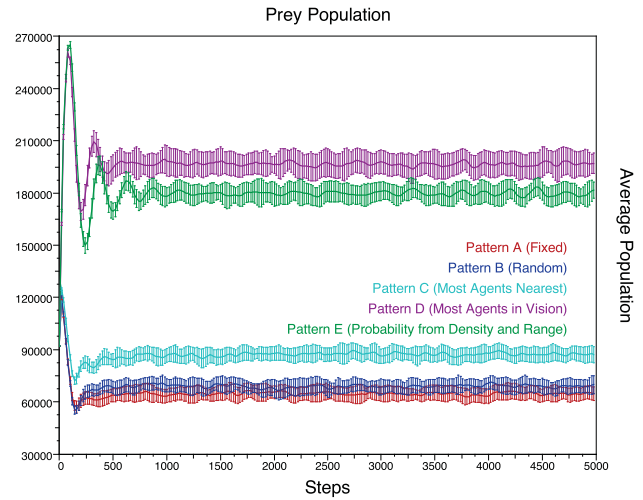
```

for all cells in vision of animat do
  if cell contains one or more valid target agents then
    get number of valid target agents;
    get distance to examined cell;
    calculate probability using equation from search pattern E;
    keep track of cells probability of being chosen;
  end if
end for
if one or more valid cells then
  randomly pick a cell based on its probability of being chosen;
  get cell information;
else
  no valid agents found in vision;
end if

```



(a) Predator population average from 0 - 5000 steps



(b) Prey population average from 0 - 5000 steps

Figure 3: Comparison of the average predator and prey populations using the search patterns A-E.

The search patterns investigated in this work are possible approaches a system can define for animats to examine their environment. The search patterns prioritise factors differently, pattern A and B factor distance as most important while pattern C combines valid target animat density with distance but prioritises distance over density. Search Patterns D uses valid target animat density as the main factor and pattern E combines both valid target animat density with distance but prioritise density over distance.

4 Simulation Results

Figure 3 shows the average populations of each search pattern over five thousand steps, results are averaged over thirty simulation runs. The results show that the different search

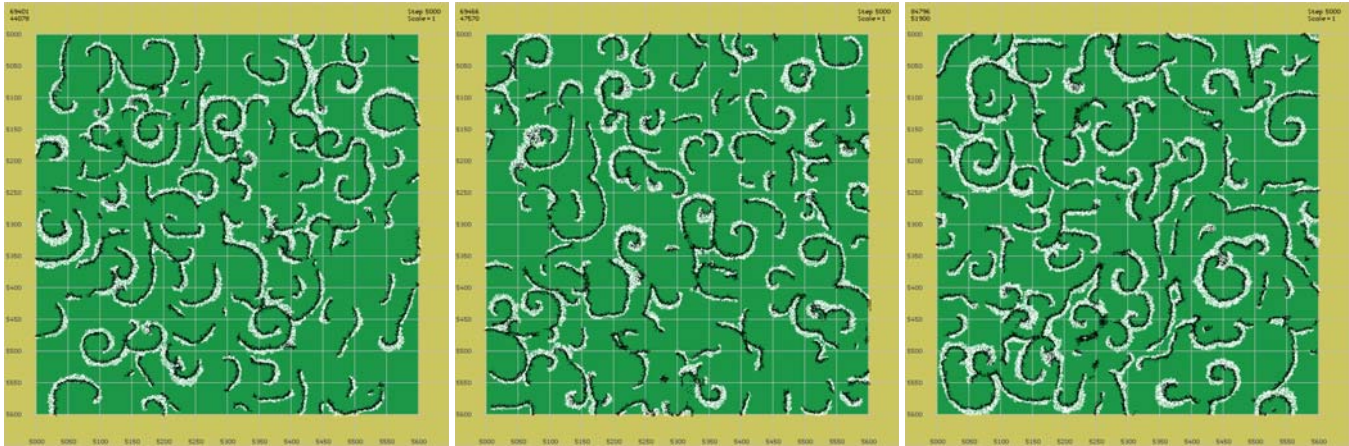


Figure 4: Nearest search patterns simulation screen capture series, from left to right pattern A, B and C

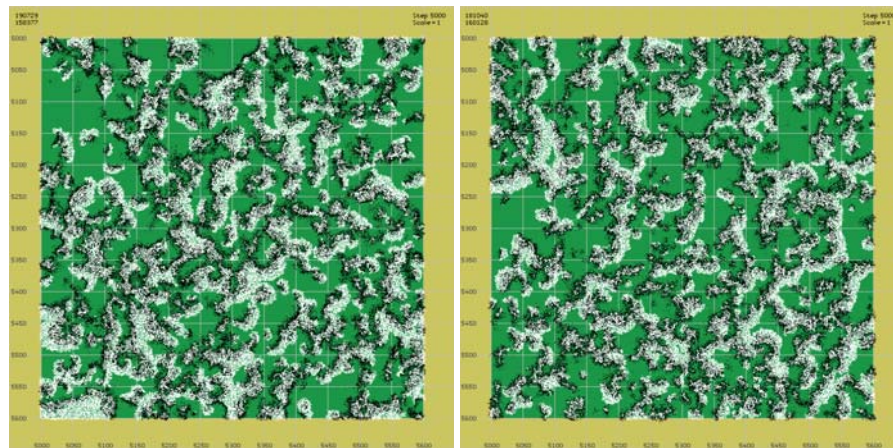


Figure 5: Ranged search patterns simulation screen capture series for search pattern D and E

patterns have a similar effect on both the predator and prey populations. This relationship is expected as a greater abundance of prey provides a larger food source for the predators and thus a higher sustainable population.

The original fixed-order nearest search pattern A has the lowest average population, randomising the nearest search order (B) slightly raises this average population. Preferring the greatest number of nearest agents (C) makes a greater difference, the average population is again increased. The patterns that prioritise the number of agents over the distance (D-E) show a much larger effect, these patterns showed an increase of approximately $2\times$ for the prey and $3\times$ for the predator populations over the distance focused patterns.

Figures 4 and 5 show snapshots of simulations using the search patterns A-E after five thousands steps. These snapshots show a difference in the spatial structures formed by the agents that prioritise distance (A-C) and those that prioritise the number of targets (D-E). While the agents that use patterns A-C show the tight defensive spiral structures

seen in previous work, the agents using patterns D-E show a greater dispersion of agents.

5 Discussion

We have investigated several alternatives for animats search patterns and how they affect the overall behaviour of the system. These search patterns affect both the spatial structures formed by the animats as well as the sustainable populations of the predator/prey species. Search patterns that prioritise distance exhibit tight spiral structures seen in previous work [28]. Prioritising the number of potential targets over the distance had a significant effect on both the structure of these spirals and the sustainable populations of both species.

Predators that always target the closest animat tend to eliminate all prey in an area before moving on. When these predators form the spiral structures seen in Figure 4, a moving ‘wavefront’ of predators forms that sweeps through an area

consuming all the available prey. The prey must flee to stay ahead of the predators in order to survive.

Alternatively when predators target the areas with the greatest number of targets, they can ignore cells with only a few agents and instead chase a larger herd of prey. This can allow a small number of prey to survive the predator 'wave-front' and continue to reproduce, replenishing the prey population in that area. The individual predator behaviour of greedily targeting a larger group of prey has the effect of allowing small groups of prey to escape. Overall this is beneficial to both the predator and the prey species, as previously mentioned a higher sustainable population of prey benefits the predators by providing them with a more abundant food source.

Population sustainability is an important effect even for simplified models such as the animats. Rule choices and microscopic behaviours can have a significant effect on the populations and can cause one or both of the species to die out. If the prey die out then the predators follow as they have no food source while if the predators die out then the prey will grow exponentially. Search patterns that provide higher sustainable population levels could provide a more robust model that could support a wider range of behaviours without suffering from species extinction. Investigations into such models and rule-sets are left to future work.

6 Conclusions

Five search patterns have been described for the animats predator-prey model that assign different priority to distance and target abundance. These search patterns are shown to have a significant effect on both the spatial structure the animats form and the sustainable population of both species.

Agents that prioritise target abundance over distance show a great spatial dispersion and higher overall population levels. These patterns (D-E) showed a $2\times$ increase for the prey and $3\times$ increase for the predator populations. Increasing the sustainable populations is an important effect for systems that may suffer from species extinction.

Future work includes exploring other options for searching patterns and interaction of other rule-sets with searching. Investigation may also include exploring evolution of searching patterns or parameters.

References

- [1] Samanidou, E., Zschischang, E., Stauffer, D., Lux, T.: Agent-based models of financial markets. *Reports on Progress in Physics* **70** (2007) 409
- [2] Chen, S.H., Chang, C.L., Du, Y.R.: Agent-based economic models and econometrics. *The Knowledge Engineering Review* **27** (2012) 187–219
- [3] Scogings, C.J., Hawick, K.A.: An Agent-Based Model of the Battle of Isandlwana. In: *Proc. 2012 Winter Simulation Conference*. Number CSTN-116, Berlin, Germany, WSC (2012) ISBN: 978-1-4673-4780-8.
- [4] Crooks, A.T., Castle, C.J.: The integration of agent-based modelling and geographical information for geospatial simulation. In: *Agent-based models of geographical systems*. Springer (2012) 219–251
- [5] Cioffi-Revilla, C., Rouleau, M.: Mason rebeland: An agent-based model of politics, environment, and insurgency. *International Studies Review* **12** (2010) 31–52
- [6] McLane, A.J., Semeniuk, C., McDermid, G.J., Marceau, D.J.: The role of agent-based models in wildlife ecology and management. *Ecological Modelling* **222** (2011) 1544–1556
- [7] Stonedahl, F., Wilensky, U.: Finding forms of flocking: Evolutionary search in abm parameter-spaces. In: *Multi-Agent-Based Simulation XI*. Springer (2010) 61–75
- [8] Fine, B.T., Shell, D.A.: Examining the information requirements for flocking motion. In: *From Animals to Animats 12 - Proc. 12th Int. Conf. on Simulation of Adaptive Behaviours (SAB2012)*. Number 7426 in LNAI, Odense, Denmark, Springer (2012) 442–452
- [9] Scogings, C.J., Hawick, K.A.: Cross-caste communication in a multi-agent predator-prey model. In: *Proc. Int. Conf. on Artificial Life and Applications (AIA 2011)*, Innsbruck, Austria, IASTED (2011) 163–170
- [10] Scogings, C.J., Hawick, K.A.: An investigation into the effects of sentinels on animat collectives. In: *Proc. Nineteenth IASTED Int. Conf on Applied Simulation and Modelling (ASM 2011)*. Number 715-095, Crete, Greece, IASTED (2011) 221–226 CSTN-121.
- [11] Holland, J.H.: Echoing emergence: Objectives, rough definitions, and speculations for echo-class models. In Cowan, G.A., Pines, D., Meltzer, D., eds.: *Complexity: Metaphors, Models and Reality*. Addison-Wesley, Reading, MA (1994) 309–342
- [12] Adami, C.: On modeling life. In Brooks, R., Maes, P., eds.: *Proc. Artificial Life IV*, MIT Press (1994) 269–274
- [13] Tyrrell, T., Mayhew, J.E.W.: Computer simulation of an animal environment. In Meyer, J.A., Wilson, S.W., eds.: *From Animals to Animats, Proceedings of the First International Conference on Simulation of Adaptive Behavior*. (1991) 263–272
- [14] Yaeger, L.: Computational genetice, physiology, metabolism, neural systems, learning, vision and behavior or polyworld: Life in a new context. In Langton, C., ed.: *Proc Artificial Life III Conference*. (1994)
- [15] Ray, T.: An approach to the synthesis of life. *Artificial Life II*, Santa Fe Institute Studies in the Sciences of Complexity **xi** (1991) 371–408
- [16] Scogings, C.J., Hawick, K.A., James, H.A.: Tools and techniques for optimisation of microscopic artificial life simulation models. In Nyongesa, H., ed.: *Proceedings of the Sixth IASTED International Conference on Modelling, Simulation, and Optimization*, Gabarone, Botswana, IASTED (2006) 90–95
- [17] Lotka, A.J.: *Elements of Physical Biology*. Williams & Williams, Baltimore (1925)
- [18] Scogings, C.J., Hawick, K.A.: Emergent system effects from

- microscopic evasion choices in a predator-prey simulation. In: Proc. 10th International Conference on Genetic and Evolutionary Methods (GEM'13). Number CSTN-188, Las Vegas, USA, WorldComp (2013) GEM3895
- [19] Turchin, P.: Quantitative analysis of movement: measuring and modeling population redistribution in animals and plants. Volume 1. (1998)
 - [20] Latombe, G., Parrott, L., Basille, M., Fortin, D.: Uniting statistical and individual-based approaches for animal movement modelling. *PloS one* **9** (2014) e99938
 - [21] Tang, W., Bennett, D.A.: Agent-based modeling of animal movement: A review. *Geography Compass* **4** (2010) 682–700
 - [22] Nathan, R., Getz, W.M., Revilla, E., Holyoak, M., Kadmon, R., Saltz, D., Smouse, P.E.: A movement ecology paradigm for unifying organismal movement research. *Proceedings of the National Academy of Sciences* **105** (2008) 19052–19059
 - [23] Holland, J.H.: Hidden order: How adaptation builds complexity. Addison-Wesley, Reading, MA (1995)
 - [24] Scogings, C.J., Hawick, K.A.: Global constraints and diffusion in a localised animat agent model. In: Proc. IASTED Int. Conf. on Applied Simulation and Modelling, Corfu, Greece, IASTED (2008) 14–19
 - [25] Hawick, K.A., Scogings, C.J.: Complex Emergent Behaviour from Evolutionary Spatial Animat Agents. Number ISBN 978-3-642-13424-1. In: Agent-Based Evolutionary Search. Springer (2010) 139–160 CSTN-067.
 - [26] James, H.A., Scogings, C.J., Hawick, K.A.: A framework and simulation engine for studying artificial life. *Research Letters in the Information and Mathematical Sciences* **6** (2004) 143–155
 - [27] Scogings, C.J.: Towards an agent-based simulation of predators developing a search image. In: Proc. International Conference on Modelling, Simulation and Visualization Methods (MSV 2015), Las Vegas, USA, CSREA (2015) 55–61
 - [28] Hawick, K.A., Scogings, C.J., James, H.A.: Defensive spiral emergence in a predator-prey model. *Complexity International* **12** (2008) 1–10 ISSN 1320-0682.

System Dynamics as a Tool for Modeling Application Layer Cyber Security

Uma Kannan¹, Rajendran Swamidurai², and David Umphress¹

¹Computer Science and Software Engineering, Auburn University, Auburn, AL, USA

²Mathematics and Computer Science, Alabama State University, Montgomery, AL, USA

Abstract - System dynamics (SD) is a methodology used to understand how systems change over time. In the 1960s, the SD modeling technique was developed to solve long-term, chronic, dynamic industrial management problems; today, SD is applied to solve various business policies and strategic problems. A typical SD study focuses on understanding how the components of a system interact; how and why the dynamics of concern are generated; and how policies and decisions affect system performance. This paper presents a study which models a computer network as a systems dynamic model to explore cyber-attacks and the resulting system-level effects that might occur on host OSI layers, layer 4 and above, in the OSI model. Preliminary results indicate that by using system dynamic cyber security simulation an organization can imitate the attacker(s) activities in OSI layer 4 and above and assess (and/or mitigate) the system's risk exposure. In this paper we are presenting a proof-of-concept SD model for application layer cyber security.

Keywords: Cyber security; cyber security modeling; system dynamics; application layer cyber security; simulation and modeling; cyber-attacks/defenses.

1 Introduction

Modeling is the process of capturing the key characteristics or behavior of a real world system under study and it helps us in understanding the essential parts of a system and the relationship between them [1-3]. A typical cyber security model has information about the network infrastructure, security settings, and list possible security vulnerabilities and threats [4]. Simulation is the process of imitating a system, based upon our knowledge or assumptions about the behavior of the parts of a system, in order to get the insight of a whole system [5]. Similarly, by using known vulnerabilities and the current knowledge about infrastructure and security controls, the cyber security simulation allows an organization to imitate the attacker activities and helps to assess the system's risk exposure [4].

Networks are normally modeled or simulated through discrete-event techniques, in which the state of system changes only at discrete points in time. Depending on the granularity of the model, this means simulating the movement

of packets throughout a network and measuring such things as throughput, latency, etc. In discrete-event simulation (DES), cyber-attacks are simulated by altering the flow or rate of packets and observing the result.

Discrete-event network simulation tools such as cnet, EcoPredictor, IT SecisionGuru, NetCracker, and NetRule are used by professional system administrators and systems application designers to model and analyze packet traffic, buffer overflow, operating system compromise, and so on. [1]. With respect to information security, these network simulation tools are normally used to model tasks such as server availability and router availability. They also used to make the in depth analysis of authentication server's loads and unusual network traffic [1].

DES approach has two flaws. First, simulations can only simulate a few seconds worth of network operations due to the massive number of packets that are transmitted during normal operations. Second, these models focus primary on packet traffic. This means that cyber-attacks (and the resulting cyber defenses) are viewed from the network layer, that is, layer 3 in the open system interconnection (OSI) model. This obscures more insidious attacks at higher layers in the OSI model. In addition the DES approach has also the following drawbacks with respect to cybersecurity system modeling [7,10,11,17]: 1) the problem scope is operational not strategic, 2) used to model a particular process of a system (that is, used for detailed analysis of a particular process) not the entire system, 3) does not allow best guesses and expert opinion in the model building process, 4) the creation of a DES model typically requires a great investment of time in data analysis and distribution fitting to ensure the model is statistically valid, 5) the DES system performance is determined by the accurate historical data or estimates of future performance, 6) DES models more often reflect systems where entities are processed in a linear fashion. Feedback plays less of a role in these systems (i.e., feedbacks and delays are not emphasized), and 7) the resulting model is an opaque box, that is, the user does not understand the underlying mechanics and is not transparent to the user. In general, the DES is more suitable to simulate systems with low level of abstraction and well defined processes and not suitable to simulate continuous processes related to extensive processes of feedback [7, 17].

This paper presents a study which models a computer network as a systems dynamic model (a.k.a. continuous simulation). Its objective is to explore more insidious cyber-attacks and the resulting system-level effects that might occur on host OSI layers (layer 4 and above); that is, on transport, session, presentation, and application layers. For modeling we have used the concept of System Dynamics (SD), because it allows us to see systemic effects – something that is not feasible with DES. In SD methodology, the stock-flow diagram is used depict the underlying mathematical model, the model structure and the interrelationships between its components. Once the underlying mathematical structure is captured, the stock-flow diagram can be easily translated into system of differential equations, and simulated using continuous simulation software such as Powersim.

2 Modeling and Simulation in Cybersecurity

For analyzing complex problems such as cyber security and developing design solutions, many approaches are used in engineering science. These methods include descriptive models, system testbeds, and system (or simulation) models [12]:

- *Descriptive Cyber Security Models:* Diagrams with supporting text are used to describe a system in descriptive models. Attack graphs are example for descriptive models. A typical attack graph consists of network diagrams plus descriptions of applicable malware methods and mitigation techniques.
- *System Testbed Cyber Security Models:* System testbeds are extreme and most rigorous tools used for model analysis. These testbeds include working prototypes and live full-scale physical testbeds. Laboratory-scale equipment may be connected to sophisticated control systems to study device-level vulnerabilities. Information Warfare Analysis and Research (IWAR) Laboratory [13] is a classic example for the cyber security testbed. IWAR is an isolated laboratory for students to practice various computer security attacks/defenses.
- *Cyber Security System Models and Simulation:* System models capture the essential characteristics or behavior of the systems under study. These are middle level and lower cost methods. In this approach, generally, fully synthetic or simulated models are used for analysis and system understanding.

Though descriptive models are simple and least expensive, they do not predict the future behaviors or states of the system under study. System testbeds are very good approach for simulating technology level network attacks/defenses. But building system testbeds consume a large amount of resources, money, and time. Moreover, the system testbeds must be brought into original state before

each and every cyber attack/defense run. In addition to these drawbacks, system testbeds are used to predict excessively narrow sets of problems due to the practical testbed sizes and practical limitations on approaches and measurement techniques. Therefore, the simulation model is used to better understand the behavior of the system under study or expected behavior or states of the proposed system and to study the effectiveness of the system design. [12,14,15]

When information security threats are not acute, both information security and lay managers can use modeling and simulation to better understand their information environment both on a concrete and abstract level. Once a model is developed and validated (using simulation), proactively it can be used to identify system vulnerabilities and reactively it can be used to investigate a real-world system or provide education and training by means of various “what if” questions [1,16]

Using modeling and simulation in the cyber security field provides many benefits including [4]: risk analysis, planned network change verification, security controls and resources optimization, complex network analysis, complex networks comparison, and cost-effective training to cyber security personnel.

3 System Dynamics and Cybersecurity

System dynamics (SD) [6] is a methodology used to understand how systems change over time. In SD, a system is defined as a collection of elements that interact continuously over time to form a unified whole [7]. In the 1960s, the SD modeling technique was developed to solve long-term, chronic, dynamic industrial management problems [8]; today, SD is applied to solve various business policies and strategic problems [9-11].

In SD, the “structure” of the system is defined by the totality of the relationship between the physical processes, information flows, and managerial policies. The structure generates the dynamic behavior patterns of the system. A typical SD study focuses on understanding how the components of a system interact; how and why the dynamics of concern are generated; and how policies and decisions affect system performance. [11]

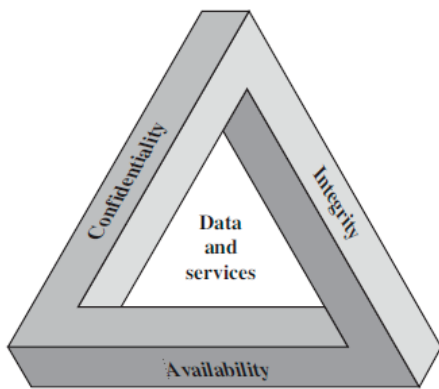
System dynamics uses a causal-loop diagram to capture the factors affecting the behavior of the system. The linkage between the system and its operating environment, and feedback loops among the elements in the system are depicted in the causal-loop diagram. This causal-loop diagram/analysis provides decision-makers with insight into how systems behave as a whole. Simulation software, such as Powersim, lets decision-makers extend their understanding of a system by either adjusting the system parameters, adding new linkages and feedback loops, or rearranging components of the system. Thus, by using a SD simulation software the

decision-maker(s) can model a variety of scenarios and observe the system performances under various conditions. [7]

When we apply SD to cyber security, the network is considered as a system, similar to a physical system of pipes through which water flows. The amount of water that can flow into and out of node represents the bandwidth of the network traffic. A denial of service attack, for example, is modeled by trying to force more water into a node than it can handle. Another dimension of the model is the quality of the water. Network traffic that contains bogus data or viruses are thought of as water that has contaminants. The degree or type of contaminants would affect the operation of the nodes.

4 Cybersecurity Requirements

According to NIST Standard FIPS 199 (Standards for Security Categorization of Federal Information and Information Systems), the three fundamental objectives of information system security are Confidentiality, Integrity, and Availability (CIA) [18]. The CIA triad is shown in figure 1.



Fi.1. The Security Requirements Triad [18]

FIPS 199 provides a useful characterization of these three objectives in terms of requirements and the definition of a loss of security in each category [18]:

- **Confidentiality:** Preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information. A loss of confidentiality is the unauthorized disclosure of information.
- **Integrity:** Guarding against improper information modification or destruction, including ensuring information nonrepudiation and authenticity. A loss of integrity is the unauthorized modification or destruction of information.
- **Availability:** Ensuring timely and reliable access to and use of information. A loss of availability is the disruption of access to or use of information or an information system.

The CIA triad is a widely used benchmark for evaluation of information systems security. It must be addressed each time an information technology team installs a software application or computer server, analyzes a data transport method, creates a database, or provides access to information or data sets [19].

5 The Application Layer and Corresponding Cyber Attacks

The OSI (Open Systems Interconnect) network model depicts network communication at varying levels of detail. It not only serves to characterize computer-to-computer communication, but can also provide a basis for categorizing the level and degree of cyber attacks. In particular, the model provides organizations an insight into where vulnerabilities that may exist within their infrastructure and how to apply appropriate control measures; and equips computer professionals with a deeper understanding of data movement through the network and how attacks can occur at each level. [20]

The application layer (or layer-7) provides actual interface for users and application processes. The major functions of this layer include resource sharing and device redirection, remote file access, remote printer access, inter-process communication, network management, directory services, electronic messaging, and network virtual terminals. [21]

Layer-7 attacks involve exploiting weaknesses in software commonly found on servers in order to gain system-level account privileges and gain access to the running applications on the system.

A common layer-7 attack on confidentiality is a Trojan horse, meaning a program designed to breach the security of a computer system while ostensibly performing some innocuous function. Trojan horses are generally used to capture sensitive information and distribute it back to the attacker, or to install viruses.

Viruses and worms are perceived as integrity attacks at Layer-7. A computer virus is a piece of code that is capable of copying itself and typically has a detrimental effect, such as corrupting the system or destroying data. A worm is self-propagating and spreads from one computer to another computer in the network.

Examples of layer-7 attacks that affect availability include HTTP POST flood, HTTP GET attacks, and slow HTTP attacks. An HTTP POST flood is a type of DDoS attack in which the volume of POST requests overwhelms the server so that the server cannot respond to them all. This can result in exceptionally high utilization of system resources and consequently crash the server. An example is the appearance of websites that use dynamic HTML methods to

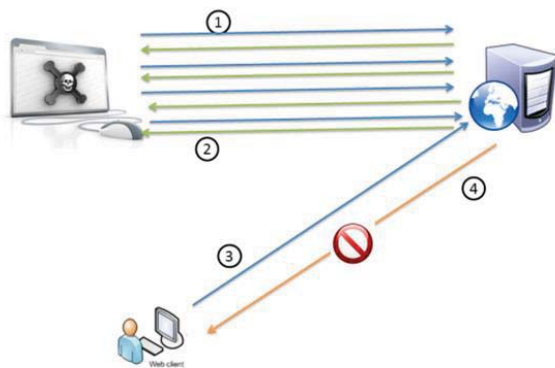
launch HTTP floods simply by loading a specific website. An HTTPS POST flood is similar to the HTTP POST flood sent over an SSL session, where the actual data transferred back and forth is encrypted. An HTTP GET flood is an application layer DDoS attack methods in which attackers inundate a server with get requests in an effort to overwhelm its resources, rendering the server slow, unreachable, or unresponsive. Slow HTTP attacks exploit a flaw in the HTTP protocol which requires requests to be completely received by the server before they are processed. If an HTTP request is not complete the server keeps its resources busy waiting for the rest of the data to be arrived. If the server keeps too many resources busy, this creates a denial of service.

6 Limited Proof-of-Concept (PoC) Model

To show the feasibility of the SD modeling approach, we constructed a Proof-of-Concept (PoC). The PoC simulates an HTTP Slow Read Attack (one of the layer-7 DDoS attack discussed earlier) on a hypothetical network.

6.1 Concept Design

Slow HTTP (Slowloris, Slow HTTP POST, and Slow HTTP GET) DoS attacks rely on the fact that the HTTP protocol, by design, requires requests to be completely received by the server before they are processed. If an HTTP request is not complete, or if the transfer rate is very low, the server keeps its resources busy waiting for the rest of the data. If the server keeps too many resources busy, this creates a denial of service (figure 2). In these types of attacks, a single machine can take down another machine's web server with minimal bandwidth.



- (1) HTTP GET request from attacker
- (2) HTTP GET response from server
- (3) HTTP GET request from client
- (4) Server busy/unavailable message

Fig.2. PoC Architecture and Data Flow

- **Normal Scenario:** Establish a connection to the server, read/download a 1MB file through several TCP packets sized 1448 bytes (Maximum Segment Size that the underlying communication channel supports) from the HTTP Server. The download will be completed in a minute or two depending on the network speed.
- **Attack Scenario:** Send as many as legitimate HTTP requests to larger web page, with size larger than server's socket's send buffer (more than 128Kb), and read/download the file as slow as possible to cause a DoS attack. That is, exploit the vulnerability that most modern web servers do not limit the connection duration if there is a data flow going on.

6.2 Model

The stock-and-flow (S&F) diagram of the model of HTTP slow read attack is shown in figure 3.

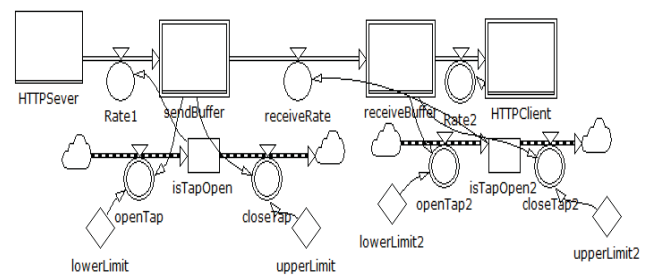


Fig.3. SD HTTP Slow Read DoS Attack Model

6.3 Model Validation

Our validation follows a two-step process [22, 23]: First establish the validity of the structure of the model (structural testing), and then evaluate the accuracy of the model behavior's reproduction of real behavior (behavioral testing).

6.3.1 Structural Testing

Structure Verification Test: Compares the form of the equations of the model with the relationships that exist in the real system or in the literature. Some semi-formal tools such as formal inspections, reviews, walkthroughs, and data flow analysis are typically used in the verification of structure confirmation tests. [22-24]

The following model equations are verified with the HTTP Slow Read Attack equations available in the literature,

1. $HTTPServer(t) = 1048576 - \int_0^t (Rate1(t))dt$
2. $sendBuffer(t) = 0 + \int_0^t (Rate1(t) - receiveRate(t))dt$
3. $receiveBuffer(t) = 0 + \int_0^t (receiveRate(t) - Rate2(t))dt$
4. $HTTPClient(t) = 0 + \int_0^t (Rate2(t))dt$
5. IF ($sendBuffer \leq lowerLimit$) Then $Rate1.openTap = True$
6. IF ($sendBuffer > upperLimit$) Then $Rate1.closeTap = True$
7. IF ($receiveBuffer \leq lowerLimit2$) Then $receiveRate.openTap2 = True$
8. IF ($receiveBuffer > upperLimit2$) Then $receiveRate.closeTap2 = True$

Parameter Verification Test: Conduct the conceptual and numerical evaluation of the constant parameters against knowledge of the real system or literature. Conceptual evaluation identifies the elements in the real system that corresponds to the parameters of the model. Numerical evaluation estimates the numerical value of the parameter with enough accuracy. [22, 23]

The values assigned to the parameters of our simulation are sourced from the existing knowledge and numerical data form Apache webserver data [25].

Extreme Conditions Test: Evaluating the validity of model equations under extreme conditions by assessing the likelihood of the resulting values against the knowledge/anticipation of what would happen under a similar condition in the real system. [22, 23]

This is verified using attack scenario (figure 4). As described in attack scenario earlier in the PoC design, the clients are reading the 1MB file for ever, causing the DoS attack.

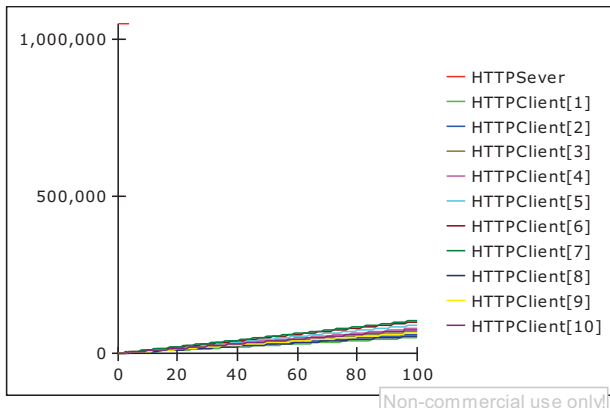


Fig.4. Attack Scenario Result

Dimensional Consistency Test: Checking the right-hand side and left-hand side of each equation for dimensional consistency (it is a theoretical test). [22, 23]

That is, we need to test all units are consistent in all mathematical equations. In our case, all times are in “sec” and all data sizes are in “bytes”; therefore, our model passes the dimensional consistency test.

6.3.2 Behavior Tests

Behavior Reproduction Test: The simulation outputs for normal scenario (figure 5) verifies the model-generated behavior matches observed behavior of the real system. This indicates that the entire 1MB file is downloaded approximately 80 seconds (as we have mentioned in the normal scenario in the design).

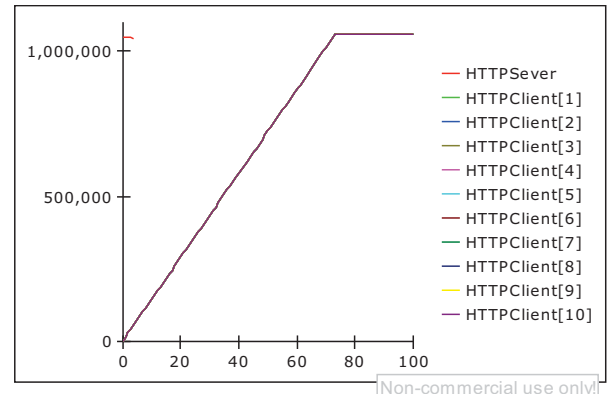


Fig.5. Normal Scenario Result

Behavior Anomaly Test: The model behaved like the real system under study and we did not discover any anomalous features of model behavior, which sharply conflict with behavior of the real system.

7 Summary and Conclusion

Networks are normally modeled or simulated through discrete-event techniques. Since the primary focus of the discrete-event simulations are on packet traffic i.e., the cyber-attacks/defenses are viewed from the network layer (layer 3 in the OSI model), it obscures more insidious attacks at higher layers in the OSI model. Therefore to model cyber security attacks on host OSI layers, we have adapted a system dynamics based simulation modeling technique. In this paper we have demonstrated an application layer cyber attack using system dynamics PoC model and also shown the structural and behavioral verification of the PoC model. Therefore, by using known vulnerabilities, similar to this, and the current knowledge about infrastructure and security controls, the system dynamic cyber security simulation modeling allows an organization to imitate the attacker activities in OSI layer 4 and above and helps to assess and mitigate the system’s risk exposure.

8 References

- [1] John Saunders, "Modeling the Silicon Curtain", SANS Institute, 2001
- [2] Wikipedia, "Computer simulation", https://en.wikipedia.org/wiki/Computer_simulation
- [3] Romano Elpidio, Chiocca Daniela, and Guizzi Guido, "An Integrating approach, based on simulation, to define optimal number of pallet in an Assembly Line", 20th Issat Conference, Reliability and quality design, 2014
- [4] "Using Risk Modeling and Attack Simulation for Proactive Cyber Security: Predictive Solutions for Effective Security Risk Management", Skybox Security Inc., whitepaper, 2012.
- [5] "System Modeling and Simulation", www.inl.gov/systemsengineering
- [6] Jay Wright Forrester, "Industrial dynamics", MIT Press; 1961
- [7] Al Sweetser, "A Comparison of System Dynamics (SD) and Discrete Event Simulation (DES)", albert.sweetser@ac.com
- [8] Barlas Y, "System dynamics: systemic feedback modeling for policy analysis in knowledge for sustainable development—an insight into the encyclopedia of life support systems", UNESCO Publishing-Eolss Publishers, 2002
- [9] Coyle RG, "System dynamics modelling: a practical approach", Chapman & Hall, 1996
- [10] Sterman JD, "Business dynamics: systems thinking and modeling for a complex world", McGraw-Hill, 2000
- [11] Dimitrios Vlachos, Patroklos Georgiadis, and Eleftherios Iakovou, "A system dynamics model for dynamic capacity planning of remanufacturing in closed-loop supply chains", Computers & Operations Research 34 (2007) 367–394.
- [12] Michael McDonald, John Mulder, Bryan Richardson, Regis Cassidy, Adrian Chavez, Nicholas Pattengale, Guylaine Pollock, Jorge Urrea, Moses Schwartz, William Atkins, and Ronald Halbgewachs, "Modeling and Simulation for Cyber-Physical System Security Research, Development and Applications", Sandia Report, SAND2010-0568
- [13] Scott Lathrop, Gregory Conti, and Daniel Ragsdale, "INFORMATION WARFARE IN THE TRENCHES: Experiences from the Firing Range", Third Annual World Conference on Information Security Education (WISE3), California, USA, 2003, DOI: 10.1007/978-0-387-35694-5
- [14] Dessouky, "System Simulation", lecture slides
- [15] Sinclair, J. B. "Simulation of Computer Systems and Computer Networks: A Process-Oriented Approach", Rice University, 2004.
- [16] Villarreal Gonzalo , De Giusti Marisa , and Texier José, "GPSS Interactive Learning Environment", Elsevier, 2012
- [17] Thiago Barros Brito, Edson Felipe Capovilla Trevisan, Rui Carlos Botter, A CONCEPTUAL COMPARISON BETWEEN DISCRETE AND CONTINUOUS SIMULATION TO MOTIVATE THE HYBRID SIMULATION METHODOLOGY, Proceedings of the 2011 Winter Simulation Conference
- [18] William Stallings, Cryptography and Network Security Principles and Practice, 5/e, Prentice Hall
- [19] Confidentiality, Integrity and Availability (CIA), <http://it.med.miami.edu/x904.xml>
- [20] Lee Hazell, Network Vulnerabilities and the OSI Model, September 26, 2014, <http://cybersecuritynews.co.uk/network-vulnerabilities-and-the-osi-model/>
- [21] The OSI Model's Seven Layers Defined and Functions Explained, Article ID: 103884 - Last Review: 06/13/2014 06:08:00 - Revision: 2.1, <https://support.microsoft.com/en-us/kb/103884>
- [22] Forrester JW and Senge PM, Tests for building confidence in system dynamics models. TIMS Studies in the Management Sciences 1980; 14:209–28.
- [23] Barlas Y, Formal aspects of model validity and validation in system dynamics. System Dynamics Review 2000; 12(3):183–210.
- [24] Osman Balci (1994), "Validation, Verification, and Testing Techniques Throughout the Life Cycle of a Simulation Study," In Proceedings of the 1994 Winter Simulation Conference (Orlando, FL, Dec. 11-14). IEEE, Piscataway, NJ, pp. 215-220.
- [25] https://httpd.apache.org/docs/2.4/misc/security_tips.html

Modeling and Simulation for Grounding a Mechatronic Test Environment for Inertial Measurement Units

Dan Tappan and Josh Czoski

Department of Computer Science, Eastern Washington University, Cheney, WA, USA

Abstract - *One of the most difficult aspects of learning to play violin is posture. Students practice endlessly in front of a teacher and mirror to ensure correct bow movement with respect to the violin. This work describes a hybrid modeling-and-simulation environment fabricated to evaluate the use of inexpensive inertial measurement units for a larger project to perform such monitoring automatically. It describes a mechatronic test rig that simulates complex bow movement for repeatable, controlled experiments. This physical hardware model introduces its own idiosyncrasies into the evaluation process and in turn requires evaluation by a virtual software model. The combined result is an objective comparative proof-of-concept framework for grounding (calibrating and tuning) the two models against each other and reality to tease out performance characteristics.*

Keywords: violin, data acquisition, inertial measurement unit, mechatronics, performance evaluation

1 Introduction

Learning to play violin is a long and challenging process. One of the greatest difficulties involves developing and maintaining appropriate posture to keep the bow oriented correctly with respect to the violin. During lessons, a teacher closely monitors this activity and indicates whenever there is a problem. Students on their own practice in front of a mirror to monitor themselves. Either way, the process is onerous. An automated system could be much more practical. The recent explosion of popularity in drones, virtual-reality gaming devices, and motion-based smartphone apps has had a profound effect on increasing the capabilities and availability of small, inexpensive inertial measure units (IMUs) that keep track of their position and orientation in three-dimensional space in real time. Attaching one to the violin and another to the bow provides their physical states. In theory, subtracting the two states would produce the relative state of the bow with respect to the violin and facilitate its evaluation within an acceptable range of motion.

In practice, however, this approach morphs into a much larger problem of using the IMUs appropriately and compensating for their many shortcomings. The focus of this paper is on how to evaluate the real-world performance of IMUs objectively for a larger project of this kind. The underlying approach involves conducting repeatable,

controlled experiments as part of the scientific method. Directly measuring the IMUs on a real violin and bow exhibits neither property: the tests are not controlled because the violinist cannot be precisely sure of his or her actions, isolate and change them individually, or quantitatively compare them to a standard of correctness; nor are they repeatable because multiple attempts cannot produce the same results or in the same way because humans are not consistent enough.

To mitigate this limitation, a physical model served as a surrogate for the real violin and bow assembly. This mechatronic device combined solutions from computer science, electrical and mechanical engineering, and fabrication with commercial off-the-shelf parts to produce a physical simulation device that was much more amenable to controlled experiments on the real IMUs. However, its idiosyncrasies introduced their own problems, which led to the need to evaluate its own performance. The final result was a virtual model in software that was both convenient for high-speed experiments and arbitrarily accurate. In order to use the virtual model as a surrogate for the physical model, which in turn was a surrogate for the real assembly, the performance characteristics of all three needed to be identified, measured, modeled, tested, analyzed, and validated.

This paper addresses a proof-of-concept integrated environment of modeling, simulation, visualization, and analysis as an objective comparative framework for this heavily underdetermined, messy comparative problem. It involves a wide range of creative *what if* engineering thinking and doing. Specifically, it addresses calibrating and tuning (i.e., grounding) each of the models to each other, executing them, and comparing their performance. It partially uses a Monte Carlo approach to perform sensitivity analysis on the parameters to tease out their independent and interdependent contributions.

2 Background

The larger violin project aimed at determining in real time whether the violinist was manipulating the bow appropriately. It addressed posture and movement only, not how to play correctly in terms of musical notes and style. The definition of acceptable manipulation is complicated and not strictly necessary here. This paper focuses only on establishing the state of the bow and violin in space and

time as correctly and reliably as possible, not on their interrelationships to produce music. For context, however, the purpose of this information was to establish and monitor a complex three-dimensional region of acceptability based on two sources of state data in Figure 1a: the violin, and the left end of the bow opposite the violinist's hand. The violinist has a wide range of freedom in holding the instrument (even upside-down is physically possible); therefore, the acceptable state of the bow is relative to the state of the violin, and both are needed. Conveniently, the same solution applied to both, but for simplicity, the rest of this paper usually refers to the bow only. The bow was also subject to the most movement and demanding tests, so it makes the better representative given the space limitations.

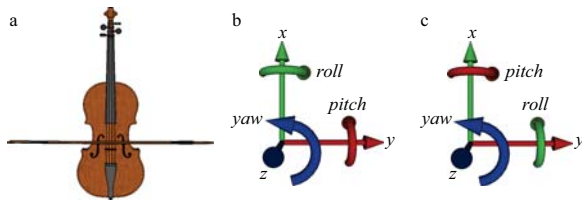


Figure 1: Violin with bow [1], violin and bow coordinate systems

State is defined in terms of three components in three-dimensional space that together compose a spatial model with six degrees of freedom (6DOF). The first is *position* as (x,y,z) relative to the initial position $(0,0,0)$ from when the measurements started. It is not necessary to know the absolute start position in the real world (e.g., two meters from the wall, one from the floor), and in fact, these values have no specific units of distance. The second is *attitude* as *yaw*, *pitch*, and *roll* in degrees relative to the initial values of zero. The third is relative *time*, which contributes to computing the speed (change in state) and acceleration (change in speed).

The state of the violin in the coordinate system in Figure 1b serves as the reference against which to measure the state of the bow. The bow uses the system in Figure 1c because it is more intuitive to swap the roll and pitch axes to account for the perpendicular interaction. In other words, pitch for the bow in the right hand should be at a right angle to the pitch of the violin in the left hand. The desired bow yaw should be 90 degrees counterclockwise from the violin yaw about the z axis at the origin, which is where the bow and strings intersect. Bow pitch is the arcing movement as the bow passes over the strings. It is the angle of the bow relative to the angle of the violin on the plane formed by the x and y axes, ranging over roughly ± 20 degrees.

Yaw deviation of the bow is what violinists at all levels strive to minimize. Pitch deviation is not an error because pitch must vary in order to interact with different strings. Roll deviation could be a consideration, but compared to yaw, it is minor. Determining the state on all axes, however, is necessary to solve the state of the complete system. The

details of the math, physics, and engineering involved in moving a bow are out of scope, but their relationship to the larger project is worth mentioning for context. *Kinematics* is the study of geometry in motion without regard to the underlying mechanism (the *kinetics*), such as force, mass, and gravity. For example, pushing the base of the bow forward in line with it moves the other end a corresponding distance in the same direction. This event translates an action (the cause) into a reaction (the effect). In other words, the violinist must do the former so that the latter happens. *Inverse kinematics* in this context is the study of how to achieve this result given the many options. For example, the upper arm, elbow, forearm, and wrist can combine in many ways to produce the same action, and other actions can also lead to the same reaction. It is the teacher's job to make sure that the appropriate actions occur in the right way for the right reasons. This system considers only the kinematics of producing the result. In the bigger picture, the correct result is necessary to play a violin well, but it alone is not sufficient because technique also matters.

3 Architecture

The architecture consists of two complementary parts for reliably executing repeatable, controlled experiments: the physical model operates directly on hardware to simulate the role of the bow, and the virtual model is its software counterpart. Section 5 discusses how the two work together.

3.1 Physical model

The hardware is a computer-controlled electromechanical device consisting of a movable turret with a movable bow holding an IMU, collectively called the test rig.

3.1.1 Turret

The turret, a stock RobotGeek Roboturret in Figure 2a, provides a flat, open platform typically intended to hold a camera [6]. A separate hobby servo motor (2b) drives the rotational movement of its two degrees of freedom, pan and tilt, which respectively correspond to yaw and pitch for the bow. The turret is a self-contained unit with its own power supply, an Arduino Duemilanove embedded controller (2c), and a thumb-sized joystick [7]. It is strong and fast enough to simulate the angular movement of the bow, but it does have notable limitations, covered in Section 3.2.

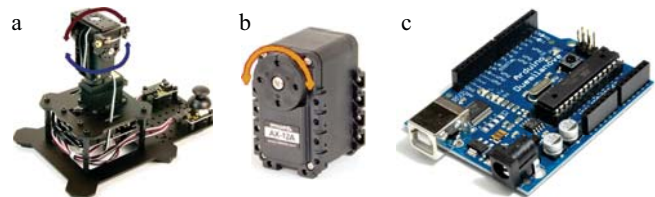


Figure 2: Turret, servo, and controller [6,6,7]

3.1.2 Bow

The drive mechanism in Figure 3a for the simulated bow occupies the platform of the turret. This component required design and fabrication from scratch. It consists of a geared motor with a pinion gear that meshes against the rack on a lightweight, extruded square aluminum shaft 50cm long. The motor driver is a JRK 12v12 (3b), which accepts a wide range of convenient parameters, such as acceleration and deceleration profiles [8].



Figure 3: Bow drive mechanism and motor controller [8]

Rotating the motor and thus the pinion gear causes the shaft to move linearly through a low-resistance channel guided by bearings. This action must be precise because the shaft has to move to a specified position at a specified rate. The hardware has two complementary positioning strategies. The first is an open-loop control system that counts how many rotations of the motor occur in 1/48th increments. The motor has a built-in quadrature encoder for this purpose, which interfaces directly with the motor controller. Each partial rotation, or step, translates into a corresponding linear movement of theoretically around 0.06mm. However, the messy operating environment causes minor counting errors that compound over time. For example, moving the bow back and forth the same number of steps many times does not return to exactly the original position. This error is minor, but in combination with many other errors inherent throughout this work, this solution alone is not acceptable.

To mitigate this limitation, a simple closed-loop control system uses an optical sensor to determine when a black dot at the back center of the shaft passes the drive mechanism, which means that the bow is back in its home position. This signal resets the step count to negate any counting errors. In fact, it is so effective that the front of the shaft includes similar dots at 5mm increments as reference points for the image processing discussed in Section 5.2. The back dot also serves to initialize the bow to the same starting position for each test. The Arduino can also initialize it to other positions in code, or the user can manipulate the joystick.

3.1.3 Sensors

Reliably knowing the position and attitude of the bow and violin (real or simulated) is essential. Three types of sensors determine these values. In general terms, an *accelerometer* measures change in position from the previous measurement; a *gyroscope*, change in yaw, pitch, and roll; and a *magnetometer*, absolute yaw (i.e., compass heading).

They collectively form an inertial measurement unit and typically reside on a single chip, here an MPU-9150 in Figure 4a [9].

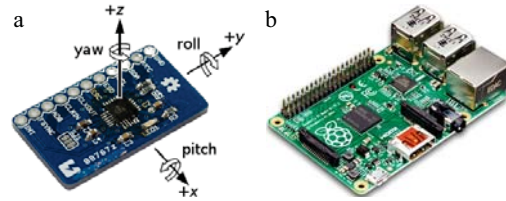


Figure 4: Inertial measurement unit and Raspberry Pi [9,10]

In reality, computing these values reliably is far more complex. Despite the apparent convenience of this IMU, it is in practice quite an unreliable device with messy output and many inherent errors. Higher-quality devices were available at acceptably higher cost, but they were larger and heavier, which was prohibitive for use on the end of a long bow in motion. (The supplier also billed this product as “the world’s first” 9DOF device with complex onboard digital motion processing, which sounded highly promising [9].) Mitigating these errors in the larger project involved complex signal processing, primarily Kalman filtering, which is out of scope here [2]. Nevertheless, even with such processing, the results were hardly ideal, which is the subject of Section 6.

The two IMUs simply acquired state data. They did not process or store anything. For this part, they communicated with a Raspberry Pi, a small, inexpensive, yet powerful single-board computer in Figure 4b [10]. The programming language for processing the data was Python. While not the fastest for number crunching, it was adequate for the requirements. Moreover, it includes convenient libraries for communicating with the test rig over the I²C and USB buses and GPIO (general-purpose input/output) pins.

3.1.4 Architectural overview

Figure 5 shows the architectural overview of the main components of the system and their communication. A laptop serves as the base station to provide the user with a convenient interface into the other components, which do not have a keyboard or display of their own. It also collects the raw data during tests.

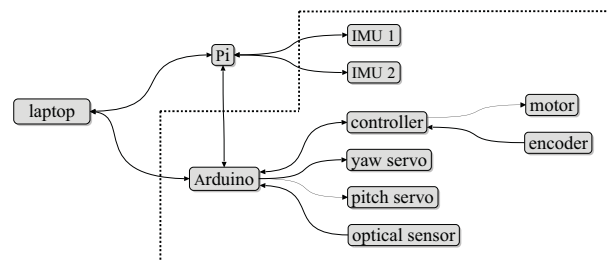


Figure 5: Architectural overview

Figure 6 shows the test rig, which contains the components within the dashed box above. The IMU is on the right end of the bow.

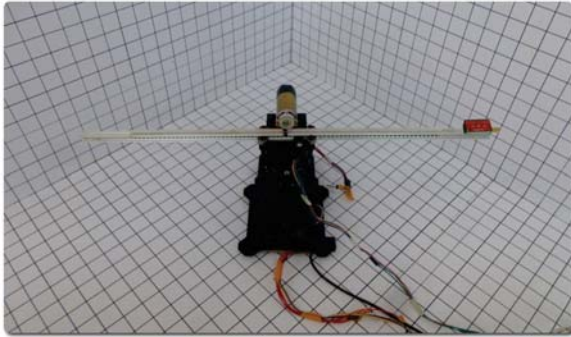


Figure 6: Test rig in calibration chamber

3.2 Virtual model

The physical model is a surrogate for the real-world bow, which does not reliably lend itself to controlled experiments. The virtual model in turn is a surrogate for the physical model, which exhibits a large number of issues that undermine its use as the only means of testing the IMU on the bow. Section 5 covers in detail how the three variants function together better in a modeling-and-simulation environment than any one could alone.

The virtual model is a simplified computational engineering representation of the turret and bow, as well as of the physical mechanisms that connect and move them. The turret has inputs for yaw and pitch, and the bow for speed and distance of motor rotation. The virtual model in this form is perfect, which would be the ideal goal for the physical model, too. However, the latter exhibits a wide range of errors and limitations owing to its real-world nature and the quality of the components and construction. In order to use the virtual model in place of the physical, it must model these errors to some configurable degree. Trial-and-error experimentation through simulation played this role.

All substantive errors are related to interconnections and movement. For instance, the yaw and pitch axes of the turret pivot through their respective servo motors. These low-cost hobby units have a noticeable amount of rotational backlash, or slop, in their ability to hold a specified angle: it can vary by plus or minus several degrees, depending on how much force is applied. The bow suffers from a similar problem in multiple respects because the motor has its own internal gears with backlash, and the pinion gear does not mesh perfectly with the rack. As a result, the bow can vary in position along its length by plus or minus a millimeter or more. The bow also has sideways slop because the bearings that hold it in position have some freedom. Reducing the tolerance improves this performance, but it introduces another problem because the friction becomes much higher, and the bow motor does not behave as uniformly.

Everything in engineering design is a compromise. The goal here was not to produce the best solution for a specific set of test cases, but rather a generalized proof of concept that fleshed out areas to investigate further.

While the contribution of any of these errors alone is relatively small, they amplify over the length of the bow. For example, one degree of error at its maximum extension (normally avoided) equates to roughly a centimeter of perpendicular distance error that the IMU at the end sees. Even worse, some errors compound. For example, the yaw axis of the turret holds the servo for the pitch axis, so pitch measurements suffer from both errors.

A completely realistic virtual model would be overwhelmingly complex and difficult to define, test, and evaluate. Therefore, this abstraction and simplification ignores contributions from vibration and resonance (compounded interacting vibrations), as well as balance and stress-related factors. For example, the bow at its maximum extension exerts far more bending moment (twisting) at the bearings than it does in balance at its minimum extension. Similarly, electromagnetic effects on the magnetometers from the motors were problematic, but effectively impossible to model.

4 Visualization

A laptop logs the data from the controllers and sensors in quantitative form in terms of the expected and actual states (i.e., position and attitude), as well as time. The nature of this work lends itself to interpreting and evaluating the results in visual form. The visualization stage thus provides a variety of perspectives that help determine performance qualitatively, which tends to be more intuitive.

4.1 Two-dimensional static visualization

Static visualization involves displaying the results after a test is complete. Although it should be possible to present them dynamically in real time with additional software, the amount of data is large, and the changes are generally too quick and subtle for a person to follow in detail anyway. Instead, the output exports natively to Excel, as in Figure 7, where any manner of post-analysis is then possible.

time	np1	py1	pi1	np4	py4	pi4	np11	py11	pi11	np2	py2	pi2	np4	py4	pi4	np8	py8	pi8	np16	py16	pi16	np32	py32	pi32
0.00	0.073	0.080	0.159	1.563	0.114	0.115	0.659	0.217	-0.051	0.005	-0.177	0.112	1.981	0.108	0.248	-0.058	0.066	0.237						
0.10	0.010	-0.050	-0.045	-1.855	-0.495	-0.354	-0.122	-0.116	0.018	-0.085	0.067	-0.015	0.038	-0.034	-0.349	-0.251	-1.74	-0.036						
0.20	0.004	0.096	-0.106	-1.116	-0.094	-0.332	-0.080	-0.031	-0.020	-0.052	0.113	0.189	0.124	-0.245	0.328	0.525	0.295	0.036						
0.30	0.002	0.282	-0.002	-0.002	-0.002	-0.002	-0.002	-0.002	-0.002	-0.002	-0.002	-0.002	-0.002	-0.002	-0.002	-0.002	-0.002	-0.002						
0.40	-0.081	0.118	-0.247	1.716	-0.404	-0.217	-0.951	-0.066	-0.009	0.100	0.152	0.126	1.541	-0.099	-0.521	-0.341	-0.050	0.026						
0.50	0.118	0.259	0.213	0.600	-0.275	-0.544	-0.681	-0.036	-0.006	-0.444	0.031	0.329	1.817	0.227	0.151	0.104	0.038	0.044						
0.60	0.083	0.058	0.238	-1.293	-0.701	1.242	0.514	0.212	-0.050	-0.095	0.122	-0.427	0.472	0.378	0.577	0.508	0.270	0.076						
0.70	0.001	0.261	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001	-0.001						
0.80	0.029	0.051	0.528	-0.348	-0.778	0.514	0.370	-0.080	-0.040	-0.010	0.120	0.188	-1.595	-0.009	1.531	0.272	0.237	0.160						
0.90	0.080	0.080	0.388	-1.593	-0.328	-1.447	0.374	0.051	0.173	-0.146	0.110	0.186	1.730	0.662	1.104	0.372	0.602	0.071						
1.00	0.017	0.141	0.554	-0.247	-0.260	0.972	0.794	0.141	-0.155	-0.097	0.101	0.529	1.804	-0.294	1.151	-0.007	-0.220	-0.165						

Figure 7: Excel numerical data

Ordinary two-dimensional graphs in countless configurations and combinations can provide a wealth of insight into the results. Figure 8 shows a notional example from a one-second test on attitude (at 10ms intervals). The

paired lines show how the expected (smooth) and actual (jagged) results varied over time.

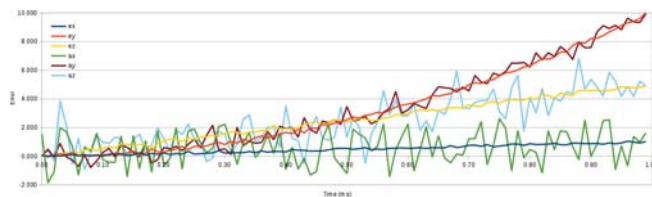


Figure 8: Excel graphical data

4.2 Three-dimensional dynamic visualization

Dynamic visualization involves displaying the results in real time as a test executes or afterwards statically in replay mode. The Java OpenGL 3D viewer in Figure 9 shows the mechanical configuration at any point in time from any perspective [4]. Its code is highly configurable and extensible to any special-purpose analysis. It can also render a variety of metadata to depict information that is not possible in the Excel visualization.

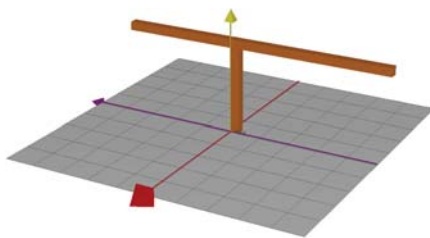


Figure 9: 3D viewer

5 Experimental framework

The overarching goal of this work was to use both physical and virtual modeling and simulation as a better integrated test environment for the IMUs than the real violin and bow could provide. In particular, it was essential to be able to run a wide range of tests for good breadth of coverage, as well as more instances of each for depth in statistical processing and analysis. The advantages of the physical and virtual models over the real bow in this respect are substantial. However, the value of their collective results depends on how well all three representations agree. There is no simple way to perform these comparisons because each representation has its own idiosyncrasies. For example, the real bow held by the violinist is, of course, the most accurate at being itself, but it is also the least reliable in accommodating a specific test. It is also completely useless for collecting multiple samples from the same test because the violinist cannot exactly repeat the same actions. In other words, the violinist would contribute the most errors and overshadow the errors in the IMU itself, which are the true interest. The physical model improves on these limitations, but it introduces inconsistencies that are not present in the real bow. Finally, the virtual model attempts to be a surrogate for both to tease out the performance

characteristics of the entire system piece by piece; i.e., which components contribute which errors and by how much. Configuring the models to reflect reality involved substantial trial and error.

The approach was to compare the performance of the bow IMU on the same tests across various combinations of the physical and virtual models. All tests used the same methodology of mapping inputs to outputs and measuring error as expected versus actual output. Specifically, in the perfect world, the same input (cause) would map to the same output (effect) every time. In practice, however, a multitude of factors introduced many errors that could not be isolated directly. The following combinations were the first attempt at building an objective comparative framework, which is still a work in progress.

5.1 IMU against physical model (C–A)

The first type of experimental comparison took the obvious route and aimed to determine how well the data from the IMU on the bow in Figure 3a corresponded to where the test rig believed it actually was. For example, in the perfect case, the bow would go to the expected state of position (x,y,z) and attitude ($yaw,pitch$), and the IMU would indeed report exactly this actual state; i.e., no error. In reality, however, there are three types of states in play: (A) where the rig believes it put the IMU, (B) where the IMU actually is, and (C) where the IMU believes it is. Therefore, there are errors likely in the comparisons between states A and B, B and C, and A and C. The difference between A (input) and C (output) is supposed to reflect the IMU error, but in fact, it really includes errors from all three. Some of these errors are additive (i.e., two wrongs make a bigger wrong), and some are subtractive (i.e., two wrongs make a smaller wrong or accidentally even a right).

This system is heavily underdetermined, and no amount of comparison can completely overcome having more unknowns than knowns, as well as a lack of confidence in the truth of the knowns. Establishing (as much as possible) which states contribute which errors and how they combine is the purpose of the next four similar (and admittedly confusing) types of experimental comparisons. To reiterate, this C–A test above intended to measure the IMU performance by comparing state C and A, but it actually does C and B because A and B are not the same due to errors in the rig. (Comparisons are reflexive, so A against B is the same as B against A.)

5.2 Physical model against ground truth (A–B)

The second type of experimental comparison aimed to determine the performance of the physical model against the best representation of reality — the ground truth. In other words, this comparison was of state A against B to discover errors in the rig. These tests used the calibration chamber in

Figure 6 in combination with three digital cameras to provide front, side, and top perspectives. The grid on the background permitted accurate visual measurement of the actual state of the IMU by hand. An LED on the test rig indicated when the test started and ended so the three streams could be synchronized. The tests were the same as earlier: command the bow to an expected state, measure its actual state, and report the error.

While this approach produced the best results on the true performance of the test rig, it was totally impractical for real tests. The image processing was a very tedious manual effort of isolating the IMU against the background grid in all three perspectives, translating the coordinates, then calculating the corresponding position and attitude. Furthermore, only the start and end states were available this way, limiting this approach to static tests only.

5.3 IMU against ground truth (C–B)

The third approach compared the IMU (C) against the ground truth (B). These tests used the same conditions as those in Section 5.2 and actually occurred at the same time. For the same reasons, they were utterly impractical for real-time tests, but they did provide more insight into the nature of the errors throughout the system.

5.4 Virtual model against ground truth (D–B)

The fourth approach compared the virtual model (D) against the ground truth (B). This process involved grounding the virtual model to match the physical model, including its errors. Each error source in Section 3.2 is actually a range from minimum to maximum with a probability distribution (typical uniform or Gaussian). Endless trial and error resulted in values that produced the same general behavior as the physical model on the limited number of data points captured.

5.5 Virtual model against physical model (D–A)

The fifth and final approach compared the virtual model (D) against the physical model (A). The values painstakingly processed in Section 5.2 served as the training data, where the actual results could be tweaked until they reasonably matched the expected results. The correspondence was generally good because it is not a fair measure of performance to know both the questions and the answers in advance. The true measure is how well the virtual model performs on tests that it has not yet seen, which Section 6 addresses.

5.6 Recap

The previous subsections capture five of the six possible comparisons in Table 1. *PM* and *VM* stand for physical and virtual model, respectively. The qualifier *believed* refers to where the device reports itself to be, whereas *actual* is

where it truly is. (*PM actual* corresponds to the omitted *IMU actual* because they are connected at the same location and would have the same values.) Comparison C–D is not an option at this point because it would require modeling the IMU and its own errors, which is far outside the scope of this work.

Table 1: Summary of comparisons

Section	Types		Description
5.2	A	B	PM believed vs. PM actual
5.1	A	C	PM believed vs. IMU believed
5.5	A	D	PM believed vs. VM actual
5.3	B	C	PM actual vs. IMU believed
5.4	B	D	PM actual vs. VM actual
	C	D	IMU believed vs. VM actual

6 Results and discussion

There was nothing elegant about the tests: they were pure brute force. This approach was actually convenient, however, because it mitigated the curse of dimensionality, which rapidly expands the test space into an intractable number of combinations as the number and range of test parameters increase [5]. Looping over the combinations in the virtual model was by intent very fast. The physical model, on the other hand, was relatively slow (and self-destructive over time as the rig wore out), but it was still immeasurably more effective than a violinist attempting such tests repeatedly.

The first category of tests involved static snapshots of the final state of the IMU after all kinematic actions had completed. Each test of the physical model involved averaging 10 independent runs, and on the virtual model, 100. (Strictly speaking, the number of runs should be the same, but the physical model would not have survived a higher value, and the probability-based virtual model would have suffered from a lower one.) Each run started from the same initial conditions. Angle parameters increased by 10 degrees per test. Bow extension increased by 10 centimeters, as measured from the pinion gear to the IMU.

Two subcategories considered parameters independently and in combination. The independent tests were:

1. yaw 0° and extension 10cm, pitch –30° to +30°; 7 tests
2. pitch 0° and extension 10cm, yaw –45° to +45°; 10 tests
3. yaw and pitch 0°, extension 10cm to 40cm; 4 tests

The combinational tests were:

4. yaw 0°, pitch –30° to +30°, extension 10cm to 40cm; 28 tests
5. yaw –45° to +45°, pitch 0°, extension 10cm to 40cm; 40 tests
6. yaw –45° to +45°, pitch –30° to +30°, extension 10cm to 40cm; 280 tests

As this paper is about using modeling and simulation in support of other work, this discussion primarily addresses the methodology, not the actual results per se. Czoski [3] covers the IMU performance in great detail. Space limitations also prevent further analysis. Tests 1–6 were static tests because they reported the final state only. The independent variants (1–3) were acceptable, whereas the combinational ones (4–6) were very inconsistent because of amplified errors. The second set of tests, 7–12, were respectively 1–6 again, but with intermediate states sampled at 10ms intervals. Figure 8 is a small representative example (based on Test 8) that shows how the expected versus actual states translated to error measurements. This graph considers only accuracy (i.e., how closely they agree); precision (how repeatable they are) and variance (spread, in terms of standard deviation) are also useful, along with many other statistical measures.

Unfortunately, while the virtual model corresponds reasonably well to the physical model in the static and dynamic independent tests and static combinational tests, it does not come close to reflecting the chaotic operating characteristics of the dynamic combinational tests. These tests are unfortunately the most representative of a violin and bow in actual use. It is questionable whether improving the virtual model would even be worthwhile because the physical model is so problematic.

7 Future work

For more meaningful and useful results, a better test rig is undeniably necessary. As a proof of concept, this one served its purpose, but it introduced far too many problems that unnecessarily complicated all aspects of this work. Likewise, better IMUs are needed. It is also likely that a second one on the other end of the bow might help correlate the raw data to mitigate some of the errors. Similarly, a more practical approach to establishing the ground truth via automated image processing could improve some of the convoluted inferences on A through D. Finally, full motion capture on a violinist could provide even more potential for appropriate grounding.

8 Conclusion

The goal of doing engineering on the cheap with commercial off-the-shelf components was reasonable, but no amount of basic modeling and simulation appeared to be on a promising track to compensate adequately for the many combinations of inherent errors throughout the system. Isolating a single type of error was indeed achievable, but in this messy, highly underdetermined environment, the final results collectively were unfit for actual use. Nevertheless, as a proof of concept, this work overall demonstrated that an integrated framework of modeling, simulation, visualization, and analysis successfully supports repeatable, controlled experiments in the otherwise intractable realm of real-time data collection and processing for complex violin movement. The widespread use of IMUs in countless other applications could benefit from this approach, as well.

9 References

- [1] Adapted from Google Sketchup Warehouse, 3dwarehouse.sketchup.com, last accessed Mar. 17, 2016.
- [2] Caron, F., E. Duflos, D. Pomorski, and P. Vanheegehe. *GPS/IMU data fusion using multisensor Kalman filtering: introduction of contextual aspects*. Information Fusion, vol. 7, no. 2, pp. 221–230, June 2006.
- [3] Czoski, J. *A Violin Practice Tool Using 9-Axis Sensor Fusion*. Masters thesis, Eastern Washington University, 2015.
- [4] Tappan, D. *A Pedagogy-Oriented Modeling and Simulation Environment for AI Scenarios*. WorldComp International Conference on Artificial Intelligence, Las Vegas, NV, July 13–16, 2009.
- [5] Thomopoulos, N. *Essentials of Monte Carlo Simulation: Statistical Methods for Building Simulation Models*. Springer: New York, 2012.
- [6] www.trossenrobotics.com, last accessed Mar. 20, 2016.
- [7] www.arduino.cc, last accessed Mar. 20, 2016.
- [8] www.pololu.com, last accessed Mar. 20, 2016.
- [9] www.sparkfun.com, last accessed Mar. 20, 2016.
- [10] www.raspberrypi.org, last accessed Mar. 20, 2016.

Multi-domain Unified Modeling of High Speed Motorized Spindle Water Cooling System Based on Modelica

Chao Nie , Zhihua Li , Huiyi Zeng

School of Mechanical Engineering, Hangzhou Dianzi University, Hangzhou 310018, Zhejiang, China

Abstract: *To build a model with high accuracy and better reusable ability for the high speed motorized spindle water cooling system (HSMS-WCS), a multi-domain unified modeling method based on Modelica was proposed. After obtaining the coupling block diagram based on the analysis of the HSMS-WCS, thermal analysis of the high speed motorized spindle (HSMS) were applied to calculate its heat power and heat transfer power. Then a multi-domain unified model based on Modelica for the HSMS-WCS was finally built. Further more, influences of the cooling system initial temperature and its channel section axial size to the spindle temperature rise were discussed respectively. Results show that the proposed model can better reflect the complex coupling relationship between the subsystems of the HSMS-WCS and get a good simulation results.*

Keywords: water cooling system; multi-domain unified modeling; coupling relations

1 Introduction

HSMS has been widely used in high grade CNC machine tools. As the core component of the machine tool, its working performance has an important effect on the machining accuracy. Usually spindle motor is built in the HSMS, so its heat dissipation condition is poor. Due to the significant importance of reasonable spindle temperature rise[1-2], it is necessary to carry out a comprehensive analysis on the HSMS-WCS and build its model to provide a useful reference for the latter optimization design.

The modeling method for the HSMS-WCS is of great significance, scholars had made lots of researches on it. It can be roughly divided into two categories: the first was just for one single field and much attention was paid to this category. Chen et al. [3] had a 3D simulation

and analysis of the experiment by Ansys CFX and then undertook a comparison for the temperature rise of the spindle under the conditions of different working conditions and environment temperature. He et al. [4] had used finite element method to characterize the heat distribution of the HSMS and the final analysis showed that temperature rise could be significantly reduced with the application of cooling system. Rui et al. [5] had studied the different cooling effects for the HSMS-WCS under various working conditions with the use of orthogonal test method and then obtained the relationship between the coolant flow rate and spindle temperature rise. The weakness of the first category was based on the fact that dynamic performances of the HSMS-WCS were decided by the coupling relations of different subsystems. If the natural coupling relations between each subsystem were separated, that may lead to low precision and can not reflect the correction of the model. The second category was that different subsystems of collaborative simulation were realized through the interface technology between different software, but it can result in poor coupling relations as a result of the difficulty with achieving a seamless data transfer despite the fact that it can solve the problem to some extent. Such kinds of papers are as follows: Ford motor company had utilized ADAMS and Xmath to acquire the simulation model of Vehicle attitude control system[6]; Visteon company using ADAMS and MATLAB software to develop torque controller[7]etc. Such above methods need to be rebuilt when the model was locally modified and the similar model can not be used before, which made the model less efficient. This is the problem of poor reusable ability for the model.

In summary, it is still to be improved in model precision and reusable ability. In this paper, a multi-domain unified modeling method based on Modelica[8] was presented. This language is a kind of multi-domain unified modeling language based on equations, in which all the models are established by one language, so that the coupling between the subsystems can be realized and the modeling efficiency is high. By using the MWorks[9] software based on Modelica language, the seamless connection between the subsystems of the HSMS-WCS can be reached and the problems caused by the integration of different software can be overcome. At last, a multi-domain unified model was obtained with high accuracy and better reusable ability, which is an innovative method.

2 Channel structure and coupling analysis

2.1 Structure of cooling channel

Spindle motor built in the HSMS with spiral cooling channel was presented in this paper. The built-in motor is located in the middle of the water cooling system and the power loss is converted into heat to heat this area. Cooling system of the HSMS generally adopts the circulating water containing additive, corresponding to the pipeline and temperature control device. The coolant circulates in the spiral cooling channel that is a rectangular slot. Its cross sectional area can be expressed as A . Structure of the cooling channel is as shown in Fig.1.

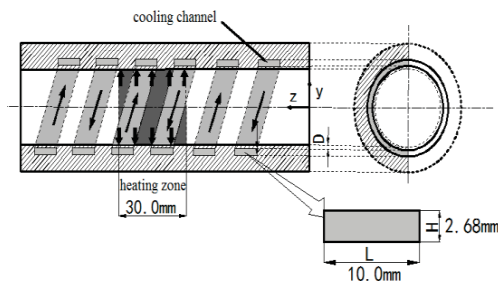


Fig.1 Structure of the cooling channel

2.2 Coupling analysis

Performance of the HSMS-WCS is not only determined by several input parameters of single

subsystem, but also subjected to the impact of other subsystems that have the coupling relations. Therefore the global coupling analysis of the HSMS-WCS should be carried out before building a multi-domain unified model. Firstly, each subsystem that influences the HSMS-WCS should be clearly defined; Then the global coupling relations of the HSMS-WCS was decomposed into several subsystems which have local coupling relations, so that each subsystem can independently accomplish their physical functions; After that, the coupling parameters and the physical relationship were discussed between the subsystems. At the end, all the subsystems were connected by a specific interface to form a global coupling block diagram. It is given in Fig.2.

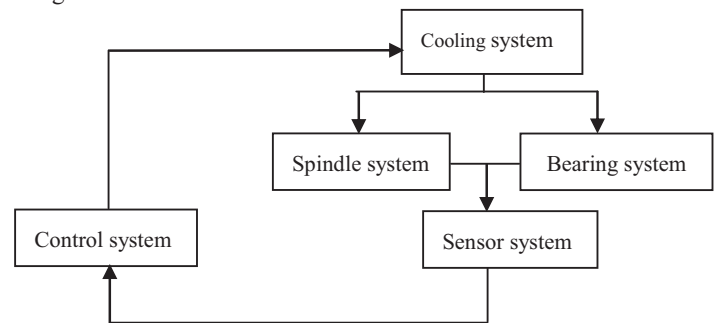


Fig.2 Global coupling block diagram of the HSMS-WCS

As shown in Fig. 2, the HSMS-WCS is involved in multiple subsystems such as mechanical system, control system, thermal system and so on, which is a multi-domain modeling problem.

3 Thermal analysis for the HSMS

Without taking into account the production of cutting heat from the machining process, spindle system mainly includes two kinds of heat sources. The former is from the spindle motor and the latter is from the spindle bearings.

3.1 Heating power of the spindle motor

The heating of the spindle motor is derived from the power loss of motor, which mainly includes mechanical loss and electrical loss.

3.1.1 Mechanical loss

Mechanical loss is caused by the friction loss between the rotor and air when the rotor is running at

high speed. It is defined as:

$$p_m = \frac{\pi}{16 \times 102} c_1 c_2 \omega^3 D^4 L \quad (1)$$

Where p_m is mechanical loss (w), c_1 is the coefficient of flow resistance, c_2 is air density (kg/m^3), ω is rotor angular velocity (rpm), D and L are the diameter and length of rotor respectively (m).

3.1.2 Electrical loss

The electrical loss is mainly from the loss between the stator coil and rotor, which can be computed as:

$$P_e = I^2 R = I^2 c_e \frac{L_e}{S_e} \quad (2)$$

Where p_e is electrical loss (w), I is stator coil current (A), c_e is electrical conductivity of coil, L_e is total length of the coil (m), S_e is sectional area of the coil (m).

3.2 Heating power of the spindle bearing

Heating power of the angular contact ball bearings is mainly from loss due to bearing friction torque, which can be calculated as:

$$p_f = 1.047 \times 10^{-4} \omega M \quad (3)$$

Where p_f is heating power of the spindle bearing (w), ω is rotor angular velocity (rpm), M is total friction torque (Nmm).

The bearing friction torque is mainly composed of two parts. One is the torque M_z due to applied load and the other is the torque M_y due to viscosity of lubricant,

that is

$$M = M_z + M_y \quad (4)$$

Where

$$M_z = f_z F_z d_m \quad (5)$$

Where M_z is load friction torque (Nmm), f_z is a factor related to the bearing type and load, F_z is bearing preload (N), d_m is mean diameter of the bearing (mm).

$$M_y = 10^{-7} f_y (v_y n)^{\frac{2}{3}} d_m^3, v_y n \geq 2000 \quad (6)$$

$$M_y = 160 \times 10^{-7} f_y d_m^3, v_y n < 2000 \quad (7)$$

Where M_y (Nmm) is viscous friction torque of the angular contact ball bearings, it is generated by the relative friction among the rolling elements of the bearing, the cage, and the lubricant, f_y is a factor related to bearing type and lubrication method, v_y is kinematic viscosity of the lubricant (mm^2/s), n is the bearing angular velocity (rpm). When $v_y n \geq 2000$, formula (6) can be used and $v_y n < 2000$, formula (7) can be used.

3.3 Convective heat transfer of coolant

In addition to the influences of these two heating powers, the HSMS-WCS performances are also affected by the heat convection of cooling system. In order to make the research process simple and clear, some necessary simplification and assumptions can be made. (1) The main research object was the cooling system, so the rotor, bearing and other parts were simplified; (2) It was assumed that the heating power produced by the spindle motor and spindle bearings was forced to take away through the heat convection, further more, the surrounding natural convection and thermal radiation was negligible. (3) Coolant can not be compressed and the physical properties are constant. There is no phase change

and it is a continuous body.

3.3.1 Convective heat transfer coefficient of cooling system

Cooling system uses coolant for cooling, in which the average speed of the coolant can be expressed as follows:

$$u_w = \frac{v_w}{A} \quad (8)$$

Where v_w is unit flow rate of coolant (l/s), A is cross sectional area of the cooling channel (m^2).

Convective heat transfer coefficient of cooling system can be written as:

$$\alpha_v = \frac{N_u K_v}{d_v} \quad (9)$$

Where N_u is Nusselt number, K_v is heat conductivity of coolant ($w/m \times k$), d_v is diameter of cooling channel (m).

3.3.2 Convective heat transfer power of cooling system

The convective heat transfer power of cooling system can be calculated by the convective heat transfer coefficient, which can be established as:

$$p_v = \alpha_v A (t_v - t_0) \quad (10)$$

Where α_v is convective heat transfer coefficient ($w/m^2 \times k$), A is cross sectional area of the cooling channel (m^2), t_v is spindle internal temperature (k), t_0 is initial temperature of coolant (k).

4 Multi-domain unified modeling of the HSMS-WCS

Considering the complex coupling relations, a

multi-domain unified model for the HSMS-WCS based on the Mworks software was built. Subsystems of the HSMS-WCS include spindle motor heating model, bearing heating model, cooling system convection model and PID control model. The final multi-domain unified model was built on the same software and connected by a specific interface.

4.1 PID control model

The temperature rise of spindle must be controlled within a reasonable range, otherwise it will affect spindle machining accuracy. In this paper, PID control system was adopted to control the temperature rise. PID control equation is as follows:

$$u = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (11)$$

Where $e(t)$ is the input signal, it is defined as:

$$e(t) = NT_d - NT \quad (12)$$

Where $e(t)$ is the input signal, NT_d is the desired operating temperature, NT is the actual operating temperature, K_p, T_i, T_d are the control parameters.

The working process of PID controller was briefly as follows: The temperature sensor detected the actual operating temperature. Through feedback, the input signal was transmitted to the PID controller. The adjusted current signal was output to the cooling system model after treatment, which made the temperature rise in a reasonable value. PID control model based on Modelica is shown in Figure.3.

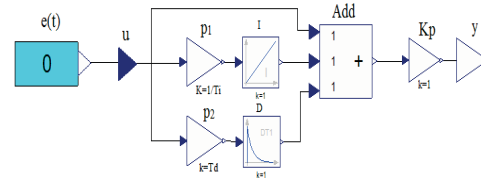


Fig.3 PID control model

4.2 Heating power model and convective heat transfer power model

The heating power model of the HSMS-WCS consisted of two parts: one for the spindle motor and the other for the spindle bearings, which can be built in Mworks by mathematical model respectively. Convective heat transfer power model was also modelled by its

mathematical model whose function is to take the heat away. After the two models were finished, a coupling model based on Modelica was built, which is shown in Figure. 4.

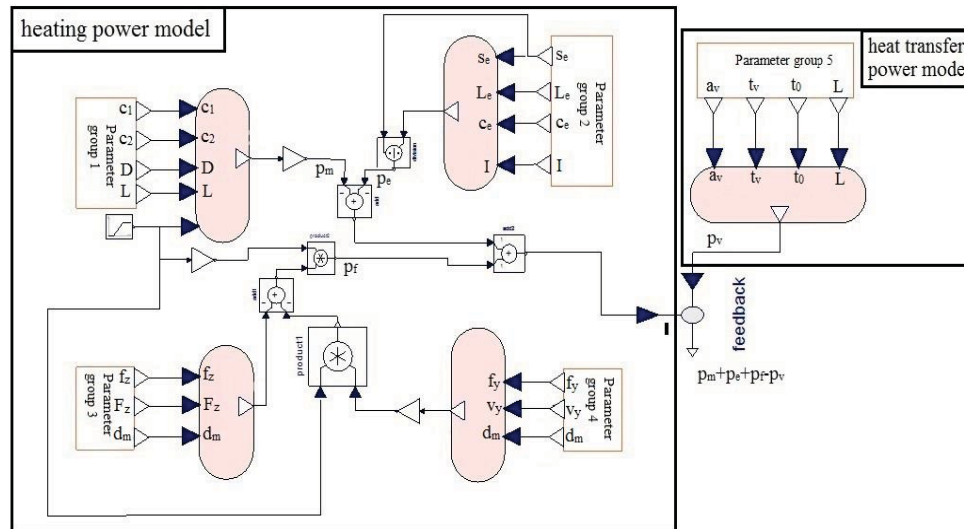


Figure.4 Coupling model

4.3 Integrated model of the HSMS-WCS

Using Modelica language to construct models at Mworks platform, each sub-model is independent of each other on describing the physical characteristics and mathematical relations. In order to ensure the reusable ability of sub-models, each sub-model has its own input and output connectors to input parameters and communicate with other sub-models. Meanwhile, a sub-model can be composed of other sub-models and components by packaging, which is the multi-level modeling method. In this paper, the heating power model, convective heat transfer power model and control model were packaged respectively and then connected by specific interface. Therefore the multi-level modeling method has the advantages of clearly organized, better reusable ability and high modeling efficiency etc.

The process of integrated modeling was as follows: At first, the temperature sensor model obtained the actual operating temperature of spindle system and then the signal difference between NT_d and NT was fed back to the PID control model. Finally NT can reach a reasonable value when the output current control signal

was transmitted to the PID control model. By packaging and connecting all of the sub-models at Mworks, the integrated model was obtained and shown in Figure.5.

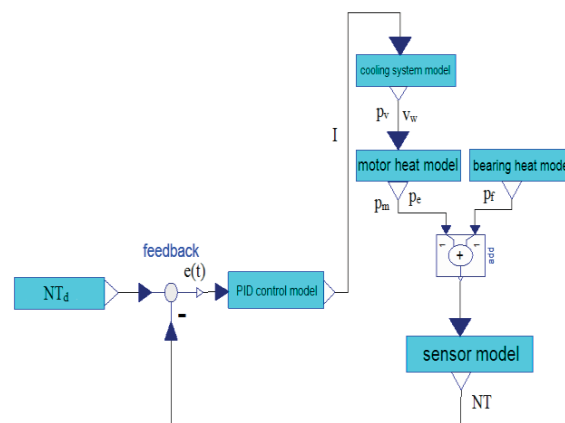


Figure.5 Integrated model of the HSMS-WCS

5 Simulation analysis of the HSMS-WCS

Typically, the HSMS-WCS can reach the cooling effect by controlling the rate of coolant flow, but such kinds of researches are more. Previous researches [10-11]

indicated that: the temperature rise of spindle does not make significant change when coolant flow rate increases to a certain threshold, therefore the method may not reach the desired goal in sometimes. Based on the above, this paper had explored the other two factors affecting the temperature rise of spindle.

5.1 The factor of initial temperature t_0

To investigate the effects on the spindle temperature rise, parameters related to the initial temperature were input to the proposed model under the conditions of keeping the spindle angular velocity and coolant flow unchanged. At the same time, another two traditional models including Ansys method and Simulink method were used to compare. Finally, the simulation results are as shown in Figure. 6.

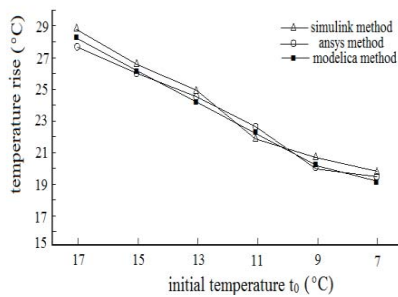


Figure.6 Initial temperature effect on the spindle temperature rise

Figure.6 can be obtained: (1) the spindle temperature rise is influenced by the initial temperature and the

Table 1 Channel section axial size L							(mm)
Channel section axial size	1	2	3	4	5	6	7
L	8.50	9.00	9.50	10.00	10.50	11.00	11.50

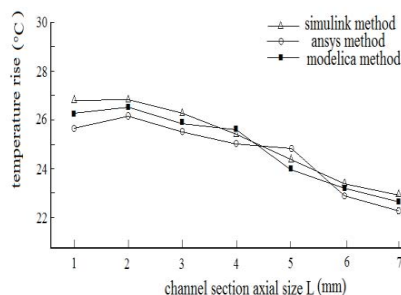


Figure.7 Channel section axial size effect on the spindle temperature rise

spindle temperature rise decreases with gradually decreasing the initial temperature; (2) when compared to another two traditional models, the presented model can acquired the similar simulation results. This can ensure that the model built by multi-domain unified modeling method is right. Further more, it has a better reusable ability when compared to the model by Ansys and can make it more efficient to build a simulate model. Simulink method has the difficult to achieve a seamless data transfer. This weakness can lead to a low precision for the model built. While the proposed method is based on Modelica language, it can reach the seamless connection between the subsystems, ensuring a high accuracy model.

5.2 The factor of channel section axial size L

Not only is the spindle temperature rise related with coolant initial temperature, but also associated with the channel section axial size L . Similar to the method above 5.1, the related initial parameters keep constant and the channel section axial size L changed without changing the change cross-sectional area A . The channel section axial size L is shown in Table 1 and the final simulation result is as shown in Figure.7.

Figure.7 shows that channel section axial size L has influence on the spindle temperature rise and the spindle temperature rise decreases with gradually increasing the channel section axial size L .

6 Conclusions

When constructing model of the HSMS-WCS, it was generally divided into two categories. One is the modeling method for one single field and the other is the method for multi-domain integration. It is not suitable to use the first category, because it separates the natural

coupling relationships between subsystems, which can not reflect the complex coupling relations. The second category may lead to problems such as poor coupling relations and low precision despite the fact that it can realize multi-domain unified modeling for multiple subsystems to some extent. A multi-domain unified modeling method based on Modelica was presented in this paper. At first, the complex coupling relations among subsystems of the HSMS-WCS were analyzed and the coupling block diagram was obtained; then the thermal analysis of the HSMS was carried out. Based on this, a multi-domain unified model of the HSMS-WCS was built and the simulation results were analyzed. The final results show that the method based on Modelica can effectively overcome the problems brought by the previous method and make the model more accurate and suitable for the modeling and simulation of complex mechanical and electrical products.

Acknowledgements

This study was supported by National Natural Science Foundation of China (Grant No.51275141).

References

- [1] Moorthy, R.S.; Raja, V.P.: An improved analytical model for prediction of heat generation in angular contact ball bearing. *Arabian Journal for Science and Engineering*. 39(11), 8111–8119 (2014)
- [2] Bian, W.; Wang, Z.H.; Yuan, J.T.: Thermo-mechanical analysis of angular contact ball bearing. *Journal of Mechanical Science and Technology*. 30(1), 297-306 (2015)
- [3] Chen, W.H.; He, Q.C.; He, Q.: Simulation and Experimental Analysis for High- speed Spindle with Water- cooling System. *China Mechanical Engineering*. 21(5), 550-555 (2010)
- [4] He, Q.; Zhang, Y.; Ye, J.: Thermal characteristics of high speed motorized spindle with helical water cooling channel. *Recent Patents on Mechanical Engineering*. 5(1), 69-76 (2012)
- [5] Rui, Z.Y.; Chen, T.; Lei, C.L.: Simulation analysis for water cooling system of high-speed motorized spindle based on CFX. *Machine Tool&Hydraulics*. 42(7), 24-28 (2014)
- [6] BMS. Co-simulation boosts vehicle design efficiency at ford. *Computer Aided Engineer*, 1999, 18(7):8-9.
- [7] C.S. Liu, V. Monkaba, et al. Co-simulation of Visteon driveline torque bias controls. *Adams User Conference 2001-North America*, Detroit, USA, 2001:248.
- [8] Zhao, J.J.; Ding, J.W.; Zhou, F.L.: Modelica and its mechanism of multi-domain unified modeling and simulation. *Journal of System Simulation*. 18(2), 570-573 (2006)
- [9] Wu, Y.Z.; Wu, M.F; Chen, Y.P.: Study on the hybrid modeling platform based on Modelica language for complex machinery system. 17(22), 2391-2396 (2006)
- [10] Zhang, J.F.; Feng, P. F.; Chen, C.; Yu, D.W.: A method for thermal performance modeling and simulation of machine tools. *The International Journal of Advanced Manufacturing Technology*. 68(5), 1517-1527 (2013)
- [11] Yang, S.Y.; Gao, X.H.; Liu, X.X.: Simulation on solid-fluid coupled heat transfer of water cooling system in high-speed electro-spindle. *Machine Tool&Hydraulics*. 39(11), 102-110 (2011)

On the Development of Models and Metrics for Safety of Soldiers

Rula Twal*, Amjad F Almatrood[†] and Harpreet Singh[‡]

Electrical and Computer Engineering, Wayne State University

Detroit, MI 48202

Email: *rula.twal@wayne.com, [†]amjad.almatrood@wayne.edu, [‡]hsingh@eng.wayne.edu

Abstract—In this paper a model for the safety of soldiers is developed. The model is based on a combination of fuzzy and binary techniques. For the safety of soldiers, the metrics for the model are developed. Based on the questionnaire given to soldiers and the responses of the soldiers, the safety is defined between 0 – 1. This date is simulated in the fuzzy model and digitalized to convert it to its equivalent digital model. Both digital and fuzzy technique are compared. It is hoped that such models will help a long way in improving the safety of the soldiers.

I. INTRODUCTION

Safety of soldiers has always been a major concern. Billions of dollars are being spent in the safety and quality of life for soldiers. Safety of soldiers is very important both at the time of war and peace. One of the first requirements for the soldier is to have safety equipments such as helmet. So many research workers have been working on the development of the new designs for safe helmets. Alexandera Foran [1] with NSRDEC Public Affairs has written an extensive report on the protection for soldiers. The military personal protective equipment covers a range variety of garments such as glasses, belts, gloves, helmets, shoes [2]–[4]. These are crucial and required for the soldier's safety. Various conferences are held every year at different locations for improving the safety of soldiers. There are a number of shows also held every year for the soldier's safety products. Grainger show is one of the important shows that is held every year to display many safety products [5]. Similarly, military combat eye protection is required for soldiers while on the job [6]. A report on advanced combat helmet technical assessment has been prepared by the department of the defense [7]. To the best of the author's knowledge no algorithm is available which describes the safety of soldiers in a unified way.

The objective of this paper is to develop an algorithm which can predict the safety of the soldiers. Similarly there is no metrics available which can describe how is the safety of soldiers is being measured. In this paper we give an algorithm for measuring the safety of the soldiers. The strategy is to develop a questionnaire which can give a different responses from different soldiers. The questions will be regarding the safety of the helmets, belts, gloves, glasses, and shoes. A soldiers is asked how good a particular parameter is? For example a helmet. Just like a doctor asks how much patients pain is from 0 – 10.

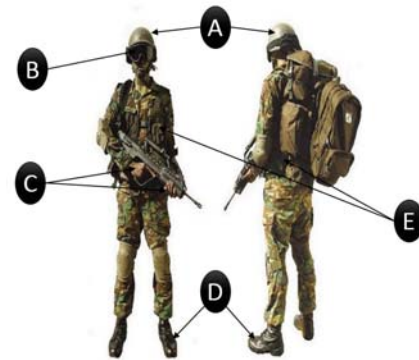


Fig. 1. Five safety features: (A) helmet, (B) glasses, (C) gloves, (D) belt and (E) shoes.

II. METHODOLOGY

The proposed algorithm is consist of the following steps:

- 1) Determine the membership functions for our 5 inputs which are helmets, glasses, gloves, shoes and one output which is the soldiers safety.
- 2) Define fuzzy rules for the fuzzy model.
- 3) Implement the fuzzy model using MATLAB.
- 4) Develop a table having n rows and 5 columns where n is the number of soldiers and each column represents a parameter such as helmets, glasses, gloves, shoes and belt. The truth table should be a 3 out of 5 majority function.
- 5) Simulate the data given in the table developed in step 4.
- 6) Digitalize the values given in step 4 in the form of 0 and 1.
- 7) Develop the Boolean expression for the function and then realize it with help of logic gates.
- 8) Draw the binary decision diagram BDD [8] from the truth table.
- 9) Write the Boolean expression in hardware description language HDL and test it using FPGA.
- 10) Compare the results of fuzzy model along with the digital model.

Fig. 1 shows the important safety features that a soldier should be asked about. These are helmet, glasses, gloves, belt and shoes. For simplicity we have taken only five parameters. The approach can be extended to any number of features.

TABLE I
MEMBERSHIP FUNCTIONS FOR INPUTS AND OUTPUT

Inputs	Membership Functions		
	Off	Damaged	On
Helmet	0 – 0.4	0.1 – 0.9	0.6 – 1
Glasses	0 – 0.4	0.1 – 0.9	0.6 – 1
Gloves	0 – 0.4	0.1 – 0.9	0.6 – 1
Belt	0 – 0.4	0.1 – 0.9	0.6 – 1
Shoes	0 – 0.4	0.1 – 0.9	0.6 – 1
Output	Low	Medium	High
Soldiers Safety	0 – 40	10 – 90	60 – 100

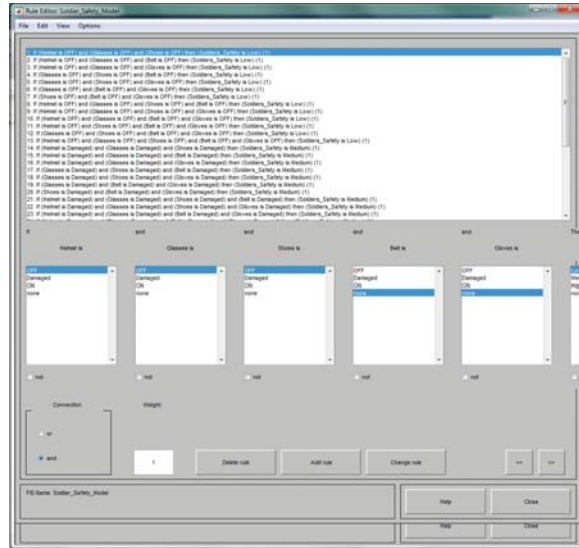


Fig. 2. Fuzzy rules.

III. IMPLEMENTATION

The implementation of all the steps is described as follows:

- 1) In this step three membership functions are considered for each inputs. These functions are defined as On, Damaged and Off. For the output (soldiers safety), the same number of membership functions are considered and defined as Low, Medium and High. The membership functions for each input and output is shown in Table I.
- 2) The fuzzy rules for the design are defined as an equivalent to a majority function. These rules are defined as shown in Fig. 2. The rule viewer is shown in Fig. 3.
- 3) The fuzzy model is designed and implemented using MATLAB as shown in Fig. 4.
- 4) Develop a table which consist of columns having five inputs and rows consist of soldiers 1, 2, 3 etc. For a particular soldiers, a questionnaire is asked regarding how comfortable and safe a soldier feels about his/ her helmet, glasses, gloves, belt and shoes. The soldiers answered the questions in the range from 0 – 10. This data is then normalized from 0 – 1 as given in Table II. The data that we consider in this table is just

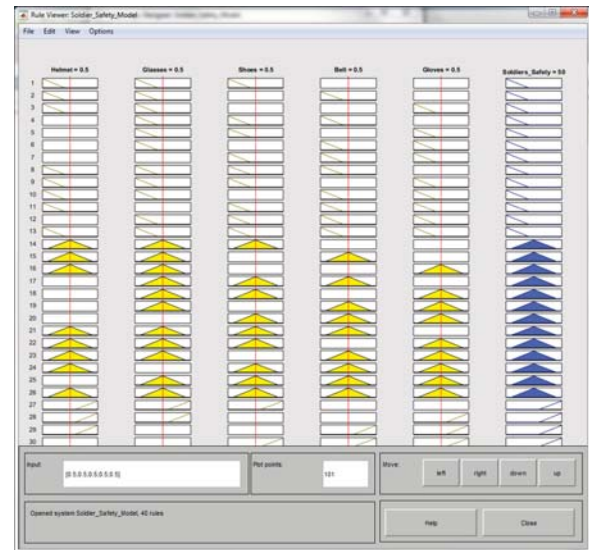


Fig. 3. Fuzzy rule viewer.

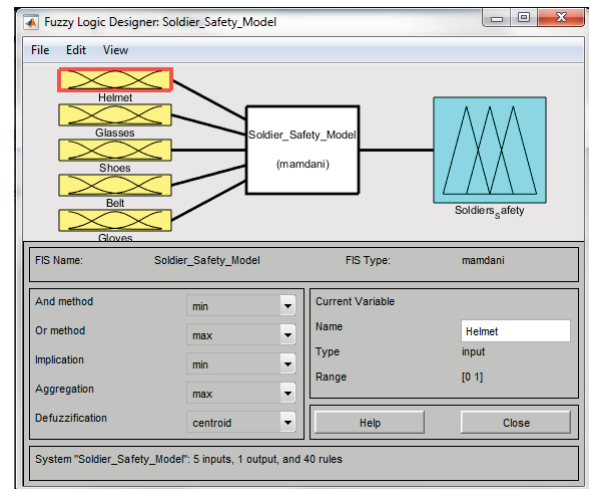


Fig. 4. Fuzzy model design.

hypothetical and arbitrary. The purpose of this data is just to describe the approach that we are suggesting.

- 5) Simulate the data in the table developed in step 3 in the designed fuzzy model. The surface viewer is shown in Fig. 5. The simulation results are given in Table III.
- 6) The table developed in step 4 is then digitalized. This results in a truth table as shown in Table IV.
- 7) From the truth table, it is noted that the model behaves as a majority function. Therefore, the obtained Boolean expression for 5-input majority function is given in (1).

$$F = ABC + ABD + ABE + ACD + ACE + ADE + BCD + BCE + BDE + CDE \quad (1)$$

- 8) Truth tables for large variables become almost impossible to write. Hence, the new alternative is to use binary decision diagram instead of the truth table. The binary

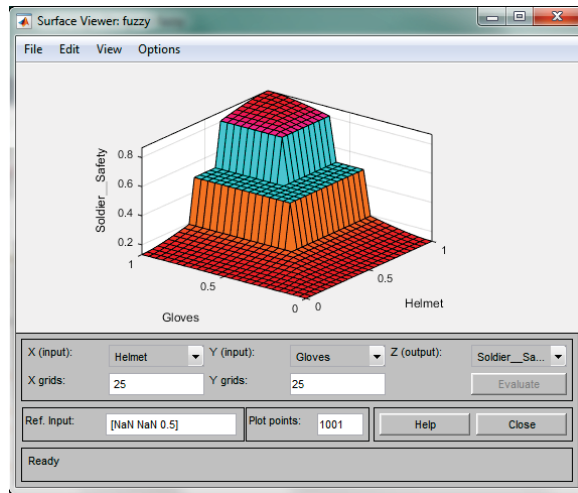


Fig. 5. Surface viewer.

TABLE II
TABLE OF FUZZY MODEL

Soldier	Inputs				
	Helmet	Gloves	Glasses	Belt	Shoes
1	0.90	0.11	0.17	0.18	0.78
2	0.86	0.11	0.87	0.84	0.86
3	0.10	0.10	0.20	0.12	0.60
4	0.31	0.88	0.78	0.95	0.15
5	0.86	0.84	0.15	0.87	0.86
6	0.88	0.11	0.15	0.93	0.16
7	0.88	0.16	0.86	0.16	0.79
8	0.21	0.80	0.30	0.20	0.80
9	0.77	0.76	0.54	0.55	0.30
10	0.10	0.23	0.87	0.99	0.10
11	0.12	0.11	0.87	0.20	0.12
12	0.88	0.79	0.89	0.87	0.13
13	0.21	0.21	0.97	0.10	0.88
14	0.11	0.88	0.44	0.93	0.66
15	0.45	0.16	0.77	0.80	0.27
16	0.12	0.20	0.12	0.98	0.10
17	0.86	0.80	0.87	0.12	0.14
18	0.30	0.32	0.11	0.87	0.96
19	0.12	0.90	0.87	0.12	0.86
20	0.33	0.11	0.65	0.14	0.41
21	0.10	0.90	0.40	0.84	0.27
22	0.49	0.66	0.80	0.11	0.60
23	0.23	0.78	0.31	0.40	0.25
24	0.89	0.14	0.13	0.84	0.94
25	0.13	0.76	0.78	0.88	0.82
26	0.85	0.11	0.87	0.13	0.16
27	0.88	0.84	0.11	0.20	0.30
28	0.23	0.56	0.59	0.32	0.17
29	0.88	0.87	0.86	0.99	0.86
30	0.20	0.20	0.21	0.11	0.301
31	0.31	0.82	0.33	0.89	0.79
32	0.85	0.15	0.88	0.94	0.11
33	0.88	0.79	0.23	0.76	0.13
34	0.74	0.17	0.87	0.37	0.27
35	0.90	0.87	0.86	0.13	0.83
36	0.88	0.89	0.23	0.11	0.87
37	0.20	0.30	0.88	0.89	0.80
38	0.89	0.55	0.56	0.91	0.59
39	0.84	0.12	0.13	0.14	0.15
40	0.11	0.78	0.88	0.22	0.33

TABLE III
SIMULATION RESULTS OF FUZZY MODEL

Soldier	Soldier Safety	Soldier	Soldier Safety
1	0.3124	21	0.5000
2	0.7661	22	0.5000
3	0.1992	23	0.5000
4	0.7707	24	0.7661
5	0.7507	25	0.6622
6	0.1799	26	0.2338
7	0.5000	27	0.3780
8	0.5000	28	0.5000
9	0.5000	29	0.7719
10	0.1812	30	0.3543
11	0.2121	31	0.6610
12	0.7477	32	0.5000
13	0.2088	33	0.5000
14	0.6764	34	0.4707
15	0.5000	35	0.7719
16	0.2215	36	0.8264
17	0.7292	37	0.6505
18	0.3332	38	0.5000
19	0.7962	39	0.2280
20	0.3763	40	0.5000

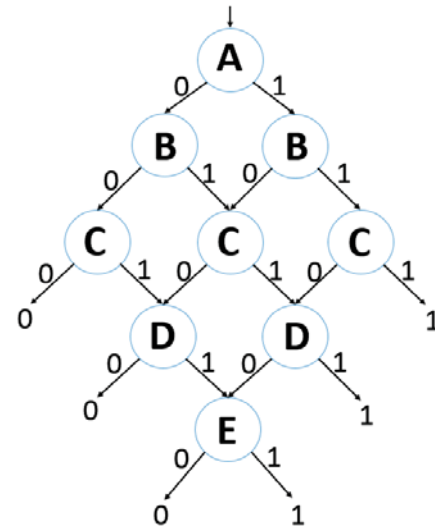


Fig. 6. Binary decision diagram for (1).

decision diagram for (1) is shown in Fig. 6. Binary decision diagrams are also used for testing large circuits. Table V gives the minimum BDD possible test cases for (1). This BDD is tested with 2 different cases. The first case is $A=0, B=1, C=0, D=1$, and $E=0$ which results in $F=0$. The second case is $A=1, B=0, C=1, D=0$, and $E=1$ which results in $F=1$. The binary decision diagrams for the first and second case are shown in Fig. 7 and Fig. 8 respectively.

- 9) The design is written in Verilog code and tested using Xilinx XC3S200 Spartan-3 FPGA. Fig. 9 shows One test case implemented in FPGA with the inputs $A=1, B=1, C=1, D=0$, and $E=0$ which result in an output $F=1$.
- 10) The fuzzy model is compared with the digital model using correlation function analysis in MATLAB. The correlation value obtained is 0.8778.

TABLE IV
TRUTH TABLE OF DIGITAL MODEL

Soldier	Inputs					Output Soldier Safety
	Helmet	Gloves	Glasses	Belt	Shoes	
1	1	0	0	0	1	0
2	1	0	1	1	1	1
3	0	0	0	0	1	0
4	0	1	1	1	0	1
5	1	1	0	1	1	1
6	1	0	0	1	0	0
7	1	0	1	0	1	1
8	0	1	0	0	1	0
9	1	1	1	1	0	1
10	0	0	1	1	0	0
11	0	0	1	0	0	0
12	1	1	1	1	0	1
13	0	0	1	0	1	0
14	0	1	0	1	1	1
15	0	0	1	1	0	0
16	0	0	0	1	0	0
17	1	1	1	0	0	1
18	0	0	0	1	1	0
19	0	1	1	0	1	1
20	0	0	1	0	0	0
21	0	1	0	1	0	0
22	0	1	1	0	1	1
23	0	1	0	0	0	0
24	1	0	0	1	1	1
25	0	1	1	1	1	1
26	1	0	1	0	0	0
27	1	1	0	0	0	0
28	0	1	1	0	0	0
29	1	1	1	1	1	1
30	0	0	0	0	0	0
31	0	1	0	1	1	1
32	1	0	1	1	0	1
33	1	1	0	1	0	1
34	1	0	1	0	0	0
35	1	1	1	0	1	1
36	1	1	0	0	1	1
37	0	0	1	1	1	1
38	1	1	1	1	1	1
39	1	0	0	0	0	0
40	0	1	1	0	0	0

IV. CONCLUSION

In this paper a combination of fuzzy and digital model has been developed. An approach for determining the safety of soldiers has been discussed. Furthermore, a metrics has been proposed in which the safety of soldier is defined between 0 – 1. For simplicity, only a few parameters are taken. The approach could be extended to any number of parameters. We hypothesize that the fuzzy model is approximated as 3 out of 5 majority function. In general, different Boolean functions such as majority functions can be considered. It is also illustrated how BDDs can be used for large number of variables. We compared both of the models and determined the correlation. The models seem to be quite satisfactory as long as the suggested approach could verified and validated by experts. Study based the response of the questionnaire from the soldiers is needed.

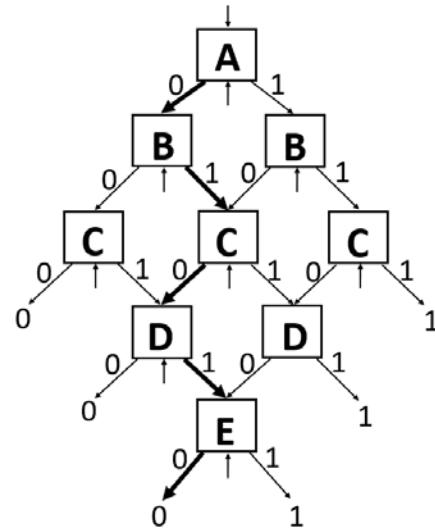


Fig. 7. First test case: the inputs are $A=0$, $B=1$, $C=0$, $D=1$, and $E=0$ which result in an output $F=0$.

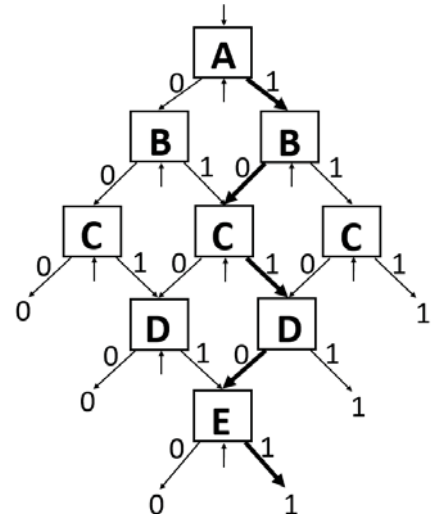


Fig. 8. Second test case: the inputs are $A=1$, $B=0$, $C=1$, $D=0$, and $E=1$ which result in an output $F=1$.

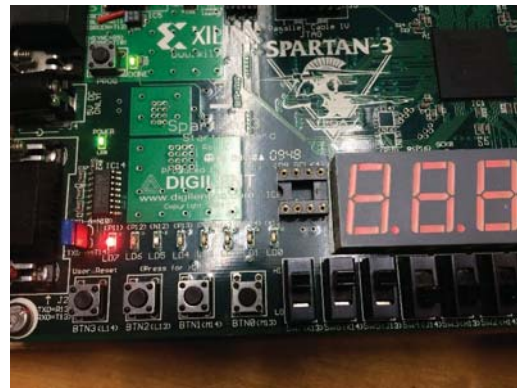


Fig. 9. One test case with the inputs $A=1$, $B=1$, $C=1$, $D=0$, and $E=0$ and an output $F=1$.

TABLE V
THE MINIMUM BDD POSSIBLE TEST CASES FOR (1)

A	B	C	D	E	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	0	0	0	0	*		*				*											
0	0	1	1	1	*		*					*						*				*
0	1	1	0	1	*			*						*					*		*	
0	1	1	1	1	*			*						*						*		
1	0	0	0	1		*			*				*				*					
1	1	0	0	1		*				*					*				*		*	
1	1	1	0	1		*				*						*						

REFERENCES

- [1] A. Foran, "Tactical protection for soldiers," *ARMY TECHNOLOGY MAGAZINE*, Orlando, FL. 2013.
- [2] J. Breeze, I. Horsfall, A. Hepper, and J. Clasper, "Face, neck, and eye protection: adapting body armour to counter the changing patterns of injuries on the battlefield," *British journal of oral and maxillofacial surgery*, vol. 49, no. 8, pp. 602–606, 2011.
- [3] M. R. Lattimore, "Combatant eye protection: An introduction to the blue light hazard," 2015.
- [4] T. Hughes, J. Williamson, A. Hess, W. Young, A. Dumas, K. Fischl, and B. Telfer, "Soldier protection benchmark evaluation (spbe) physiological data collection and analysis, fort greely, alaska, 17 september-5 october 2012," DTIC Document, Tech. Rep., 2013.
- [5] G. Show, "products, services, resources," *Orange County Convention*, 2012.
- [6] M. M. Thomas J Pojeta, P E and D. Phelps, "Military combat eye protection mcep program advanced planning briefing for industry apbi," Tech. Rep., 2011.
- [7] R. Stone, "Advanced combat helmet technical assessment," DTIC Document, Tech. Rep., 2013.
- [8] S. B. Akers, "Binary decision diagrams," *Computers, IEEE Transactions on*, vol. 100, no. 6, pp. 509–516, 1978.

A Model for Commodity Hedging Strategies

Sakir Yucel¹ and Ibrahim Yucel²

¹yucel@bluehen.udel.edu

²iyucel@colgate.edu

Abstract - Commodity prices are known to be volatile in general. The volatility of the commodity prices brings up challenges for the producers that use such commodities in their production. One challenge is to determine what prices companies will pay via different hedging strategies for the needed commodities over the production period. This paper presents a systems dynamics model that incorporates various dynamics for commodity market. This model is then used for developing an algorithm to simulate hedging strategies. How the system dynamics model along with the hedging simulation algorithm can help the producers with their hedging decisions is discussed.

Keywords: commodity market, hedging strategies, system dynamics modeling

1 Introduction and Problem Definition

Producers need different commodities in order to produce their products. Many kinds of commodities may go into a certain product. For example, a frozen food production requires the purchase of numerous food inputs, including wheat, sugar, poultry and flash-frozen vegetables, as well as various packaging materials. Key commodity inputs may come from a variety of suppliers and generally have different seasonal and cyclical price characteristics.

Commodity prices are known to be volatile [5] [7] [8]. The volatility of the commodity prices brings up challenges for the producers that use such commodities in their production. Fluctuations in the price of key inputs make it difficult for manufacturers to anticipate future costs and optimize production, which constrains profitability. Furthermore, producers that must purchase a wide range of commodities with differing supply/demand dynamics may face difficulty managing effective procurement and commodity hedging strategies.

In competitive markets, some producers may be able to purchase certain inputs at lower prices than others and subsequently reflect this difference in the price of their product. In turn, greater pricing flexibility gives the producer a competitive advantage over other producers, allowing it to gain market share and sustain greater gross margins as a result of lower purchase costs.

Producers continually seek ways to smooth volatility in input prices, more accurately forecast future costs or otherwise minimize the risk associated with fluctuating commodity prices. In order to overcome commodity price fluctuations, companies are using hedging for managing the risks that come due to volatile commodity market. Hedging has become a useful tool for producers to manage what prices they will pay for their raw materials over the production period. However, since a number of different contracts may exist for each commodity such as forward buys, toll agreements, relative value, and over varying time periods from 30 days to 3 years, hedging is complicated and the inherent risk of price fluctuation remains with the producers.

This paper presents a systems dynamics model that incorporates various dynamics for modeling the commodity market. Our objectives with the model include:

- Identify and integrate dynamics for understanding the overall commodity market.
- Develop a model for simulating different dynamics in the commodity market for the purpose of optimizing hedging activity.
- Develop a model for maximizing profit margins and minimizing the effect of price fluctuations.
- Address uncertainty and volatility of commodity cost through a risk variance approach.

We will then use the model to simulate hedging strategies. Our objectives with the hedging simulation include:

- Identify the key market dynamics affecting commodity prices and incorporate them into the model.
- Present the best possible commodity candidates for hedging together with price and amount to hedge.

2 Commodity Hedging Dynamics Considered

A producer company takes into account many internal and external impacts in making hedging decisions for required commodities.

Internal impacts include:

- Vertically integrated vs standalone producer
- Hedging strategy
- Overall business structure
- Marketing and promotional expenses (and how management views how such activity will affect demand for their product, and subsequently their expected purchase of the commodity)
- The company's cost structure

External impacts on the industry/sector only include:

- Product positioning
- Industry conditions
- Average cost structure of the industry

External impacts (global) include:

- Geopolitical factors
- Global demand determinants
- Product substitution
- Technological innovation, new product development
- Shifts in consumer preferences
- Regulatory changes

Considering all above and possibly other impacts together, hedging decisions could be very complex. In this paper, we developed a conceptual model considering various dynamics of the commodity market assuming an unregulated, free market. Our objective is to model the overall commodity hedging system for a producer company by incorporating select internal and external dynamics for effective hedging of the needed commodities. We consider the following internal and external dynamics in the model:

- Product demand
- Product inventory
- Product manufacturing capability
- Product marketing strategy
- Product advertising strategy

- Product distribution capability
- Cost of product
- Product pricing strategy
- Commodity hedging strategy
- Cost of commodity
- Overall production and inventory of the commodity
- Demand to commodity
- Other macro dynamics that affect commodity prices

These dynamics are considered after extensive literature reviews on the commodity market [5] [6] [7] [8]. We believe these factors should be considered in hedging decisions and we incorporated them all in our model which we will present in Section 3. There are other dynamics we haven't incorporated into the model to keep it simple. We will mention them together with how they can be incorporated in the decision later in the paper.

3 System Dynamics Model

This paper assumes the reader already has a broad understanding of system dynamics modeling [1] [11] [12]. System dynamics is a useful analysis tool for analyzing and studying the behavior of complex nonlinear dynamic systems by identifying the cause and effect relationships and the feedback control mechanism. In system dynamics, a system is represented by a closed-loop structure which models the relationship and feedback among system factors. A problem or a system is first represented as a stock and flow diagram. Stock shows the quantity of factor under study while flows demonstrate factors which come in and out to change the stock level [1].

The system dynamics (SD) model for the commodity market considering the dynamics mentioned in the previous section is shown in Figure 1 and Figure 2 where Figure 1 showing part 1 and Figure 2 showing part 2 of the same model. This model applies the system dynamics approach to study the overall behavior of the commodity market from the perspective of a manufacturer that relies on several commodities to make its products. The qualitative and conceptual model illustrated in Figure 1 and Figure 2 is built based on the considered dynamics above. The figures are for illustrative purposes only and does not present a comprehensive representation of the model discussed.

In developing this model, we assume producers produce multiple products. Producers use many commodities in producing their products. Most commodities are used in multiple products. For example, a food producer needs many ingredients. These ingredients are used in multiple of their

products. The model is shown above for general products and commodities for simplicity. Also, the model assumes existing and established products for which data for the included

dynamics exist or can be reasonably estimated. It does not address new product launch and associated unknowns.

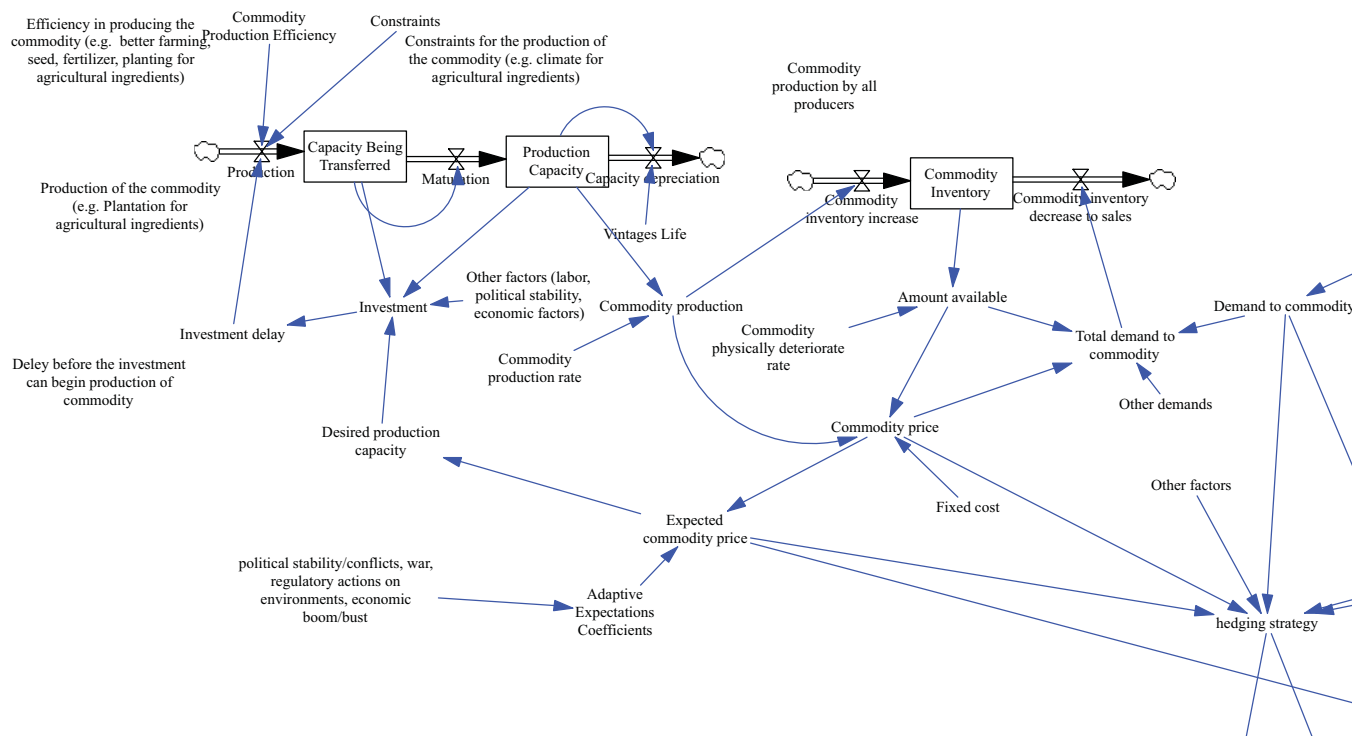


Figure 1 System Dynamics Model for Commodity Hedging – Part 1

The model is modular in nature. By looking at Figure 1, we can distinguish the following module:

Commodity Price Dynamics Module: This module addresses the overall cycle of producing the commodities. This module models the demand and the expected price of the commodity by incorporating the following dynamics from the list in Section 2: Cost of commodity, Overall production and inventory of the commodity, Demand to commodity, Other macro dynamics that affect commodity prices. The stocks in this module are “Capacity Being Transferred”, “Production Capacity” and “Commodity Inventory” and uses variables such as the production capacity of the commodity along with parameters affecting it, the commodity inventory, the commodity production, investment in producing commodity and delay of investment before beginning the production. Another variable is “political stability/conflicts, war, regulatory actions on environments, economic boom/bust” for the decision maker to input his/her estimate about the macro conditions that affect the price of the given commodity.

By looking at Figure 2, we can distinguish the following modules:

Product Advertisement and Marketing Dynamics Module: This module models the impact of marketing and

advertisement efforts on the product demand. It also models the influence of desired product demand on the marketing and advertisement strategies. This module incorporate the following dynamics from the list in Section 2: Product marketing strategy, Product advertising strategy.

Product Demand Dynamics Module: This module addresses estimating the market share and product demand expected by using input from other modules and incorporating the following dynamics from the list in Section 2: Product demand. Since our model is generic, we kept this module very simple. This part of the model can be enhanced for a specific class of products if the producer has more insight into the demand dynamics.

Product Manufacturing and Inventory Dynamics Module: This module models the production and inventory control using stock variables “Product Supply Line”, “Product Inventory” and many supporting variables. This module yields the producer’s demand to commodity. This module incorporates the following dynamics from the list in Section 2: Product inventory, Product manufacturing capability.

Product Distribution Dynamics Module: This module estimates the total required distribution capacity based on production as well as the capacity to rent, if any.

“Distribution Capacity” and “Distribution Capacity to Rent” are stock variables. This module incorporates the following dynamics from the list in Section 2: Product distribution capability, Product inventory.

Product Pricing Strategy Module: This module applies the producer’s pricing strategies taking input from other modules and incorporating the following dynamics from

the list in Section 2: Cost of product, Product pricing strategy.

Commodity Hedging Strategy Module: This module implements the hedging algorithm presented in Section 4. This module takes input from other modules and incorporates the following dynamics from the list in Section 2: Commodity hedging strategy.

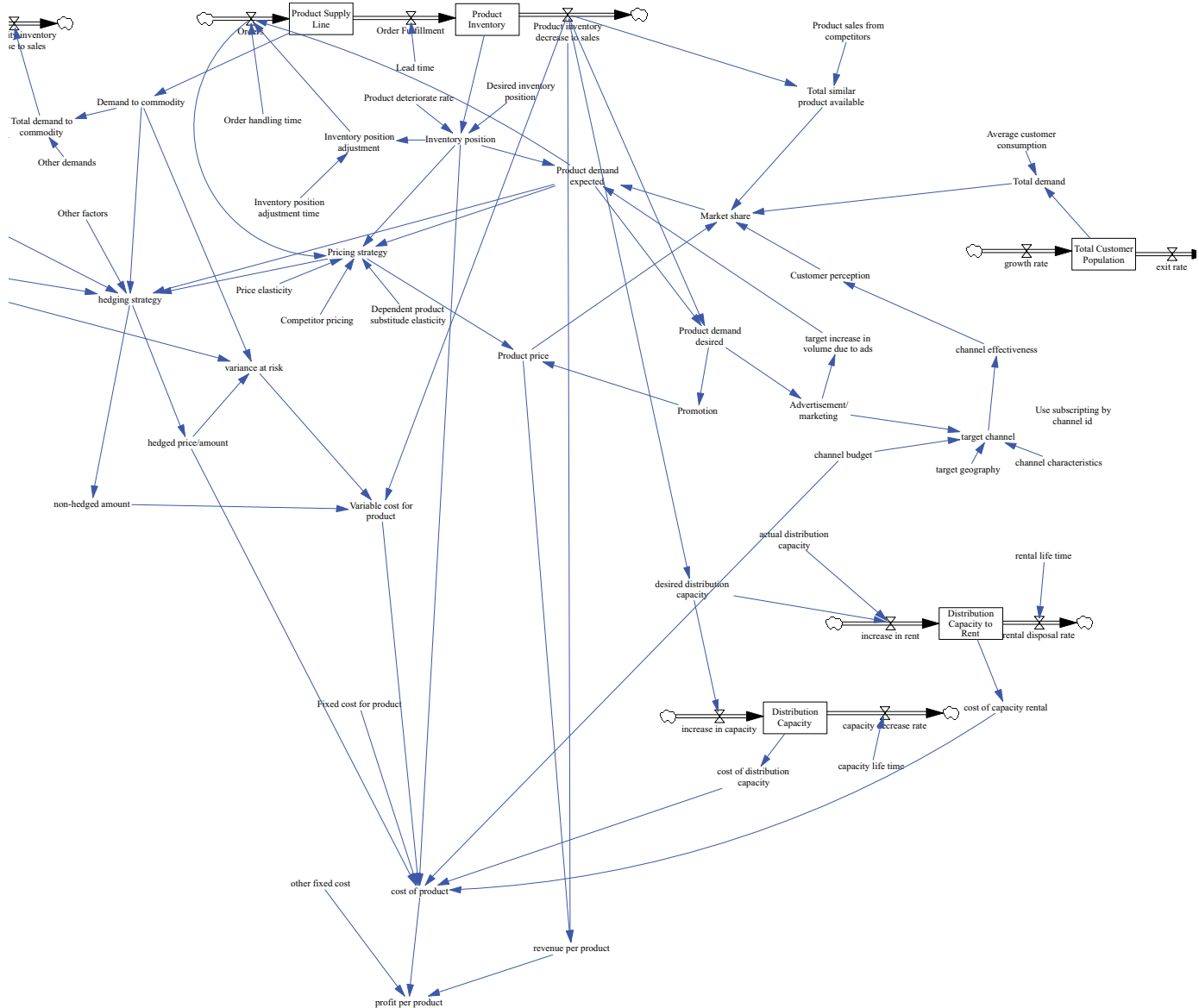


Figure 2 System Dynamics Model for Commodity Hedging - Part 2

In summary, for modeling the production, this model involves not only marketing variables, such as pricing, advertising, and channel development, but also supply chain elements such as production capacity and inventory on hand, as well as distribution capabilities. Interactions between production and

distribution capabilities, inventory management and advertising, are incorporated in the model as they are researched in the literature [2] [3] [4] [9] [10]. All of these factors indirectly affect product pricing and hedging strategy

undertaken by producers. Expected demand for products influences the expected need for commodities.

4 An Algorithm for Hedging

Decisions on what different types of hedging tools to use, the price and the quantity of commodities to be hedged depend on many dynamics that are internal and external to the company as explained in Sections 2 and 3.

Simulating hedging strategies requires a systematic view of the entire supply chain that considers both marketing and supply chain activities and their interactions, in addition to the overall commodity production cycle. Our model in the previous section provides a good platform for developing a simulation model for hedging. In our hedging model, we consider the entire product portfolio for a specific producer. For example, a certain food producer may have several different food products and all these products may use the same commodity (e.g. flour or sugar).

In this section, we will elaborate the commodity hedging strategy module of the model, particularly the “hedging strategy”, “hedged price/amount”, “non-hedged amount” variables. As seen in the model, “hedging strategy” takes “commodity price”, “expected commodity price”, “product demand expected”, “pricing strategy” as inputs and it produces the “hedged price/amount” and “non-hedged amount” values. The “non-hedged amount” variable contributes to “variance at risk” variable. Different hedging approaches can be simulated in the “hedging strategy” variable of the model. We will develop a particular implementation within this variable by executing programming code in Java. Most modern simulation tools allow executing Java code within the model variables [11].

Our hedging method considers all products in the portfolio and all commodities needed to produce the products. The amount to produce each product is obtained as an input to this method via the “product demand expected” variable of the model. It is assumed that how much of which commodity is needed for each product is known (e.g. can be obtained via a table lookup). From this information, a list of <product, amount of commodity> for all products and commodities is generated. This initial list is used in the implementation below.

First, the gross margin is determined for each product by taking the price of the commodities in the futures market. Then, gross margin for all products is calculated by adding the gross margins of each product. In this case, the risk due to hedging is zero because only prices in the futures market are incorporated in this calculation. This gross margin is recorded as a baseline for the next steps.

The next step is to rank the <product, amount of commodity> pairs in order to determine which pairs will be subject to hedging. First, calculate the impact of change in the

commodity price on the baseline product gross margin. For this step, the price of each commodity is assumed to be given by the “expected commodity price” variable. Normally, this variable is determined by (1) weighted running average of the commodity prices of several earlier years, (2) the “commodity price” value returned by the SD model, (3) and a factor for an adaptive expectation coefficient for various exogenous inputs that may influence the expected commodity price. Once the expected price of the commodity is determined, the impact on the baseline gross margin is calculated for each <product, amount of commodity> pair.

Here how it is done: Take each product one at a time. For each commodity needed for this product, assume the expected price of this commodity but assume futures market price for all commodities needed for this product. Calculate the new gross margin for this product. The impact is defined as the difference between the new gross margin of the product and its baseline gross margin. Save this impact value for this <product, amount of commodity> pair. For commodities with “expected commodity price” lower than the futures market price, the impact will be positive.

Next calculation is to determine how much price change is allowed per product so that the gross margin is the same as the baseline gross margin despite the change in commodity price. This value is the allowable change in the product price. We rather want to calculate the percentage of this price change since this percentage change indicates elasticity of the product for our ranking. This percentage value provides more information to the producer regarding the impact of change in the product price. One advantage of using this elasticity is to help the producer with pricing flexibility if company is willing to go aggressive in the pricing and increase the volume.

We rank the <product, amount of commodity> pairs first based on this elasticity value so that minimum impact product is at the top. Within each product, we rank the <product, amount of commodity> pairs based on the impact on the gross margin where the pair with biggest positive impact will be at the top and the pair with biggest negative impact will be at the bottom. At the end, <product, amount of commodity> pairs are all sorted. For each <product, amount of commodity> pair, a variance-at-risk value is calculated taking into account the two values used above in ranking.

The objective of this ranking algorithm is to identify the <product, amount of commodity> pairs with minimum product elasticity thereby lowering the risk of hedging and with biggest positive impact on the product gross margins thereby increasing the overall gross margin.

In the next step, the output of previous step is presented to the decision makers as a sorted list of <product, amount of commodity, cumulative variance-at-risk> where the third value is the cumulative of variance-at-risk values of all previous items in the list. At this step, decision makers can

determine the risk tolerance by choosing a cumulative variance-at-risk value calculated in previous step. This is a significant decision: the decision maker makes a cut on the level of risk tolerance for the producer in producing all products. Any <product, amount of commodity> pair above the cut will be subject to hedging. That means, the decision maker decides on which <product, amount of commodity> pairs will be considered for hedging based on his/her opinion on the correct risk tolerance for the company at that moment. Based on the selection of the decision maker, all <product, amount of commodity> pairs above the chosen level will be considered in the next step for executing the hedging decision.

The next step in the algorithm is to present the business decision maker the output of “hedged price/amount” and “non-hedged amount” values. Non-hedged amounts are easily calculated for each commodity based on decision maker’s cut in the previous step. The amount to hedge for each commodity is also easily added up. For specific hedge prices and amounts, the algorithm needs to match the hedge amount with available hedge offers from suppliers in commodity markets. We assume available hedge offers are stored in a database for easy matching.

At the last step, for each commodity to be hedged, the algorithm tries matching the available offer starting from the lowest priced offer provided the offer is good for the duration of product horizon and is less than the minimum of futures market price and “expected commodity price” value. The algorithm tries matching for as many commodities possible. For remaining amounts of commodities that are not matched due to lack of available offers in the market, futures market prices are assumed. At the end of this matching, all <product, amount of commodity> pairs are revisited and each pair is assigned a hedging price for the amount of commodity. If the amount of commodity cannot be handled by a single price but multiple prices at the end of the algorithm, <product, amount of commodity> pair is sliced into multiple <product, amount of commodity> pairs for each price.

This concludes the algorithm for the “hedging strategy” variable.

The SD model further outputs the profit margin for the product, the product volume and the overall gross margin. The SD model can be run many times for the purpose of sensitivity analysis. The output helps hedging decision makers and the risk analysis team.

5 Further Dynamics to Consider

The model provides a framework for adding new dynamics, simulating different scenarios, and conducting sensitivity analysis. Several variables, which were not included in the model, can easily be incorporated:

- Trade-weighted index for the U.S. Dollar is a significant factor affecting the commodity prices because most commodities are priced in U.S. dollar. Appreciations or depreciations of dollar affects the commodity prices. This can be added into the model. The model can flexibly incorporate other currencies (not US) as well.
- The world price of crude oil has a significant effect on the price of other commodities. Although the oil price itself is affected by many dynamics including the trade-weighted index for the U.S. Dollar, the oil price can be easily incorporated as an exogenous variable into the model.

The model can be enhanced by introducing other modules such as the following:

- Capacity acquisition module: This module can model the dynamics of capacity acquisition through renting and/or outsourcing. This is needed if the available capacity is not enough for producing the product. Similarly, a module for capital investment to increase the capacity can be introduced.
- Competition pricing module: The SD model presented above does not consider competition pricing. A new module can be developed to model the competition pricing dynamics in various engagements (e.g. Cournot, Bertrand) and leader/follower scenarios.
- The opportunity cost module: This new module can be developed to simulate the opportunity cost by sensitivity analysis of various parameters on risk, pricing, manufacturing and advertising modules.
- Product promotion strategy: This new module can be developed to measure the impact of product promotions on the product demand. It should also model the influence of desired product demand on the promotion strategies.
- Product substitution module: Some products of the same producer may be substitutable with one another. This new module can be developed to simulate the effect of product pricing and manufacturing on the substitute products.

These are left to investigate for other papers.

6 Conclusions

The objective of this study was the development of a generic model for abstracting the dynamics in commodity hedging. System dynamics modeling has been proven to be an effective tool for analyzing complex nonlinear systems that inherently have feedback loops. In this paper, we presented a generic system dynamics model that integrates various commodity and product dynamics. This generic

model is a good abstraction for variety of for variety of products and a variety of commodities including agricultural, metals and mineral, chemicals, and The generic model has been shown to be a suitable platform for implementing an algorithm to abstract a hedging strategy.

The hedging algorithm implemented a risk-return simulation model and provided useful output for decision makers. The model can be customized by implementing different hedging strategies. It is also extensible to flexibly add new modules for incorporating further dynamics. To our knowledge, our model is the most comprehensive one taking into account a great deal of commodity and production dynamics, and meanwhile flexible to allow running Java code for implementing hedging algorithm. Our paper presents a model allowing to apply various dynamics and to experiment with different hedging algorithms.

The model allows simulation of a variety of scenarios:

- Different hedging algorithms
- Different manufacturing capacities
- Different level of success of product advertising
- In general, by modifying different dynamics in the system

One challenge corporations face is to coordinate different functional areas such as manufacturing, advertising, marketing, distribution, procurement, hedging, pricing. Our model could facilitate analysis encompassing all these functions, and once the decision makers agree on what the right action for hedging and production would be, then it can help with coordinating the efforts by different functional areas of the organization.

One difficulty using this model is that it contains some exogenous variables representing the extent of external dynamics onto different variables, for example "political stability/conflicts, war, regulatory actions on environments, economic boom/bust" variable. Choosing the right value for such variables is dependent on the decision maker's intuition and past experience. A lot of times, sensitivity analysis is performed over a set of possible values.

7 Future Work

There is plenty of future work to further enhance the presented model. One area of work is to incorporate further dynamics listed in the previous section into the model. Another area is to try the model for different types of commodities and products.

Another area that we are actively working is to combine the neural network algorithms with system dynamics modeling in order to employ machine learning in policy development.

Although this effort could be independent of the presented model in this paper, we would like to apply the approach first into this model for intelligent and adaptive commodity hedging.

8 References

- [1]. J. D. Sterman. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. Boston: Irwin/McGraw-Hill, 2000.
- [2]. F. A. Osorio, S. A. Aramburo. A System Dynamics Model for the World Coffee Market. *Proceedings of the 27th International Conference of the System Dynamics Society* <http://www.systemdynamics.org/conferences/2009/proceed/papers/P1312.pdf>
- [3]. W. Tharmmaphornphilas, H. Lohasiriwat, P. Vannasetta. Gold Price Modeling Using System Dynamics, *Engineering Journal*, Vol 16, No 5 (2012), <http://engj.org/index.php/ej/article/view/293>
- [4]. A. S. Cui, M. Zhao, T. Ravichandran. Market Uncertainty and Dynamic New Product Launch Strategies: A System Dynamics Model, *IEEE Transactions on Engineering Management* (Volume:58 , Issue: 3)
- [5]. A. Deaton, G. Laroque. On the behavior of commodity prices. *Review of Economic Studies*, 59: 1 – 24, 1992
- [6]. A. Deaton, G. Laroque. Competitive storage and commodity price dynamics. *Journal of Political Economy*, 104: 896–923, 1996
- [7]. UNCTAD secretariat to the G20 Commodity Markets Working Group. "Excessive commodity price volatility: Macroeconomic effects on growth and policy options". (2012).
- [8]. R. S. Pindyck. Volatility and commodity price dynamics, *The Journal of Futures Markets* 24(11) 1029–1047, 2004
- [9]. B. J. Angerhofer, M. C. Angelides, *System Dynamics Modelling In Supply Chain Management: Research Review, Proceedings of the 2000 Winter Simulation Conference*
- [10]. P. Georgiadis, D. Vlachos, E. Lakovou. A system dynamics modeling framework for the strategic supply chain management of food chains, *Journal of Food Engineering* 70 (2005) 351–364
- [11]. AnyLogic Multimethod Simulation Software, <http://www.anylogic.com/>
- [12]. Vensim PLE by Ventana Systems, Inc., <http://vensim.com/>

Multi-domain modeling and Simulation of Quad-rotor aircraft based on Modelica

Xiaohui Ma, Zhihua Li, Chao Nie

School of Mechanical Engineering, Hangzhou Dianzi University, Hangzhou 310018, Zhejiang, China

Abstract - Quad-rotor aircraft is a complex system that involves multiple disciplines, which is provided with multivariate, strong coupling and nonlinear characteristics. In order to set up a high-precision model, a multi-domain unified modeling based on Modelica is raised. The paper first analyzes the propellers of quad-rotor aircraft, the coupling relationship between the driving force of aircraft and structural parameter of propellers is clear. Then it makes an analysis of propellers, the coupling relationship between the speed as well as attitude angle of aircraft and driving force can be obtained. Finally, the multi-domain unified nonlinear mathematical model of quad-rotor aircraft is established on the Mworks platform. By conducting PID decoupling control simulation on the flying attitude of aircraft, the consequence is that the nonlinear model can satisfy the requirements of controlling quad-rotor aircraft. This study shows that it is more accurate to create quad-rotor aircraft in the way of multi-domain unified modeling. This simulation lays the foundation for subsequent optimization design and product development of quad-rotor aircraft.

Keywords: quad-rotor aircraft; Modelica; multi-domain modeling

1 Introduction

The four rotor aircraft is an aircraft with 4 cross symmetrical propellers. Compared with the conventional helicopter, the quad rotor aircraft has a relatively simple mechanical structure. It can be achieved various flight maneuvers by changing the speed of the propellers [1]. It can be operated on wide area regardless of the effect of ground configuration. The merit of aircraft is maximized for the practical use in the places that dangerous or difficult to approach. Further, four rotor aircraft is much cheaper and safer in dangerous tasks than piloted aircraft.

Four rotor aircraft is a complex system with multi variable, strong coupling and nonlinear characteristics. In recent years, domestic and foreign researchers mainly focus on the control of the hovering state of the aircraft. One study is directly obtaining the physical model by some necessary assumptions and approximation according to the physical structure of four rotor aircraft and dynamic balance equations. The other is based on data driven. The enough intermediate data of a particular quad rotor can be acquired through experiment. Then using the method of nonlinear time series modeling, hovering state of the aircraft can be homeostatic controlled by getting corresponding nonlinear identification model [4][5]. Ramirez. A [6] put forward a

vision based on control method to test the four rotor aircraft. Bosnak. M [7] raised an implementation of computer vision to hold a quadcopter aircraft in a stable hovering position using a low-cost, consumer-grade and video system. Alexis. K [8] used model predictive to control the autonomous flight of the quad-rotor aircraft. The current researches focus on the steady state control of the hovering aircraft instead of the steady flight performance of the aircraft.

Modern aircraft design involves the multi-domain; different parameters have different effects on the performance of the aircraft. Traditional simulation methods and softwares focus more on a single field. For the construction of multi-domain models, the usual method is to model each domain separately by different softwares and then integrate them, which resulting in poor model system coupling degree and low simulation accuracy and efficiency. This cannot be qualified for the unified design and analysis of complex electromechanical systems. However, aircraft is a complex electromechanical system with multi variable, strong coupling and nonlinear characteristics. Building aircraft model and doing simulation analysis, performance evaluation by using multi-domain unified modeling method on an open, object-oriented language—Modelica [9] and at Mworks [10] platform, the problems brought by softwares integrating can be solved, which is an innovative method. In this paper, quad-rotor aircraft which is based on PID control is set up based on Modelica language. Performance evaluation indexes are pitch angle, yaw angle, roll angle and air speed.

2 Dynamic models

Since the quad-rotor aircraft includes highly nonlinear factors, we need to consider several assumptions in order to get a desired model:

1. The body is rigid and symmetrical.
2. The center of mass and body fixed frame origin coincide.
3. Quad-rotor aircraft by the drag and gravity is not affected by the altitude of the flight and other factors, the total remains unchanged
4. The force of each direction of the aircraft is proportional to the square of the speed of the propeller.

In order to get the mathematical model of flight vehicle, two coordinate systems are established: the inertial coordinate system and the body coordinate system. As shown in Figure 1 below.

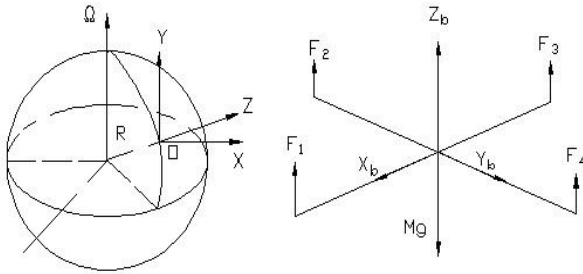


Figure 1 Inertial coordinate system and body coordinate system

External force of the aircraft: gravity G , which is in the oz negative direction; lift of the quad-rotor $F_i (i=1,2,3,4)$, which are in the oz positive direction; moment $M_i (i=1,2,3,4)$, which are perpendicular to limb plane and contrary to the direction of the rotation.

2.1 Propeller model

In this paper, the force F and drag Q of the propeller are only considered, and the air drag and lateral moment are ignored.

$$F = K_f \omega^2, \quad Q = K_d \omega^2$$

$$K_f = \frac{\rho S R^2 C_L}{2}, \quad K_d = \frac{\rho S R^2 C_D}{2}$$

Where C_L is the coefficient of lift, C_D is the coefficient of drag, their relationship with the angle of attack is shown in Figure 2; ρ is the air density; R is the propeller radius; S is the rotor disk area, and $S = \pi R^2$.

Above all, the propeller Modelica model is obtained as

$$P = \begin{bmatrix} \cos \psi \cos \phi & \cos \psi \sin \theta \sin \phi & \cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi \\ \sin \psi \cos \theta & \sin \psi \sin \theta \sin \phi & \sin \psi \sin \theta \cos \phi - \sin \phi \cos \psi \\ -\sin \theta & \cos \theta \sin \phi & \cos \theta \cos \phi \end{bmatrix} \quad (1)$$

Where ϕ is the roll angle; θ is the pitch angle; ψ is the yaw angle.

According to Newton's second law, we have:

$$\vec{F} = m\vec{a} = m \frac{d\vec{v}}{dt} = m \frac{d^2}{dt^2} \vec{r}$$

$$\vec{F} = \left(\sum_{i=1}^4 F_i \right) \vec{e}_3 - m g \vec{k} = m \frac{d^2}{dt^2} \vec{r} = m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} \quad (2)$$

Where F is the sum of lift forces; m is the quality of aircraft; v is the air speed. According to transformation matrix P , we have:

shown in Figure 3.

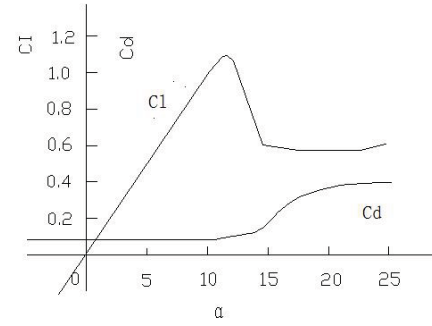


Figure 2 C_L and C_D curves

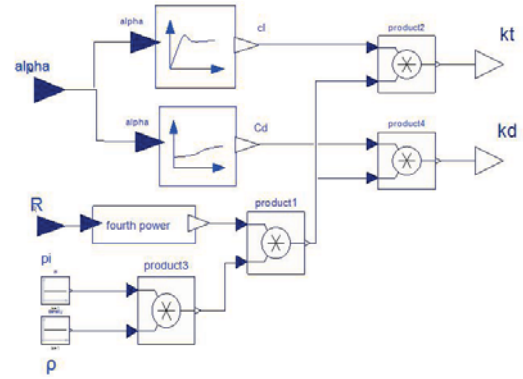


Figure 3 Modelica model of propeller

2.2 Mathematical model of quad-rotor aircraft

As an important basis for the derivation of the dynamic characteristic model of the quad-rotor aircraft, the transformation matrix of the body coordinate system to the geographical coordinate system can be shown as Equation 1:

$$\begin{cases} \ddot{x} = \sum_{i=1}^4 K_f w_i^2 (\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi) / m \\ \ddot{y} = \sum_{i=1}^4 K_f w_i^2 (\sin \psi \sin \theta \cos \phi - \sin \phi \cos \psi) / m \\ \ddot{z} = \sum_{i=1}^4 K_f w_i^2 (\cos \theta \cos \phi) / m - g \end{cases} \quad (3)$$

According to theorem of angular momentum

$$\vec{M} = \frac{d\vec{H}}{dt}, \text{ we have:}$$

$$\vec{M} = \vec{M}_1 + \vec{M}_2 = (\vec{h}_1, \vec{h}_2, \vec{h}_3) \begin{bmatrix} l(F_4 - F_2) \\ l(F_3 - F_1) \\ K_d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{bmatrix} \quad (4)$$

Where l is the distance from rotor center to the origin of the coordinate system.

The body is rigid and symmetrical, and the moment of inertia is a diagonal matrix:

$$J = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix}$$

Where J_x, J_y, J_z is the inertia moment of each coordinate axis.

Angular momentum moment of the aircraft is:

$$\vec{H} = (\vec{b}_1, \vec{b}_2, \vec{b}_3) \begin{bmatrix} J_x \omega_x \\ J_y \omega_y \\ J_z \omega_z \end{bmatrix}$$

From equation (4) and (5), we have:

$$\vec{M} = \frac{d\vec{H}}{dt} \Big|_b + \vec{\omega} \times \vec{H} = (b_1, b_2, b_3) \begin{bmatrix} J_x \dot{\omega}_x + (J_z - J_y) \omega_y \omega_z \\ J_y \dot{\omega}_y + (J_x - J_z) \omega_x \omega_z \\ J_z \dot{\omega}_z + (J_y - J_x) \omega_x \omega_y \end{bmatrix} \quad (5)$$

$$\begin{cases} \dot{\omega}_x = [l(F_4 - F_2) + (J_z - J_y) \omega_y \omega_z] / J_x \\ \dot{\omega}_y = [l(F_3 - F_1) + (J_z - J_x) \omega_x \omega_z] / J_y \\ \dot{\omega}_z = [K_d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) + (J_x - J_y) \omega_x \omega_y] / J_z \end{cases} \quad (6)$$

By defining U_1, U_2, U_3, U_4 as the control inputs for the quad control channels of the quad-rotor aircraft, the system could be controlled directly.

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} K_t \sum_{i=1}^4 \omega_i^2 \\ K_t(\omega_4^2 - \omega_2^2) \\ K_t(\omega_3^2 - \omega_1^2) \\ K_d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{bmatrix} \quad (7)$$

Where U_1 is the input of vertical direction; U_2 is the input of roll; U_3 is the input of pitch; U_4 is the input of the yaw; ω_i is the speed of each propeller.

From what we have discussed above and equation (3), (6) and (7), we have dynamic model of the aircraft as shown in equation (8):

$$\begin{cases} \ddot{x} = (\cos\psi \sin\theta \cos\phi + \sin\psi \sin\phi) U_1 / m \\ \ddot{y} = (\sin\psi \sin\theta \cos\phi - \cos\psi \sin\phi) U_1 / m \\ \ddot{z} = (\cos\theta \cos\phi) U_1 / m - g \\ \ddot{\phi} = \dot{\omega}_y = [l U_2 + \dot{\theta} \psi (J_y - J_z)] / J_x \\ \ddot{\theta} = \dot{\omega}_x = [l U_3 + \dot{\phi} \psi (J_z - J_x)] / J_y \\ \ddot{\psi} = \dot{\omega}_z = [U_4 + \dot{\phi} \dot{\theta} (J_x - J_y)] / J_z \end{cases} \quad (8)$$

And the Modelica model of the aircraft is shown as Figure 4.

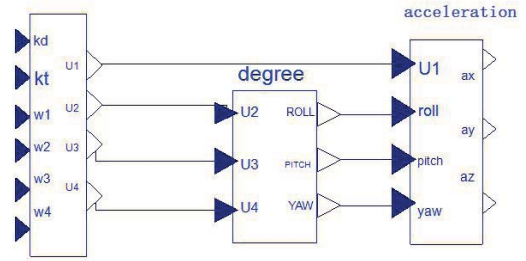


Figure 4 Dynamic Modelica model

2.3 PID control model

Quad-rotor aircraft is a complex system with multi variable, strong coupling and nonlinear characteristics. It is difficult to control directly. Therefore, the model is decoupled, and divided into four channels, the speed, the roll angle, the pitch angle and the yaw angle in order to reduce the control difficulty.

Compared with other control methods, the traditional PID control has the advantages of simple structure, easy implementation, high reliability, good stability and so on. By adjusting the proportion, integral and differential coefficients of the PID controller, the system can be quickly put into stable. PID control equation is as follows:

$$u = K_p [e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt}] + u_0$$

The optimal parameters of the controller have been determined by trial and error as shown in table 1.

Table 1 Parameters of PID controller

Attitude	P	I	D
Speed	3	0.01	1
Roll	1.5	0.01	0.1
Pitch	2	0.1	0.1
Yaw	0.9	0.03	0.1

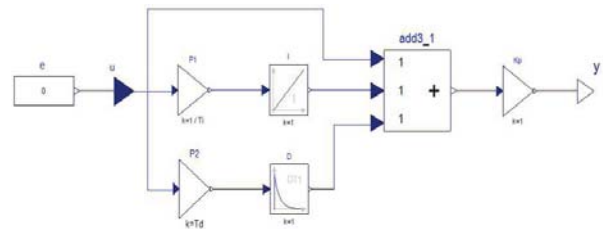


Figure 5 PID controller Modelica model

2.4 Integrated model

Using Modelica language to construct models at MWorks platform, each sub-model is independent of each other on describing the physical characteristics and mathematical relations. In order to ensure the reuse of submodels, each submodel has its own input and output connectors to input parameters and communicate with other submodels. Meanwhile, a submodel can be composed of other submodels and components by packaging, which is the multi-level modeling method. Multi-level modeling has

the advantages of clearly organized, reusable, high modeling efficiency etc. By modeling, packaging and

connecting all the submodels of the aircraft at MWorks, the integrated model is obtained and shown in Figure 6.

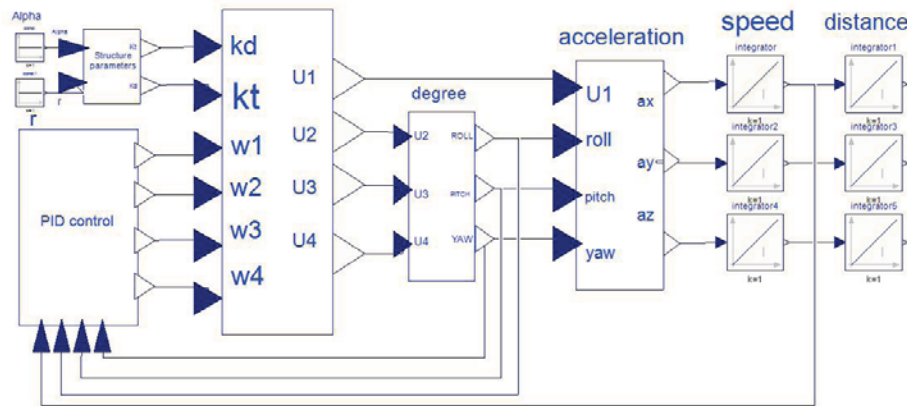


Figure 6 Integrated Modelica model of quad-rotor aircraft

3 Simulation results and analysis

The parameters of this simulation are derived from the quad-rotor aircraft designed by our group. As shown in table 2.

Table 2 Parameters of Quad-rotor aircraft

parameter	value
M/kg	1.25
$g/(m \cdot s^{-2})$	9.8
l/m	0.25
$J_x/(kg \cdot m^2)$	0.033
$J_y/(kg \cdot m^2)$	0.033
$J_z/(kg \cdot m^2)$	0.061
$\alpha / (^{\circ})$	13
R/m	0.15

The speed was set at 5m/s, the simulation results of the integrated model of quad-rotor aircraft at MWorks platform are shown in Figure 7- Figure 10, and. Figure 7 is the comparison curves of speed with PID controller and speed without controller. Figure 8-10 are the comparison curves of attitude angles with PID controller and attitude angles without controller. In figure 7, the speed with PID controller reaches steady in 9 seconds, while the other reaches steady in almost 20 seconds. In figure 8 and 9, the roll angle and yaw angle with PID controller are fluctuating within a very small range. In figure 10, under the effect of the PID controller, the rapidity and stability of the system are promoted, the pitch angle reaches steady in 10 seconds. The simulation results show that the aircraft system can reach steady quickly by the PID controller. The availability of the method was proved by the simulation.

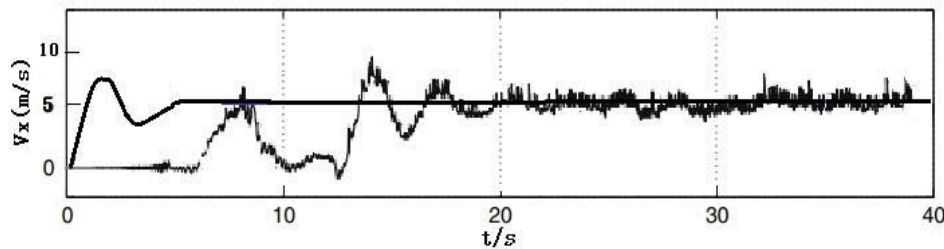


Figure 7 Speed curves.

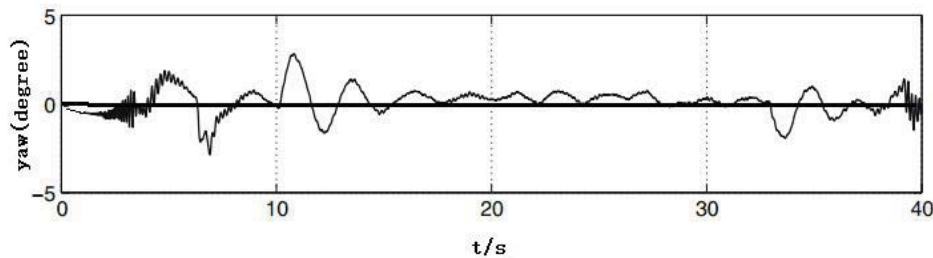


Figure 8 Yaw angle curves

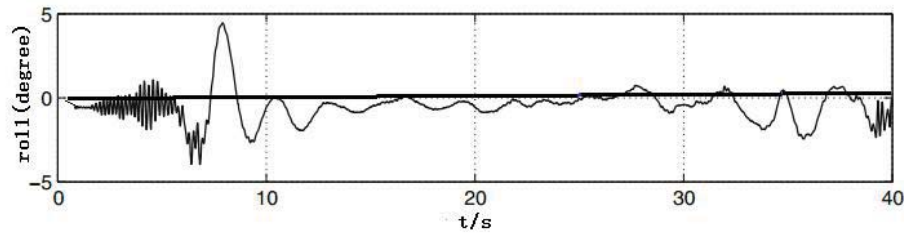


Figure 9 Roll angle curves

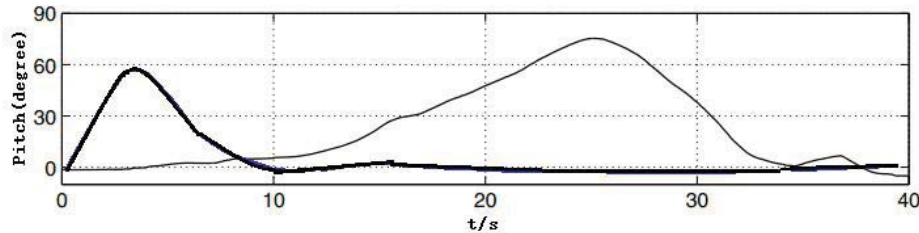


Figure 10 Pitch angle curves

4 Conclusions

Quad-rotor aircraft is a complex system which is provided with multivariate, strong coupling and nonlinear characteristics. A multi-domain unified modeling with PID controller based on Modelica is raised. Through the multi-domain unified modeling method, the accuracy of the system is improved. Simulation results show that PID control is an effective control method for the quad-rotor aircraft. The control strategy of this paper can provide a certain basis for the research of the control of the four rotor aircraft. As the model is an ideal model, and some unstable factors are not considered, the simulation results are relatively different from the real experiments. We need to continue the study on this basis with a more concise and effective control of the simulation to achieve a more stable state in the future.

Acknowledgements

This study was supported by National Natural Science Foundation of China (Grant No.51275141).

References

- [1] Wu J, Peng H, Chen Q. RBF-ARX model-based modeling and control of quad-rotor[C]. IEEE International Conference on Control Applications. Yokohama, Japan, 2010:1731-1736.
- [2] Cetinsoy E, et al. Design and construction of a novel quad tilt-wing UAV [J]. Mechatronics. 2012, 22(6):723-745
- [3] Mostafa Mohammadi, Alireza Mohammad Shahri. Adaptive nonlinear stabilization control for a quad-rotor UAV: Theory, Simulation and Experimentation [J]. Journal of Intelligent & Robotic Systems. 2013, 72(1):105-122
- [4] Gonzalez I, Salazar S, Lozano R, Escareno J. Real-time altitude robust controller for a quad-rotor aircraft using sliding-mode control technique[C]. International Conference on Unmanned Aircraft Systems. Mexico City, Mexico. 2013:650-659
- [5] Senkul F, Altug E. Adaptive control of a tilt - roll rotor quad-rotor UAV [C]. International Conference on Unmanned Aircraft Systems. Istanbul, Turkey 2014:1132 - 1137
- [6] Ramírez, A, et al. Stability Analysis of a Vision-Based UAV Controller [J]. Journal of Intelligent & Robotic Systems, 2014, 74(1-2):69-84
- [7] Bosnak, M, et al. Quadrocopter hovering using position-estimation information from inertial sensors and a high-delay video system [J]. Journal of Intelligent & Robotic Systems. 2012, 67(1): 43-60
- [8] Alexis K, et al. Model predictive control scheme for the autonomous flight of an unmanned quad-rotor[C]. IEEE International Symposium on Industrial Electronics. Gdansk, Poland, 2011:2243 - 2248
- [9] Wu. Y.Z., Wu, M.F., Chen, L.P. Study on the Hybrid Modeling Platform Based on Modelica Language for Complex Machinery System. China Mechanical Engineering, 2006, 17(22):2391-2396.
- [10]. Suzhou Tongyuan Information Technology Limited Company. MWorks Toolkit Model Optimization. <http://www.tongyuan.cc/>

Multi-domain unified modeling and simulation of semi-active suspension in magneto-rheological damper

Huiyi Zeng, Zhihua Li, Chao Nie

School of Mechanical Engineering, Hangzhou Dianzi University, Hangzhou, 310018, Zhejiang, China

Abstract : *In order to build high-precision and high-efficient MR damper system model, a multi-domain unified modeling method was proposed. Firstly, based on the analysis of a MR damper semi-active suspension system model, various fields of MR, such as structure, magnetic field, control and mechanics, were studied to find the main influencing parameters. And then for the multi-domain coupling characteristics of MR damper, a multi-domain unified modeling of MR damper system was established based on Modelica/MWorks platform. By adjusting the main influencing parameters, the relationship diagrams between the parameters and the performances were obtained. Finally, some performance results were compared with the traditional method used Matlab/simulink modeling and the method proposed. The results show that the model based on Modelica can preferably reflect the complex relationship between the MR damper coupling among the parameters within the system, and can get a better simulation result.*

Keywords: MR damper; multi-domain; simulation

1 Introduction

With the development of the performance of the car, there are higher requirements on the ride comfort and handling stability than before. One of the main functions of the vehicle suspension system is to provide support, effectively isolate vibration and shock caused by the road surface, which determines the ride and handling stability during in the process of driving. For traditional passive suspension systems can not meet the different road conditions and the driving-state, the semi-active suspension system consists of passive springs and adjustable damping force of the active shock absorber, which is able to adapt to different driving-state, and its price is low, the manufacturing process is relatively simple, damping effect is good, which is becoming the development direction of modern automotive suspension systems[1-3].

MR damper are compact structure, low power consumption, high damping force, wide dynamic range, and its damping force can be controlled by adjusting the size of the magnetic field intensity, which was designed based on MR effect. By controlling the external magnetic field intensity, rheological properties of the liquid of MR fluid can be varied from a liquid to semi-solid within milliseconds time to achieve active control of the damper

characteristics. Currently, scholars have conducted various studies in MR damper, which can be roughly divided into three categories. The first category is to study MR damper control strategies, such as Yang[4] proposed a input saturation characteristics sliding mode control based on the nonlinear characteristics and adjustable damping force output saturation characteristics of absorber solenoid valve; Sulaiman[5] proposed TFC control strategy based on semi-active vehicle suspension MR damper, and compared with GRD control method and showed that the proposed TFC control method can significantly reduce the size of the force on the tire. The second category is MR damper structure design, such as Pang[6] and Imaduddin[7] respectively designed vehicle MR damper structure, and verified that the design were feasible and reasonable. The third is to establish accurate mathematical model of MR damper for analysis, such as Kasprzyk[8] controlled a model by using two control methods for Skyhook and FxLMS (Filtered-x LMS). When the mathematical model is not accurate, using two control strategies can not meet the requirements; Ma[9] have conducted studies in fluid dynamics analysis of the MR fluid in damping passage based on the theory of fluid dynamics and MR fluid rheological properties, and have derived MR damper force model in detail, which can describe the basic mechanical properties of MR damper and provide theoretical guidance for the research on semi-active suspension control.

In summary, although there were a number of studies on MR damper, it mainly was to a single field of modeling and simulation. However, MR damper system involves multiple disciplines, such as mechanical dynamics, structure and control and other science. Single-domain simulation of complex electromechanical system is difficult to perform the whole simulation, so it is necessary to use methods and tools which can be unified modeling and simulation among different disciplines. This paper will adopt a multi-domain unified modeling method based on Modelica[10] language. The language is a multi-domain unified modeling language based equation, in which all models are established through a language, so that the coupling can be achieved seamlessly between the various subsystems. By the MWorks[11] platform, MR damper multi-domain system unified modeling was built, simulation and further analysis, performance evaluation, overcome the problems brought from the approach using integrated tools.

$$B = \frac{nI}{SR_m} = \frac{nI}{\pi l \left(\frac{d_3 + d_4}{2} \right) (R_{mA1} + 2R_{mA2} + 2R_{mA3} + R_{mA4})} \quad (9)$$

According to the B-H curve of MRF-132DG type MR fluid shown in Figure 4 (LORD Corporation, 2009, $a = 0.24$, $b = 1$, $\eta = 0.09 Pa \cdot s$), the value of the magnetic field strength H on the working surface can be obtained.

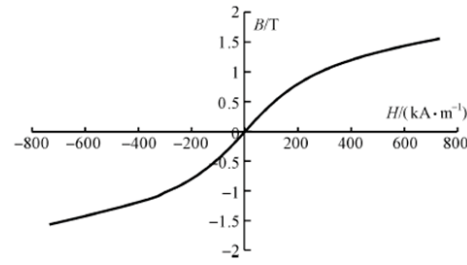


Fig.4 The B-H curve of MRF-132DG type MR fluid

According to the above analysis, magnetic field Modelica model of MR damper has been shown in Figure 5 based on MWorks platform.

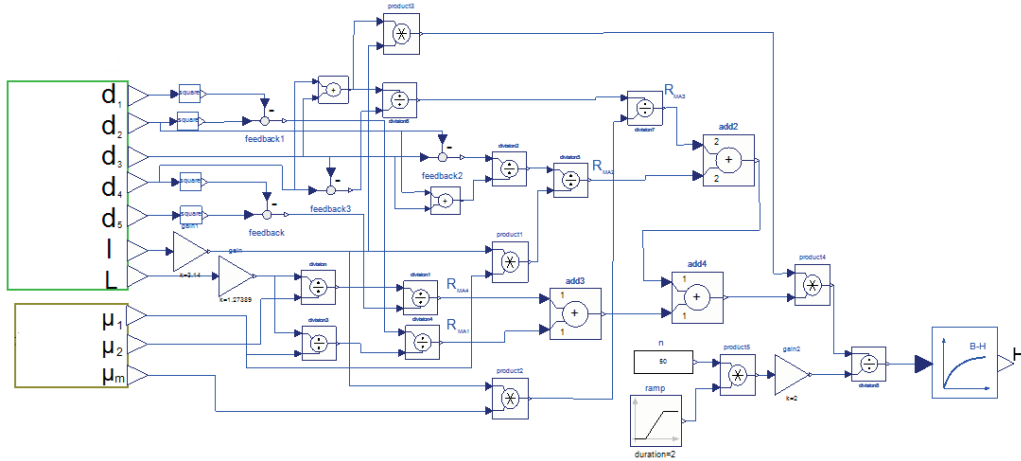


Fig.5 Magnetic field Modelica model of MR damper

3.2 Damping force model

Damping force model of MR damper is a multi-domain coupling model involved with structure parameters, mechanical parameters, MR fluid parameters and electromagnetic parameters. This model reflects the relationship between the parameters and the damping force.

In the case of an applied magnetic field, the MR fluid shows Bingham plastic fluid characteristics. MR damper adopted mixed mode using a combination of flow mode and shear mode in this study, as shown in Figure 1.

Damping force Mathematical model of MR damper can be obtained according Bingham plastic fluid equation:

$$\begin{cases} \Delta p = \frac{24\eta A_p}{\pi d_3 h^3} v + \frac{3l\tau_y}{h} \text{sgn}(v) \\ p_1 = p_2 \pm \Delta p \\ p_3 = p_2 = \frac{p_0 V_0}{V_0 + sA_g} \\ F_l = (p_1 - p_2)A_h - (p_1 - p_a)A_g \\ F_y = (p_2 - p_1)A_h + (p_1 - p_a)A_g \end{cases} \quad (10)$$

Where h is the gap of the damper channel, $h = (d_4 - d_3)/2$; Δp is damping pressure difference across the gap; η is MR fluid zero magnetic field viscosity; A_p is effective cross-area of piston, $A_p = \pi(d_3^2 - d^2)/4$; v is piston speed (when stretching is positive); τ_y is shear yield stress of MR fluid in the magnetic field, $\tau_y = aH^b$; sgn is sign function; p_1 is the pressure in the upper chamber; p_2 is the pressure in the lower chamber; p_3 is the working pressure in gas chamber; p_0 is the air pressure in gas chamber; p_a is the standard atmospheric pressure; V_0 is gas chamber inflatable volume; s is the displacement of piston vibration; A_g is effective cross-area of piston rod, $A_g = \pi d^2/4$; F_l is damping force of the shock absorber during extension stroke; F_y is damping force of the shock absorber during compression stroke.

When the piston is stretched, the upper chamber pressure is greater than the lower chamber pressure, so:

$$p_1 - p_2 = \Delta p \quad (11)$$

Therefore, damping force F_1 of the shock absorber during the extension stroke can be obtained by:

$$\begin{aligned} F_1 &= (p_1 - p_2)(A_p + A_g) - (p_1 - p_a)A_g \\ &= (p_1 - p_2)A_p + (p_a - p_2)A_g \\ &= \Delta p A_p + (p_a - p_2)A_g \\ &= \frac{192\eta l A_p^2}{\pi d_3(d_4 - d_3)^3} v + \frac{12l A_p \tau_y}{(d_4 - d_3)} \text{sgn}(v) + \left(p_a - \frac{p_0 V_0}{V_0 + s A_g} \right) A_g \end{aligned} \quad (12)$$

Similarly, the damping force F_y of the shock absorber during the compression stroke can be obtained by:

$$F_y = \frac{192\eta l A_p^2}{\pi d_3(d_4 - d_3)^3} v + \frac{12l A_p \tau_y}{(d_4 - d_3)} \text{sgn}(v) - \left(p_a - \frac{p_0 V_0}{V_0 + s A_g} \right) A_g \quad (13)$$

By the equation (12), (13), the damping force of MR damper can be obtained by:

$$F = \frac{192\eta l A_p^2}{\pi d_3(d_4 - d_3)^3} v + \frac{12l A_p \tau_y}{(d_4 - d_3)} \text{sgn}(v) + \left(p_a - \frac{p_0 V_0}{V_0 + s A_g} \right) A_g \text{sgn}(v) \quad (14)$$

Through analysis, the damping force of MR damper as shown in equation (14) consists of three parts: the first item is related with the dynamic viscosity of MR fluid, namely the viscous damping force; the second item is related with the yield stress of MR fluid, namely the Coulomb damping force; the third item is related with inflation pressure and volume, namely the compensation force. Thus, the damping force of MR damper can be written as:

$$F = F_\eta + F_{MR} \text{sgn}(v) + F_p \text{sgn}(v) \quad (15)$$

Where F_η is the viscous damping force, $F_\eta = C_e \cdot v$; F_{MR} is the Coulomb damping force; F_p is the compensation force.

Through the above analysis, damping force Modelica model of MR damper has been shown in Figure 6.

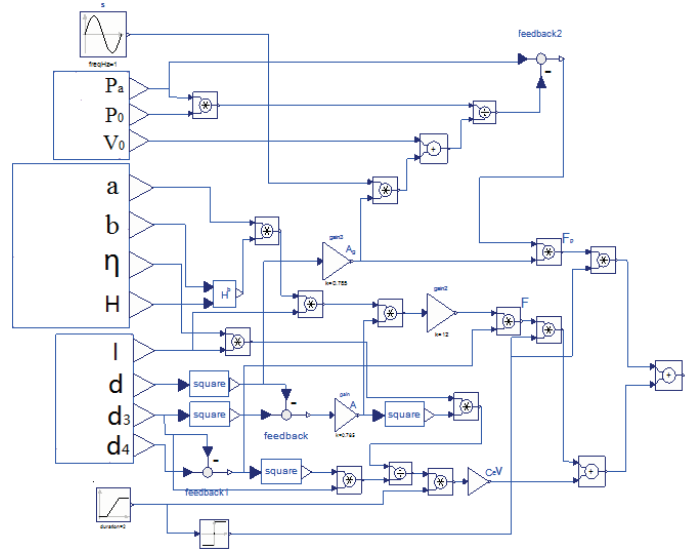


Fig.6 Damping force Modelica model of MR damper

3.3 Dynamical model

Figure 7 is a simplified two DOF of two-mass of quarter-cart vehicle dynamics model, which has a simple structure and it can be able to accurately reflect essential characteristics of the vehicles, such as the sprung mass acceleration, suspension dynamic deflection and tire dynamic load. x_0 is the road surface excitation; x_1 is the vertical displacement of the wheel; x_2 is the vertical displacement of the body; m_1 is the non-sprung mass; m_2 is the sprung mass; k_1 is the tire linear stiffness; k_2 is the suspension linear stiffness; c_e is the viscous damping coefficient. According to Newton's laws of motion,

corresponding semi-active suspension kinematic equations can be obtained:

$$\begin{cases} m_2 \ddot{x}_2 = -c_e(\dot{x}_2 - \dot{x}_1) - k_2(x_2 - x_1) - F_{MR} - F_p \\ m_1 \ddot{x}_1 = c_e(\dot{x}_2 - \dot{x}_1) + k_2(x_2 - x_1) + F_{MR} + F_p - k_1(x_1 - x_0) \end{cases} \quad (16)$$

Where:

$$F_p = \left(p_a - \frac{p_0 V_0}{V_0 + (x_2 - x_1) A_g} \right) A_g \text{sgn}(\dot{x}_2 - \dot{x}_1) \quad (17)$$

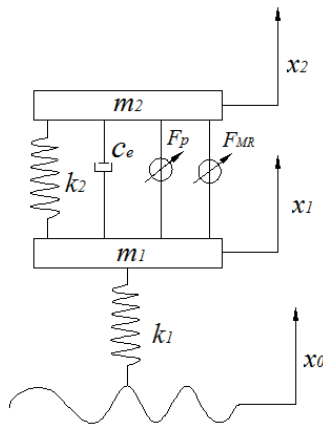


Fig.7 MR damper semi-active suspension dynamics model

Through the above analysis, dynamics Modelica model of MR damping system has been shown in Figure 8.

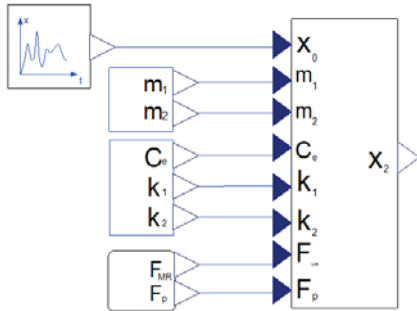


Fig.8 Dynamical Modelica model of MR damping system

3.4 Control system model

Since the MR damper is unlike with conventional damper, damping force size can be controlled directly by the current. In order to reduce the vibration and shock caused by road, the vertical acceleration of the vehicle body must be controlled within a certain range, otherwise it will influence ride comfort and stability of manipulation. Control method of vertical acceleration of the vehicle body adopt PID control, PID control equation is as follows:

$$u = K_p(e(t) + \frac{1}{T_i} \int_0^t e(t)dt + T_d \frac{de(t)}{dt}) + u_0 \quad (18)$$

Where $e(t)$ is the input signal, $e(t) = \ddot{x}_{2d} - \ddot{x}_2(t)$, \ddot{x}_{2d} is the desired body acceleration values, $\ddot{x}_2(t)$ is the calculated body acceleration values, u_0 is the initial value of control, k_p , T_i , T_d are control parameters. The PID controller operates briefly as follows: the body acceleration signal transfer to the PID controller, after PID control system processed the body acceleration signal, the adjusted current signal is input to the damping force model to adjust the damping force. Difference $e(t)$ between the desired body acceleration and the calculated body acceleration is input, and the control current signal is output. PID control module model as shown in Figure 9:

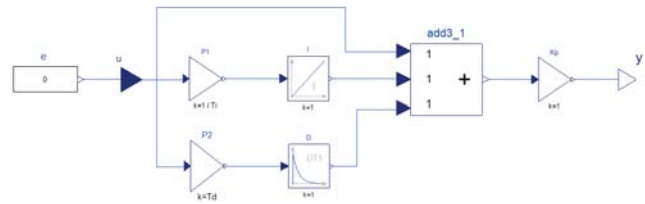


Fig.9 PID Control system Modelica model

3.5 MR damper system integrated model

By using Modelica language to construct models at MWorks platform, each sub-model is independent of each other on describing the physical characteristics and mathematical relations. In order to ensure the reuse of sub-models, each sub-model has its own input and output connectors to input parameters and communicate with other sub-models. Meanwhile, a sub-model can be composed of other sub-models and components by packaging, which is the multi-level modeling method. Multi-level modeling has the advantages of clearly organized, reusable, high modeling efficiency etc.

The simulation process of the integrated MR damper model is as follows: Firstly, vibration excitation is detected by the wheel sensors, and then they are delivered to MR damper semi-active suspension dynamics model to calculate the body acceleration. The difference value between the body acceleration signal and the desired body acceleration is delivered to the PID controller, PID controller output a control signal to the magnetic field model and directly control the input current to adjust the damping force, further to reduce the body acceleration and reach optimum damping state. By modeling, packaging and connecting all the sub-models of the MR damper at MWorks, the integrated model is obtained and shown in Figure 10.

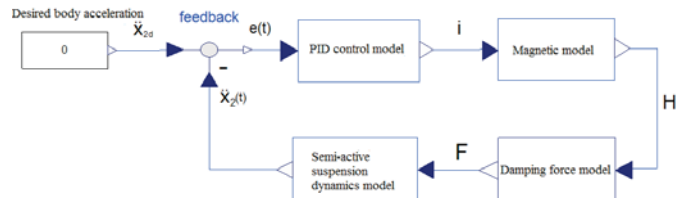


Fig.10 MR damper system integrated Modelica model

4 Simulation results and analysis

4.1 The basic parameters of the suspension system

The basic parameters of the suspension system: $k_1 = 160000 N/m$, $k_2 = 16000 N/m$, $m_1 = 30 kg$, $m_2 = 210 kg$. Cylinder and piston use 40Cr steel and 20 steel with high magnetic permeability respectively.

4.2 Effect of various parameters on the damping properties

As can be seen from the MR damper system integrated model, main factors which influences the damping performance of MR damper: structural parameters、magnetic circuit parameters、material

parameters and excitation current. we could simulate the performance of MR damper by changing the values of the parameters under different conditions, as shown in Figure (11 to 13).

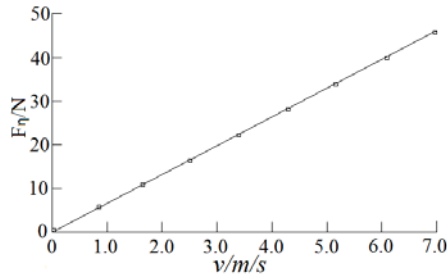


Fig.11 The values of F_{η} under different values of piston speed

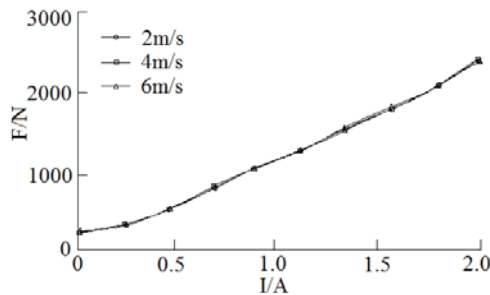


Fig.12 The values of F under different values of current

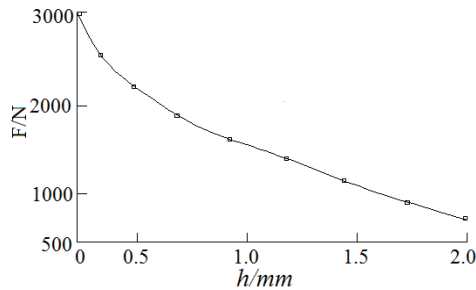


Fig.13 The values of F under different values of the damping passage gap

The value of F_{η} under different values of piston speeds as shown in Figure 11. In the case that other parameters are fixed and magnetic field does not work, the viscous damping force F_{η} and piston speed v has a linear relationship, the greater of v , the greater of F_{η} . But the viscous damping force account for only a small portion of the total damping force, as shown in Figure 12. Three curves almost coincide at different piston speeds in Figure 12, the small difference is only affected by the viscous damping force. Therefore, the total damping force of MR damper is mainly provided by the Coulomb damping force F_{MR} , and it increases with excitation current increasing. Therefore, by adjusting the magnitude of excitation coil current, we can adjust the size of the total damping force.

The damping passage gap not only influences the magnetic field strength H to change the Coulomb damping force F_{MR} , but also directly influences the viscous damping force F_{η} . The values of F under different values of the damping passage gap shown in Figure 13. the smaller the damping passage gap, the greater the total damping force. However, considering the difficulty of manufacturing and assembly, as well as MR fluid effect, the damping passage gap can not be too small.

4.3 Analysis of the control performance

On the condition that the vehicle speed is 20km/h, 40km/h and 60km/h on C-class road excitation, the comparisons between the control strategy based on modeling methods used in this paper and the control strategy based on Matlab/ Simulink model[12] were compared. As shown in Table 1:

Tab.1 The comparison from control strategies based on the different modeling methods

C-class road excitation	Sprung mass acceleration/($m \cdot s^{-2}$)		
	20km/h	40km/h	60km/h
Skyhook control based on simulink model	0.663	0.894	1.263
Fuzzy control based on simulink model	0.752	0.962	1.424
PID control based on Modelica model	0.637	0.827	1.137

As can be seen from Table 1, compared to the Skyhook control strategy and the fuzzy control strategy based on simulink modeling, the PID control strategy based on multi-domain unified modeling under Modelica language of the proposed can obtain better results and its results are better with the increasement of vehicle speed. The Skyhook control strategy can better reflect the internal properties of MR fluid damping system compared with the fuzzy control strategy, which can obtain better control performance. The Skyhook control strategy based on

simulink modeling and the PID control strategy used in this paper can obtain a similar control performance, but the algorithm of the Skyhook control strategy is more complicated compared to the PID control, and its operability is inconvenient, but its control performance is better, so the model based on multi-domain unified modeling under Modelica language of the proposed can better reflect the complex relationship compared with simulink modeling between the MR damper coupling among the parameters within the system, and can get a

better simulation results. With the increasement of vehicle speed, the modeling method in this paper based on the Modelica language compared with the modeling methods which has put forward, the control effect is more obvious. Thus, it shows that control effect of control strategy is better based on multidisciplinary unified modeling for complex electromechanical system.

5 Conclusions

MR damper semi-active suspension system is a typical multi-domain coupling complex electromechanical system. It has been working mainly on a single field to complete modeling and adopted traditional single field simulation tools, such as ANSYS and Matlab, to simulate. In this paper, it has been taking the coupling relation between various fields into consideration, whose coupling relation involves structure, electromagnetic, mechanical and fluid dynamics and other fields. The results show that the model can better reflect the complex relationship between the MR damper coupling among the parameters within the system, and can get a better simulation result.

Acknowledgements

This study was supported by National Natural Science Foundation of China (Grant No.51275141).

References

- [1] Kaldas M, Caliskan K, Henze R, et al. Rule optimized fuzzy logic controller for full vehicle semi-active suspension[J]. SAE Int. J. Passeng. Cars-Mech. Syst., 2013, 6(1): 332-344.
- [2] Mihai I, Andronic F. Behavior of a semi-active suspension system versus a passive suspension system on an uneven road surface[J]. Mechanics, 2014, 20(1): 64-69.
- [3] Tseng H E, Hrovat D. State of the art survey: active and semi-active suspension control[J]. Vehicle System Dynamics, 2015, 53(7): 1034-1062.
- [4] Yang L Q, Chen W W, Gao Z G, et al. Nonlinear control of quarter vehicle model with semi-active suspension based on solenoid valve damper[J]. Structural and Multidisciplinary Optimization, 2014, 45(4): 1-7.
- [5] Sulaiman S, Samin P M, Jamaluddin H, et al. Tyre Force control strategy for semi-active MR damper suspension system for light-heavy duty truck[J]. International Journal of Vehicle Autonomous Systems, 2015, 13(1): 65-90.
- [6] Peng Z Z, Zhang J Q, Yue J, et al. Design and analysis of magnetorheological damper paralleling with constant throttling orifices[J]. Journal of Mechanical Engineering, 2015, 51(8): 172-177.
- [7] Imaduddin F, Mazlan S A, Zamzuri H. A design and modelling review of rotary MR damper[J]. Materials and Design, 2013, 51: 575-591.
- [8] Kasprzyk J, Krauze P. Vibration control for a half-car model with adaptation of the MR damper model[C]//Proceedings of 2014 International Conference on Modelling, Identification and Control, January 23, 2015, ICMIC 2014: 243-248.
- [9] Ma R, Zhu S H, Liang L, et al. Modelling and testing of magnetorheological damper[J]. Journal of Mechanical Engineering, 2014, 50(4): 135-141.
- [10] Zhao J J, Ding J W, Zhou F L, et al. Modelica and its mechanism of multi-domain unified modeling and simulation[J]. Journal of System Simulation, 2006, 18(S2): 570-573.
- [11] Wu Y Z, Wu M F, Chen L Q. Study on the Hybrid Modeling Platform Based on Modelica Language for Complex Machinery System[J]. China Mechanical Engineering, 2006, 17(22): 2391-2396.
- [12] Li X S, Liu M A. Research on control and performance of base on MR semi-active suspension in automobile[J]. Journal of Central South University of Forestry&Technology, 2011, 31(9): 143-147.

A model against crime: Crime and intelligence led policing in Nigeria

Okonigene Dorcas¹, Okonigene Robert², John Samuel³, Agbator Austin⁴, and Agbator Eunice⁵

¹Department of Physical and Health Education, Ambrose Alli University Ekpoma, Nigeria

²Department of Electrical and Electronics Engineering, Ambrose Alli University Ekpoma, Nigeria

³Department of Electrical and Information Engineering, Covenant University, Ota, Nigeria

⁴Private and Property Law, Faculty of Law, Ambrose Alli University Ekpoma, Nigeria

⁵Public Law, Faculty of Law, Ambrose Alli University Ekpoma, Nigeria

Abstract - This paper presents a review of some current technology deployed by the Police, worldwide, to combat crime. A holistic study was carried out on how governments and their security agents combat organized crime. Also, examined and critically analyzed are the structures and duties of the police departments charged with the responsibilities of combating crime. Hence, this research reviewed how these Departments function in the United States of America (USA), Canada, Australia, United Kingdom, Europe, and Israel. These studies led to this proposed model that shows how the Nigerian police can effectively combat crime. The model tends to profound solutions to the defects in its level of technological detection, prevention and investigation of crime. The proposed simple model tagged as "Intelligence led Policing" create databases and real time dynamic network linking all the Police Departments, patrol teams, units, members of the public, other security agents and also undercover agents together. Although the study is still ongoing, early simulated results of this model show its feasibility which however indicates high cost of real live implementation.

Keywords: Police and crime, intelligence policing, internal security, human rights, area command.

1 Introduction

For decades governments all over the world have intensified efforts to provide law enforcement agents with the tools to investigate criminal organizations and to otherwise aid in the fight against criminal elements in the society. To effectively combat crime the law enforcement agents have to ensure that it: (i) develop crime prevention strategies; (ii) co-ordinate national and regional initiatives; (iii) undertake research and analysis; (iv) engage in public education [1].

In the United States, law enforcement agencies can legally monitor the movements of people from their mobile phone signals using Stingrays. In the USA the FBI, DEA, Secret Service, National Security Agency (NSA), U.S. Marshals Service, Immigration and Customs Enforcement (ICE) and

ATF as well as the U.S. Army, Navy and Marine Corps all use Stingrays [2,3,4].

Nigeria police was first established in 1820 and in 1963, under the First Republic, these forces were nationalized. The Nigeria Police Force duties were conventional police functions and were responsible for internal security. The Force also performed military duties outside Nigeria.

Section 214 of the 1999 constitution, of the Federal Republic of Nigeria, specifies the Nigeria Police as the national police of Nigeria with exclusive jurisdiction throughout the country. The Nigeria Police (NP) is the principal law enforcement agency in Nigeria with a staff strength of about 371,800 [5,6,7]. Section 215 of the constitution empowers the Inspector General of Police with the general operational and administrative control of the Nigeria Police.

The general assumption is that the police are legitimate, officially articulated organizations that can use force to sustain political and civil order. However, the Nigeria Police since 1999 has failed in its constitutional duties [8].

Unfortunately what have characterized the Nigerian Police are cases of:

- Indiscipline: improper dressing, consumption of alcohol in glare public while on duty, lack of respect of junior officers to senior officers, receiving bribes, drunk on duty, human right abuses and extrajudicial killing [9,10].
- Lack of insurance cover: Poor budget funding, misappropriation of security votes.
- Welfare: Very poor housing for the police, poor salary and allowances, no enough vehicles for patrolling or conveying members.
- Network: Lack of citizen's biometrics, forensic lab is without database, patrol officers lack network contacts, use of media to identify suspects is absent, NPF website has gross inadequate information for the public.
- Logistics: NPF helicopters, armored personnel carriers, light weapons, heavy machine guns, arms and

ammunitions, communication equipment, are grossly inadequate.

- Most police posted to polling units, on election day connive with hoodlums to aid politicians to falsify election results
- Police illegally arrest citizens and detain them in order to extort money from them
- Police extortion of motorist is with impunity

The NPF has seven Area Commands and five Departments (Department Criminal Investigations, Department Logistics, Department Supplies, Department Training, and Department Operations).

The Department Criminal Investigation (DCI) is the highest criminal investigation arm of the Nigeria Police [11]. DCI is tasked with investigation and prosecution of serious and complex criminal cases within and outside Nigeria. The DCI has the following sections with most of them headed by Commissioners of Police (CPs): i) Administration, ii) Anti-Fraud Section, iii) The Central Criminal Registry (CCR), iv) Special Anti-Robbery Squad (SARS), v) X-Squad, vi) General Investigation, vii) Special Fraud Unit (SFU), viii) Legal Section, ix) Forensic Science Laboratory, x) Interpol Liaison, xi) Homicide, xii) Anti-Human Trafficking Unit, xiii) Force Intelligence Bureau (FIB), xiv) DCI Kaduna Annex.

There are several eyewitness reports in social, electronics and print media of Nigeria police collusion with criminals. In most police stations there are bills well-advertised stating that bail is free. But the reality is that for these corrupt officers bail is never free. Accused persons are intimidated by the police and are threatened to pay a ransom for their bail. The relationship between the police and the general public is so bad. Despite all appeals to police officers to change their attitude towards the public, to be fair and honest, and to avoid corrupt practices they remain defiant.

Between the year 2013 and 2015 the crimes committed such as kidnapping, armed robbery, murder, arson, extrajudicial killing, rape, armed banditry, violent militant groups, religious insurrection and stealing of public funds with impunity were on the increase.

2 Methodology

The aim of this study is to review studies that have assessed the effectiveness of crime control strategies, by the police, and develop a model against crime for the Nigeria Police. This study review covers a broad range of research methods and data sources. This includes review studies by the Department of Justice in Canada, USA, UK, Europe and Israel on organized crime control strategy. Also reviewed are

studies by academics and law enforcement agencies. Most documents for this study literature review were from the search conducted with the help of several electronic databases.

Most of the documents provided empirical evidence on crime control strategies, also included are definitional and methodological issues bearing on evaluations in this area.

This study reviewed each crime control strategy and critically examined the merits of the different control strategies to combat crime.

The study resulted in the development of a model for the Department Criminal Investigation (DCI) Nigeria Police. The developed model took advantage of today's advancement in technology to produce an active network. The model is a combination of different modules/sections actively networked together. Some characteristics of this developed model are: "Intelligent led policing" app, crime record databases, biometric databases, interactive web pages, communication security, internet links and mobile network app.

3 Educating Police Officers

The culture of relevant authorities to turn a blind eye to Police officers corrupt practices with impunity and the continued failure to train police officers properly has resulted in complete lack of public trust.

This study also examined how members of the Nigeria Police are trained, their educational background vis-à-vis duties. The Nigeria Police Academy (NPA) was established to train police cadets. There is no known University in Nigeria that specializes in the training or education of professionals for policing.

If some of the courses offered in NPA are introduced in public schools it will help to produce better educated recruits. Police recruits are trained at Police colleges in Oji River State, Maiduguri Borno State, Kaduna Kaduna State, and Ikeja Lagos State [10, 11]. The Police also have provision for in-service training schools, including the Police Mobile Force Training School at Guzuo, southwest of Abuja, the Police Detective College at Enugu, the Police Dogs Service Training Centre, and the Mounted Training Centre.

With proper education the NPF will achieve the following:

- Provide safety and security in Nigerian communities; protect and respect human rights, and promote community partnership in preventing and controlling social disorder.

- Will be a leading national, professional and efficient law enforcement organization.
- Reduced cases of extrajudicial killings
- Improve on Police public relationship by organizing educative programs for well-informed citizens on their role in combating crime.
- Effectively combat drug trafficking, economic crimes, high-tech crimes, money laundering, illegal immigration and trafficking of humans, and corruption
- Maintain discipline within its ranks
- Avoid illegal road blocks. Also this will **reduce** cases of NPF extortion of motorist
- Reduce cases of illegal arrest and detention
- Ability to cope with modern technology to combat crime

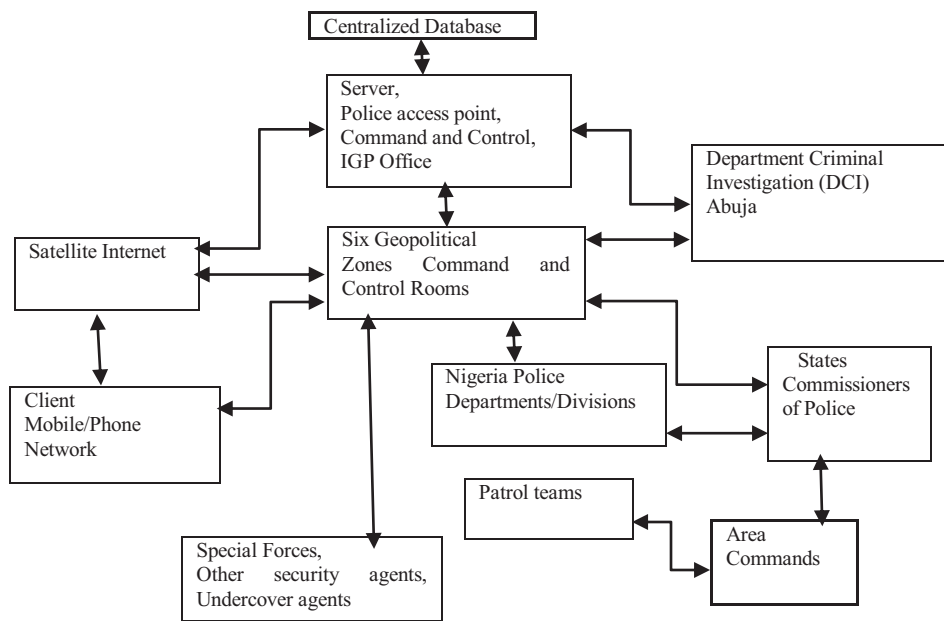


Figure 1 Intelligent led policing management for Nigeria Police

4 Intelligence Policing led Model

The DCI lacks adequate and proper intelligence gathering mechanism to support and interpret criminal's data.

The sharing of information among the various police departments are isolated and with no proper or efficient networked databases. This led to lack of adequate intelligent among patrol officers and other undercover agents.

This study therefore proposes a model based on the concept of active networking. The essence is to build a network that is easier to change and customize. Hence, adding new functionality to the network in future will result in minimal disruption to existing services. The advantage of this model design is its flexibility. The command and control room will serve as the network management station. Here information gathered are compared to data stored in the database, concerning previous crime committed, and including those

involved as well as the nature and level of investigation carried out are established and analyzed. Figure 1 shows simple components of the proposed intelligent led policing management for the Nigeria Police.

The security of a network is said to be very good if the network security is able to reliably authenticate communication partners and other network entities [12]. The security provided identifies the recipients of information within this network. The fundamental services provided by this unit are confidentiality and authentication.

Database is sited in each of the six geopolitical zones. In this model the current 36 States, the 774 Local Government Councils and the Federal capital territory are networked to these databases. Each of the databases is linked together through the centralized database. With the existence of these

databases crime analysis will help, by early detection, to prevent, reduce, and control crime and disorder in Nigeria. Nigeria Police currently lacks these infrastructures.

A characteristic of this propose model is an automatic response that ensures that immediately a call is linked to the network, in any part of Nigeria, the satellite immediately search the area and record vital data for analysis. The system immediately alerts the nearby command and all the chain of command that are required to take action. The members of the police unit responding to the distress call are guided by the system to the search area. Even when the caller dials and then switch off the phone the system will be able to act effectively. The caller identity can only be disclosed to the IGP after due legal application process. By simply dialing a three digit number from any network the system will respond. The model is built to respond to reported cases of crime. If the caller is in danger to speak, then by dialing the three digits is enough for a quick response from the system. Any statement will be very useful. The three digit number response is independent of network service provider. We are equally developing an application (called "Police and Me App") for this purpose that will be freely downloaded by mobile phones. Members' of the public can interact with the system from the comfort of their homes and from anywhere in the country via mobile phone, land phone or Internet connectivity.

The software application is web based and enables tracking of caller location and area. Hypertext Markup Language (HTML), Hypertext Preprocessor (PHP), Javascript, Dreamweaver and MySQL were used to realize the interface and Web Based solutions for the automated model.

Cases of inmates with minor offences that are illegally detained in Nigerian prisons for so many years without trial will easily be detected.

5 Conclusions

The search for materials began with a search of several major electronic databases. This report was not about defining crime. The different circumstances under which crimes were committed, however, form the basis for developing this model.

The study is still in its early stage. The model is being developed in modules and the entire study when completed is expected to combat crime, such as insurgencies, terrorism, armed robbery, murder and kidnapping. We do not intend to give detail information about the firewall nor how the model detect and identify a criminal/crime.

References

- [1] Department of Justice, Canada, "Assessing the Effectiveness of Organized Crime Control Strategies". http://www.justice.gc.ca/eng/rp-pr/csj-sjc/jsp-sjp/rr05_5/p1.html
- [2] Bott, Michael; Jensen, Thom (March 6, 2014). "9 Calif. law enforcement agencies connected to cellphone spying technology". ABC News, News10.
- [3] Richtel, Matt (December 10, 2005). "Live Tracking of Mobile Phones Prompts Court Fights on Privacy" (PDF). The New York Times. .
- [4] <http://www.tomsguide.com/us/cellphone-tracker-stringray,news-21718.html>
- [5] 1999 Constitution of the Federal Republic of Nigeria
- [6] Interpol <http://www.interpol.int/Membercountries/Africa/Nigeria>
- [7] Vanguard Newspaper. "Jonathan sacks IGP Suleiman Abba, appoints DIG Solomon Arase".
- [8] Manning P. K., "Technology's Ways: Information Technology, Crime Analysis and the Rationalizing of Policing". <http://crj.sagepub.com/content/1/1/83.abstract>
- [9] <http://www.nigeriapolice.org/mobile-police.html>
- [10] <http://news.bbc.co.uk/2/hi/africa/1322017.stm>
- [11] "Departments || Nigeria Police". www.npf.gov.ng.
- [12] Savo Glisic and Beatriz Lorenzo, "Advance Wireless Network: Cognitive Cooperative and Opportunistic 4G Technology" second Edition, John Wiley and Sons, Ltd, 2009.

FLOWER INSPIRED THUNDER PROTECTING UMBRELLA

Kuldip Acharya¹, Dr. Dibyendu Ghoshal²

¹Computer Science and Engineering Department, National Institute of Technology Agartala, Agartala, Tripura, India

²Department of Electronics and Communication Engineering, National Institute of Technology Agartala, Agartala, Tripura, India

Abstract - *The present study has dealt with an innovative idea regarding thunder protecting umbrella. The proposed umbrella can be folded and unfolded smoothly, and an animation algorithm is made to mimic the blooming of flower petals. The proposed umbrella is capable of protecting the user from any thunderstorm or lightning of any magnitude by providing a shielded conducting chord from the apex of the umbrella to the conducting spikes fitted at the bottom most layer of the shoe. The use of such an umbrella may be expected to provide a sound protection of the user to move within full of frequent thunder fall and lightning. The function of the proposed umbrella has been shown through computer animation. The movement of the user is easy in the presence of long flexible thin cable with appropriate connector jacks. The proposed design if manufactured at an industrial level may find some commercial utility also.*

Keywords: Animation, Flower, Protecting, Thunder, Umbrella.

1 Introduction

An umbrella is an essential appliance, which is always used irrespective of any country, climate and geographical location in the world. The main problem for using umbrella is that under severe lightning and thunder the person holding the umbrella can get an electric shock and with very much fatal. The material, construction, and the area are among various factors related to the umbrella. Out of this, materials of the constituent's parts of the umbrella can play a vital role to make it thunder protecting. Hence, a thorough study umbrella [1-4] material draws a significant attention in the construction of the umbrella to save human lives. For the optimization of various parts of the umbrella can be considered as a paramount topic well deserving to address. A large number of bio-inspired optimization techniques have been already reported and are available in current journals [5]. In the present study flower inspired thunder protecting umbrella has been proposed. The methods indicate the folding and unfolding of a typical umbrella in the computer animated version based on Autodesk Maya software [6,7] has been carried out in the present study. In the present computer

animation, based study a double layer umbrella sheet has been proposed. These two sheets are pasted with the help of strong adhesive which itself is an insulator i.e. it prevents the throw of electric current and voltage through it. The material of the outer sheet of the umbrella is chosen such a way that it should be flexible and malleable so that the umbrella can be folded and unfolded easily. The material of the outer sheet is proposed to be made of polyvinyl chloride (PVC) [8] in addition to few other materials can be used, but the cost may be higher. PVC is reported to be very user-friendly and have wide used in the plastic and rubber industries. For the present design, the fast and foremost needs are that they should be cheap and easily adaptable. The top portion of the umbrella, which protrudes, is a constant part of the umbrella handle. The breaths, width, length of umbrella stick are proposed to be prepared or manufacture as per the conventional dimension. The standard dimension of the umbrella and it is different components is found in [9,10]. This kind of work has not been found in any journal or online research article.

2 Methods

The top portion of the umbrella which is protruded outside is made up of ferromagnetic material [11]. In this regard still or iron may be used for this purpose. The electricity carried out by the thunder or lightning passes through the iron, and this goes to the spike (made up of the insulator) of the shoes through an insulator wire with the high quality of insulation. When any thunder is attracted by the umbrella its safely passed to the ground without affecting the person holding the umbrella. The setting of the insulator wires under the umbrella cloth to spikes of the shoes is fashioned in such a way so that it should lose enough to enable the person to place his steps. Sufficient provision should be there so that the individual can move toward the back and towards the side and he can move through the 360 degrees around the axes of the body. A flexible clip covered with an insulator fixed at the end of the insulator wire so that the wire can be kept within the umbrella in a wound form. When thunder comes, the wound wire is unwounded. All the elements should be set is such a way so that the folding and unfolding of anti-thunder protection become very easy. When the thunder falls in the top of the umbrella, the apex which is a conductor and it only

pass the high current to the ground through the plastic insulator shielded thin cable. The other end of the cable can be connected with the shoe made up of foam leather or canvas. The bottom part of the shoe is laminated with an insulating material like polyvinyl plastic or alike. Below the insulating layer, there are metallic spikes which can conduct current. The tip of the cable is fitted with a small connector jack, and this is inserted into a small drilled hole with conducting walls. The hole continues up to the spike and spikes are internally connected through thin conducting wire (encased in the insulating layer below the shoe as mentioned above). The metallic conducting spike is well suited to pass the current generated by the current.

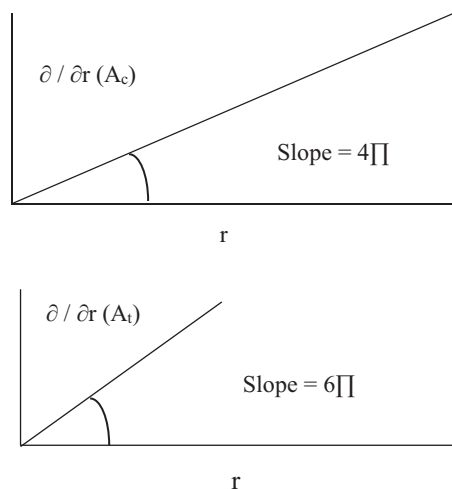
2.1 Maximization/minimization length of the umbrella area

The area $A_c = 2\pi r^2$ for the curved surface. The area $A_t = 3\pi r^2$ for the total surface. Both are functions of radius. Area of the full sphere = $4\pi r^2$. Area of the hemisphere (curved surface) = $2\pi r^2$. Total area of the outer surface area = $2\pi r^2 + \pi r^2 = 3\pi r^2$.

$$\begin{aligned} 1. \quad & A_c = f(r), A_t = f(r) \\ \text{or, } & \partial A_c / \partial r = \partial / \partial r \{2\pi r^2\} = 4\pi. \quad (2r) = 4\pi r \\ \text{or, } & \partial^2 A_c / \partial r^2 = \partial^2 / \partial r^2 (4\pi r) = 4\pi \Rightarrow \text{constant.} \quad (1) \end{aligned}$$

$$\begin{aligned} 2. \quad & A_t = f(r) = 3\pi r^2 \\ \text{or, } & \partial A_t / \partial r = \partial / \partial r (3\pi r^2) = 6\pi. \quad (2r) = 6\pi r \\ \text{or, } & \partial^2 A_t / \partial r^2 = \partial / \partial r [6\pi r] = 6\pi \Rightarrow \text{constant.} \quad (2) \end{aligned}$$

Equation 1 and 2 shows that the derivatives are proportional to the radius.



The constant value indicates that the function does not have minimum or maximum and shows a constant value.

- i. $D^2y / dx^2 = \text{Negative} \rightarrow \text{Maximum.}$
- ii. $D^2y / dx^2 = \text{positive} \rightarrow \text{Minimum.}$

During the manufacture of such an umbrella, the one fundamental question is lying unsolved. The initial of the

smaller diameter the length of thinner diameter of the stick is to be determined so that an optimized value (minimum or maximum can be reached).

3 Results and explanation

The shape of the umbrella is taken as hemispheric to provide better protect from the rain. There should be a consistency between the area of the outer layer of the umbrella and length and diameter of the inner stick. Special care has been taken in the apex area of the umbrella which serves a dual purpose. The first is this that it divides extra mechanical strength when the third layer of the small diameter of polyvinyl chloride material is pasted by using a strong adhesive. The apex point of the outer protruded portion of the handle has got great importance to be effective of thunder and lightning as its quite well known that this pointed portion is fast effected by thunder.

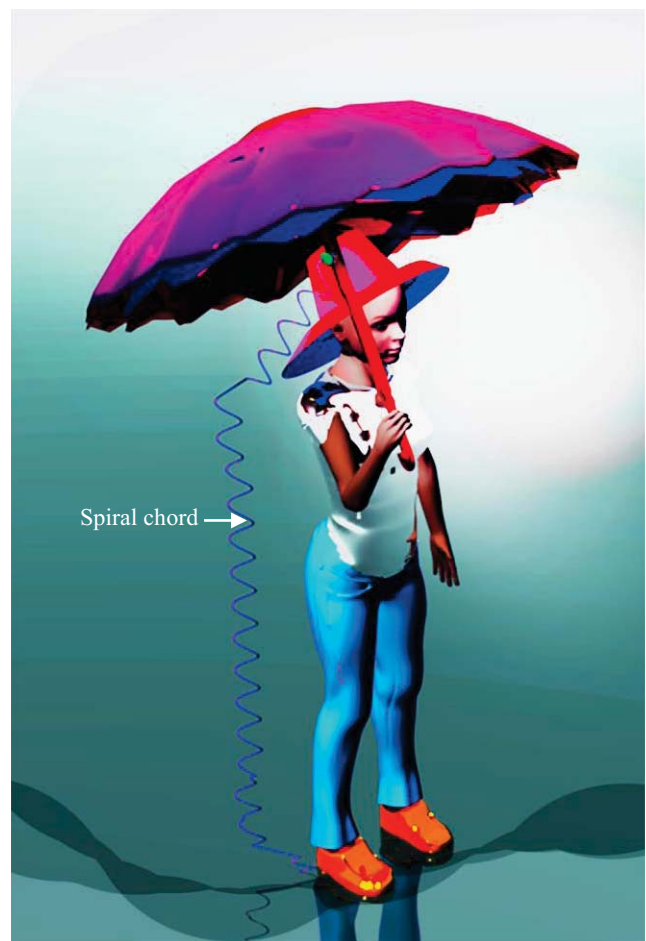


Fig. 1. Different components of thunder protecting umbrella i.e., spiral chord, shoes and the proposed umbrella itself.

The function of the proposed thunder protecting umbrella has been shown through 3D (three-dimensional) modeling and computer animation by using Autodesk Maya student version software.



Fig. 2. A rainy scene with an umbrella.

Fig. 2 shows the utilization of proposed umbrella to provide security from the thunder shower, downpour, and lightning.



Fig. 3. Spiral chord top end jack inserted in the shaft connector.

Fig. 3 shows a light, adaptable length of spiral cable and a jack.

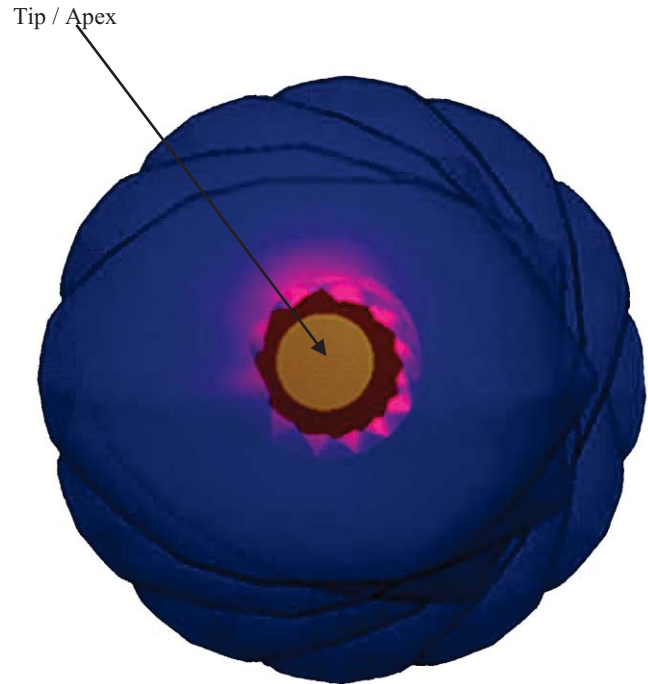


Fig. 4. Tip of the umbrella.

Fig. 4 shows the apex or tip of the umbrella which is specially designed to protect the users both from the thunderstorm and lightning.

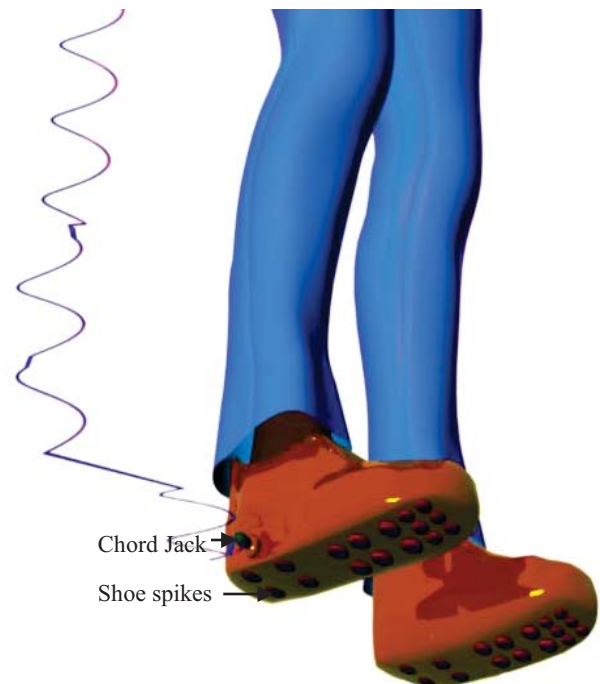


Fig. 5. Spirel chord bottom end jack inserted in to the bottom most layer of the shoe which is connected with the shoe spikes.

Fig. 5 shows the conducting spikes fitted at the bottom most layer of the shoe made up of plastic leather, canvas, etc.

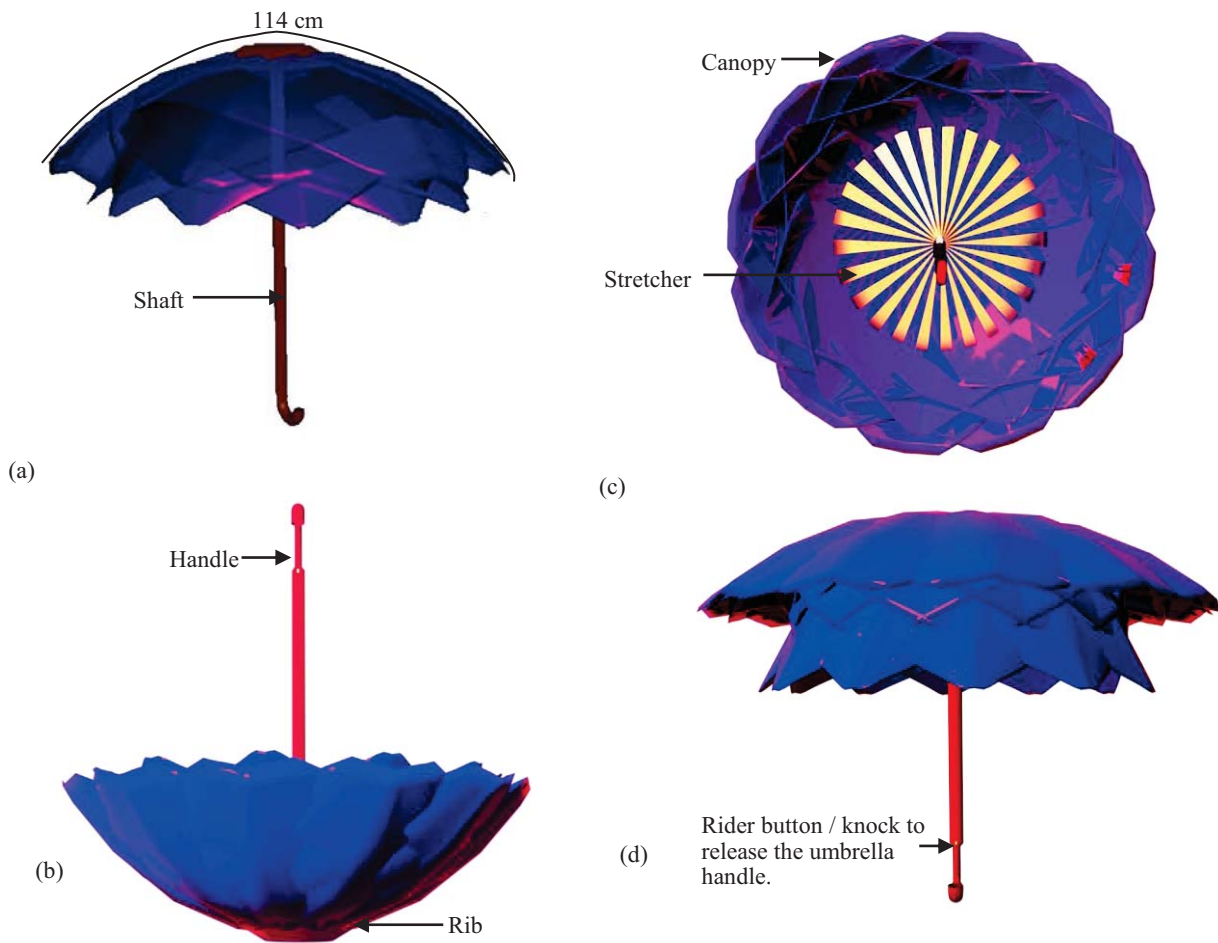


Fig. 6 (a) (b) (c) (d). *Thunder protecting umbrella's views from different angles and it's different components.*



Fig. 7. *Another rain scenario of the thunder protecting umbrella.*

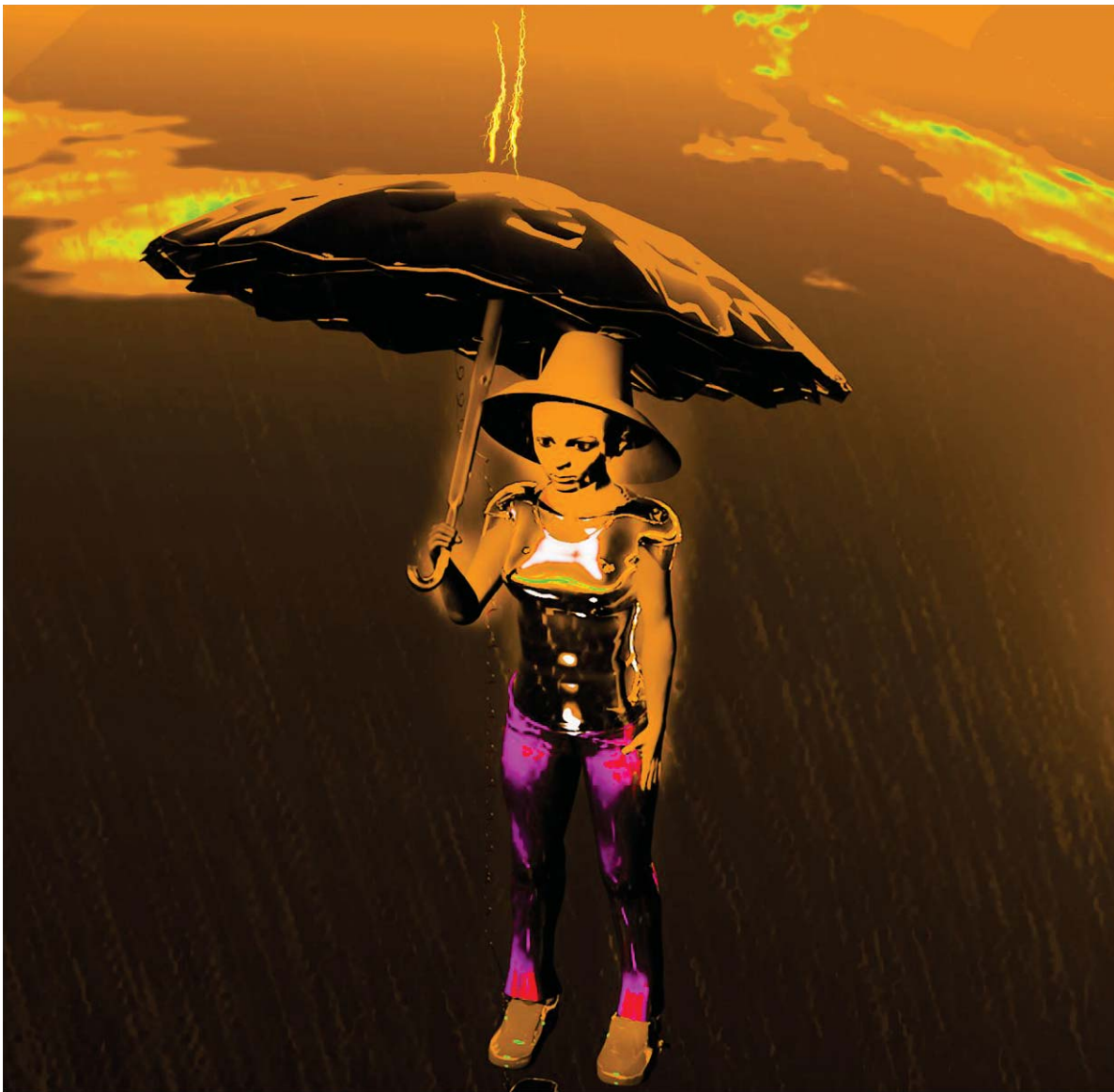


Fig. 8. A scenario where the thunder protecting umbrella is protecting the user from the lightning effects.

Table I. Sizes of umbrella components

<i>Umbrella components size</i>	<i>Approximate values</i>
Umbrella total height	80 cm
Umbrella stretcher length	57 cm
Canopy maximum radius when it open	114 cm
Handle height	10 cm
Shaft width	2 cm
Rib total radios when umbrella is open	107 cm
Spring spiral chord length	198 cm
Tip size	10 cm

Table II. Umbrella components

<i>Umbrella components</i>	<i>Materials</i>
Top tip	Steel
Shaft	Fiberglass
Handle	Wood, rubber
Canopy	Lightweight polyester fabric
Rid	Steel
Rider button / knock	Steel
Tip	Plastic
Wires	Spring spiral chord made of High-quality solid copper
Tip	Fiber plastic
Rider	Fiber plastic
Top spring	Steel

Table III. *Approximate weight of different components of proposed thunder protecting umbrella.*

Component names	Approximate mass (gram)
Cloth	65
Screw and Nuts	1
Release button	4
Rider	11
Rib	<1
Handle	38
Spiral chord	1

The total approximate cost of the proposed thunder protecting umbrella is Rs. 1000 and the shoe price is assuming Rs. 600. The approximate total weight of the proposed, "Thunder protecting umbrella" is 600 grams.

4 Conclusions

A computer animation has been created by using Autodesk Maya student version software to introduce a novel thunder protecting umbrella. The proposed umbrella has been designed by mimicking the flower blooming, and a safety cable is used between the metallic apex part of the umbrella and the spikes of the shoe of the user. The cable has been designed in such a method so as to make it light, flexible, fully safe, having optimum length and user-friendly. The connecting cable will provide a path for the passage of the current caused by the thunder or lightning to pass to the ground through the conducting spikes and connectors to be inserted into the narrow hole touching the spikes which are interconnected. The designed umbrella can be well used by both male and females and during non-thunder or sunny condition also. Provision is kept to wind up the cable and keep it under the umbrella sticks during normal condition. The animation version would provide a visual representation to the practical designers and can find a good commercial response.

5 Acknowledgement

The author one greatly acknowledges National Institute of Technology Agartala, India for providing Ph.D. fellowship.

Dedication

Kuldip Acharya (author one) dedicates his creative work to his father Dr. Kalidas Acharya residing into the eternity of love.

6 References

- [1] Apple, Phillip C. "Rotating canopy umbrella." U.S. Patent No. 5,020,557. 4 Jun. 1991.
- [2] Di Cesare, John David. "Umbrella." U.S. Patent No. D503,036. 22 Mar. 2005.
- [3] Castano, Francisco. "Base for an umbrella." U.S. Patent No. 5,060,907. 29 Oct. 1991.
- [4] "Folding multiple rigid section umbrellas." U.S. Patent 2,967,379, issued January 10, 1961.
- [5] Valdez, Fevrier. "Bio-Inspired Optimization Methods." Springer Handbook of Computational Intelligence, 2015. pp.1533-1538.
- [6] Palamar, Todd. "Mastering Autodesk Maya 2016", Autodesk Official Press, John Wiley & Sons, 2015, pp. 1.
- [7] Autodesk Maya 3D animation student version software [online]. Available : <http://www.autodesk.com/education/free-software/maya>. © 2016 Autodesk Inc.
- [8] "Poly (vinyl chloride) (CHEBI :53243)". CHEBI. Retrieved 12 July 2012.
- [9] ALL small - ST Umbrellas, [online]. Available : <https://stumbrellas.co.za/wp-content/uploads/2013/06/2014-catalogue.pdf>, pp. 10-33.
- [10] James Carver Umbrellas, Parts of an Umbrella [online]. Available : www.umbrellaman.co.uk/page/parts-umbrella.htm, February 28, 2007.
- [11] Freiser, Marvis J. "A survey of magneto-optic effects." Magnetics, IEEE Transactions on 4.2 (1968) : 152-161. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

Modeling the Effect of V_{th} -Variations on Static Noise Margin

Azam Beg

College of Information Technology, United Arab Emirates University, Al-Ain, United Arab Emirates

Abstract—The threshold voltages of CMOS logic gates based on modern technology nodes are highly susceptible to variations. As a result, the static noise margin of the gates also varies. In this paper, we present an analytical model for representing the variations in the noise margin. The model can serve as an expedient alternative to Monte Carlo simulations. With three use-cases, we have demonstrated how the model can be used as a first-order sizing mechanism for a variation-prone gate.

Keywords: CMOS logic, process variations, threshold voltage, static noise margin (SNM), mathematical models

1. Introduction

The increasing trend of number of devices on integrated circuits has faithfully followed the Moore's Law for many decades, which has been possible mainly due to the down-scaling of the device dimensions. This continuous down-sizing has brought into limelight the effects of process variability, for example, due to lithography, ion implantation, oxide thickness (t_{ox}), etc. A circuit's operational variations include signal cross-talk, power-supply noise, temperature, etc. One of the main process variants is the dopant density which manifests into fluctuations in the threshold voltages (V_{th}) of the devices.

A logic circuit is deemed reliable if its signal variations occur within the specified ranges of *logic-low* and *logic-high*. Incorrect evaluation of a single binary value can sometimes cause the entire circuit to fail. The sensitivity of a logic gate to input changes is usually characterized with the *noise margin* (NM).

Spice-based Monte Carlo (MC) simulations are commonly used for studying the effects of parameter variations in individual devices as well as circuits. Depending on the requirements, the MC simulation counts vary from a few hundred to tens of thousands; such simulations may run for many hours or even several days.

As an expedient alternative to the MC simulations, this paper presents a set of mathematical equations for characterizing the NM-variations in a CMOS inverter under V_{th} -variations. The paper is organized as follows: The literature related to our work is concisely reviewed in Section 2. To make this paper self-contained, the related fundamental concepts are included in Section 3. The equations for an inverter's NM *sans*-variations are derived in Section 4. The next Section 5 presents a set of NM equations when the inverter is subject to V_{th} variations. A few use-cases of the

equations are presented in Section 6, and the conclusions are in Section 7.

2. Related Work

Nussbaum [1] made an early attempt at analytically representing the statistical behavior of resistor-transistor logic circuits. Hill's [2] is among the earliest papers on NM of logic circuits. A few years later, Lahstroh *et al* [3] further explained and proposed mathematical representation of the NMs. Hauser [4] compared different definitions of NM present at that time, and proposed an alternative to the inflection point approach. Taylor and Fortes [5] estimated the standard deviation for V_{th} to find the transistor failure rates but no analytical models were discussed by them. Choudhury and Mohanram [6] proposed a method for finding the reliability of gate-based circuits, but did not cover how the failure rates for the individual gates were determined. The authors of [7] utilized the NM as one of the design parameters for low-energy circuits; their method relied on MC simulations to find the means and standard deviations of the NMs. Merino *et al* [8] proposed artificial neural network models for finding the NMs; but the creation of such models required large number of circuit simulations covering a huge design space. The gate-transistor sizing techniques in [9], [10] also relied on time-consuming MC simulations for determining the NMs. We have not come across any NM models that consider the process-related variations. Therefore, we deem ours to be the first known mathematical representation of the effect of V_{th} -variations on the NMs; the set of mathematical equations is proposed as an alternative to the time-intensive MC circuit (Spice) simulations.

3. Preliminaries

3.1 MOS Transistor Operation

A MOS transistor generally operates in three different regions: *cutoff*, *triode*, and *saturation*. Tables 1 and 2 summarize the *i-v* behavior of nMOS and pMOS transistors (nMOST and pMOST) in the three regions. For each transistor type, we define β as:

$$\beta = \frac{\mu \epsilon}{t_{ox}} \times \frac{W}{L} \quad (1)$$

where μ is the mobility of electrons (or holes), W is the channel width, and L is the channel length.

Table 1: An nMOS transistor's i - v behavior

Region	Conditions	Current (i_D)
Cutoff	$v_{GS} \leq V_{thn}$	0
Triode	$v_{GS} - V_{thn} \geq V_{DS}$	$\beta_n(v_{GS} - V_{thn} - V_{DS}/2) v_{DS}$
Saturation	$v_{DS} \geq (v_{GS} - V_{thn}) \geq 0$	$\frac{\beta_n}{2}(v_{GS} - V_{thn})^2 v_{DS}$

Table 2: A pMOS transistor's i - v behavior

Region	Conditions	Current (i_D)
Cutoff	$v_{GS} \geq V_{thp}$	0
Triode	$0 \leq v_{DS} \leq v_{GS} - V_{thp} $	$\beta_p(v_{GS} - V_{thp} - V_{DS}/2) v_{DS}$
Saturation	$ v_{DS} \geq v_{GS} - V_{thp} \geq 0$	$\frac{\beta_p}{2}(v_{GS} - V_{thp})^2 v_{DS}$

3.2 Noise Margin of an Inverter

The NMs of a logic gate represent the ranges of input voltages that produce valid output values. The NM for an inverter (Fig. 1) can be found by utilizing the output voltage (v_{out}) curve (voltage transfer curve/VTC), in response to a ramp voltage (v_{in}); a sample VTC is shown in Fig. 2. Normally, there are two points on the curve that have slope $\delta V_{out}/\delta V_{in} = -1$: For the input level $v_{in} = V_{IL}$, the output is V_{OH} , while $v_{in} = V_{IH}$ corresponds to the output V_{OL} . These two points correspond to the *low noise margin* (NM_{low}) and the *high noise margin* (NM_{high}). The two NMs and the *static noise margin* (SNM) are defined as:

$$NM_{low} = V_{IL} - V_{OL} \quad (2)$$

$$NM_{high} = V_{OH} - V_{IH} \quad (3)$$

$$SNM = \min(NM_{low}, NM_{high}) \quad (4)$$

Finding the NM (and consequently, the SNM) of a multi-input gate entails injecting an appropriate set of constant and ramp-inputs, and measuring the two inflection points of the VTCs. Any parameter fluctuation (for example, V_{th}) directly impacts the positions of the inflection points, and hence nm_{low} , nm_{high} , and the SNM [10].

3.3 Properties of Independent Random Variables

Suppose there are n independent random variables X_1, X_2, \dots, X_n , and their means are $\mu_1, \mu_2, \dots, \mu_n$, and the

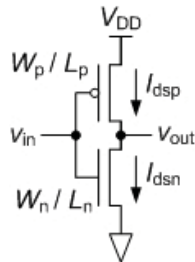


Fig. 1: The schematic of a CMOS inverter

variances are $\sigma_1^2, \sigma_2^2, \dots, \sigma_n^2$. Assume that a_i and C are real constants and that there exists a linear relationship:

$$Y = \sum_{i=1}^n a_i X_i + C, \quad (5)$$

Then the mean (μ) of Y is:

$$\mu_Y = \sum_{i=1}^n a_i \mu_i + C, \quad (6)$$

and the variance (σ^2) and the standard deviation (σ) of Y are:

$$\sigma_Y^2 = \sum_{i=1}^n a_i^2 \sigma_i^2, \quad \text{and} \quad \sigma_Y = \sqrt{\sum_{i=1}^n a_i^2 \sigma_i^2}. \quad (7)$$

4. Mathematical Model of Noise Margin

As mentioned earlier, the NMs (and the SNM) of an inverter are derived from the two inflection points on a VTC (Fig. 2) where the slopes are -1 . It is known that when the input $v_{in} > V_{thn}$, nMOST is in saturation and the pMOST is in triode/linear mode. The gate-source voltage for pMOST, $v_{GS} = v_{in} - V_{DD}$ and $v_{DS} = v_{out} - V_{DD}$; and for nMOST, $v_{GS} = v_{in}$ and $v_{GS} = v_{out}$. We consider the drain currents through nMOST and pMOST to be equal, i.e., $I_{dsn} = I_{dsp}$. (NMs are measured under no-load conditions, so the output current is zero.) By referring to Tables 1 and 2, we can write [11]:

$$\frac{\beta_n}{2}(v_{in} - V_{thn})^2 = \beta_p \left(v_{in} - V_{DD} - V_{thp} - \frac{v_{out} - V_{DD}}{2} \right) (v_{out} - V_{DD}) \quad (8)$$

We define $\beta_R = \beta_n/\beta_p$ and $V_{dp} = V_{DD} - V_{out}$, and substitute them in equation (8). After re-arrangement, we get:

$$\frac{1}{2} V_{dp}^2 - V_{dp}(V_{DD} - v_{in} - V_{thp}) + \frac{\beta_R}{2}(v_{in} - V_{thn})^2 = 0 \quad (9)$$

The solution for the *quadratic* equation (9) is:

$$V_{dp} = (V_{DD} - v_{in} - V_{thp}) \pm \sqrt{(V_{DD} - v_{in} - V_{thp})^2 - \beta_R(v_{in} - V_{thn})^2} \quad (10)$$

For pMOST to be in triode/linear mode, $V_{dp} = V_{DD} - v_{out} \leq V_{DD} - v_{in} - V_{thp}$, so we use the solution with the negative-sign. Substituting $V_{dp} = V_{DD} - v_{out}$ in equation (10) and by re-arranging, we get:

$$v_{out} = v_{in} + V_{thp} + \sqrt{(V_{DD} - v_{in} - V_{thp})^2 - \beta_R(v_{in} - V_{thn})^2} \quad (11)$$

In order to find V_{IL} , we differentiate v_{out} (from equation 11) with respect to v_{in} and equate it to -1 (i.e., $\delta v_{out}/\delta v_{in} = -1$). Then we solve for $v_{in} (= V_{IL})$:

$$V_{IL} = \frac{2\sqrt{\beta_R}(V_{DD} - V_{thn} + V_{thp})}{(\beta_R - 1)\sqrt{\beta_R + 3}} - \frac{(V_{DD} - \beta_R V_{thn} + V_{thp})}{\beta_R - 1} \quad (12)$$

By substituting V_{IL} in equation 11, we get:

$$V_{OH} = \frac{(\beta_R + 1)V_{IL} + V_{DD} - \beta_R V_{thn} - V_{thp}}{2} \quad (13)$$

When v_{in} is close to V_{IH} , the v_{DS} for the pMOST is large while nMOST's v_{DS} is small. Specifically, for pMOST, $v_{GS} = v_{in} - V_{DD}$ and $v_{DS} = v_{out} - V_{DD}$, and for nMOST, $v_{GS} = v_{in}$ and $v_{DS} = v_{out}$. Therefore, we can imply that pMOST is in saturation and nMOST is in triode mode. Therefore, we can equate the drain currents of both transistors, i.e., $I_{dsn} = I_{dsp}$ (refer to Tables 1 and 2):

$$\beta_n(v_{in} - V_{thn} - v_{out}/2) v_{out} = \frac{\beta_p}{2}(v_{in} - V_{DD} - V_{thp})^2 \quad (14)$$

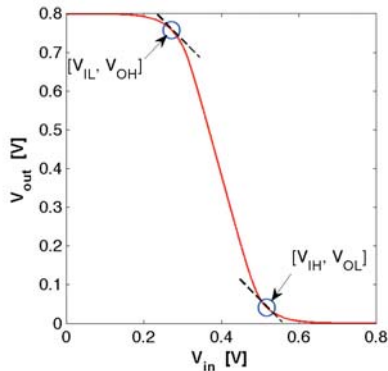


Fig. 2: Voltage transfer curve of an inverter ($L_{pmos} = L_{nmos} = 22$ nm; $W_{pmos} = 66$ nm; $W_{nmos} = 44$ nm; $V_{DD} = 0.8$ V).

After re-arranging equation 14, we obtain a quadratic equation for v_{out} :

$$\frac{1}{2}v_{out}^2 - (v_{in} - V_{thn})v_{out} + \frac{1}{2\beta_R}(v_{in} - V_{DD} - V_{thp})^2 = 0 \quad (15)$$

The solution of equation 15 yields:

$$v_{out} = (v_{in} - V_{thn}) \pm \sqrt{(v_{in} - V_{thn})^2 - \frac{(v_{in} - V_{DD} - V_{thp})^2}{\beta_R}} \quad (16)$$

With nMOST in linear region ($v_{out} < v_{in} - V_{thn}$), we retain the solution with negative sign. We can find V_{IH} by solving the equation $\delta v_{out}/\delta v_{in} = -1$ (as done earlier):

$$V_{IH} = \frac{2\beta_R(V_{DD} - V_{thn} + V_{thp})}{(\beta_R - 1)\sqrt{1 + 3\beta_R}} - \frac{(V_{DD} - \beta_R V_{thn} + V_{thp})}{\beta_R - 1} \quad (17)$$

By substituting V_{IH} in equation 16, we obtain:

$$V_{OL} = \frac{(\beta_R + 1)V_{IH} - V_{DD} - \beta_R V_{thn} - V_{thp}}{2\beta_R} \quad (18)$$

When we use the special condition $\beta_R = 1$ in equations 11 and 16, the derivations of V_{IL} , V_{OH} , V_{IH} , and V_{OL} are significantly simplified as shown below. (We are investigating the condition $\beta_R \neq 1$ and will disseminate our findings in the near future).

$$V_{IL} = 0.625V_{thn} + 0.375V_{thp} + 0.375V_{DD} \quad (19)$$

$$V_{OH} = 0.125V_{thn} - 0.125V_{thp} + 0.875V_{DD} \quad (20)$$

$$V_{IH} = 0.375V_{thn} + 0.625V_{thp} + 0.625V_{DD} \quad (21)$$

$$V_{OL} = -0.125V_{thn} + 0.125V_{thp} + 0.125V_{DD} \quad (22)$$

We can use equations 19–22 to find NM_{low} , NM_{high} , and the SNM:

$$NM_{low} = V_{IL} - V_{OL} = 0.75V_{thn} + 0.25V_{thp} + 0.25V_{DD} \quad (23)$$

$$NM_{high} = V_{OH} - V_{IH} = -0.25V_{thn} - 0.75V_{thp} + 0.25V_{DD} \quad (24)$$

$$SNM = \min(NM_{low}, NM_{high}) = -0.25V_{thn} - 0.75V_{thp} + 0.25V_{DD} \quad (25)$$

5. V_{th} -Variation-Aware Model of the Noise Margin

The V_{th} of a MOS transistor can be calculated using equations given in BSIM4v4.7 level 54 [12]. For the 22 nm node, nominal $V_{thn0} = 0.503$ V, and nominal $V_{thp0} = -0.461$ V [13]. The main factors affecting a MOS transistor's probabilistic behavior are (1) the type (nMOS or

pMOS), (2) the size (W and L), and (3) the input voltage [14]. The effect of random fluctuation of doping levels on the transistor's V_{th} can be estimated by [15]:

$$\sigma_{V_{th}} \approx 3.19 \times 10^{-8} \frac{t_{ox} N_{dep}^{0.4}}{\sqrt{L_{eff} W_{eff}}} \quad (26)$$

L_{eff} and W_{eff} are the effective channel length and width, respectively. N_{dep} is the channel doping concentration at depletion edge for zero body bias.

As NM_{low} , NM_{high} and SNM (as seen in equations 23–25) are all linearly dependent on V_{thn} and V_{thp} , we can apply equations 6 and 7 to determine the means ($\mu_{NM_{low}}$, $\mu_{NM_{high}}$, and μ_{SNM}) and the standard deviations ($\sigma_{NM_{low}}$, $\sigma_{NM_{high}}$, and σ_{SNM}) of the NMs and the SNM. (The underlying assumption is that V_{thn} and V_{thp} are independent variables).

$$\mu_{NM_{low}} = 0.75\mu_{V_{thn}} + 0.25\mu_{V_{thp}} + 0.25V_{DD} \quad (27)$$

$$\sigma_{NM_{low}} = 0.25\sqrt{9\sigma_{V_{thn}}^2 + \sigma_{V_{thp}}^2} \quad (28)$$

$$\mu_{NM_{high}} = -0.25\mu_{V_{thn}} - 0.75\mu_{V_{thp}} + 0.25V_{DD} \quad (29)$$

$$\sigma_{NM_{high}} = 0.25\sqrt{\sigma_{V_{thn}}^2 + 9\sigma_{V_{thp}}^2} \quad (30)$$

$$\mu_{SNM} = -0.25\mu_{V_{thn}} - 0.75\mu_{V_{thp}} + 0.25V_{DD} \quad (31)$$

$$\sigma_{SNM} = 0.25\sqrt{\sigma_{V_{thn}}^2 + 9\sigma_{V_{thp}}^2} \quad (32)$$

6. Use-Cases of Variation-Aware Noise Margin Models

As we mentioned earlier, the MC Spice simulations are commonly used for studying the effects of variations in circuits. Such simulations can be very time-consuming. The set of mathematical equations derived in the last section is a speedy alternative to the MC simulations. For example, to find the μ_{SNM} and σ_{SNM} for an inverter under *only-the- V_{th}* -variations, we ran 1000 simulations. The simulations took approximately 9.2 minutes to finish on a MacBook Pro computer (with 2.4 GHz Intel Core i7, 8 GB DDR3 1333-MHz RAM, and an SSD drive). A set of 1000 values of the μ_{SNM} and the σ_{SNM} were acquired using the proposed NM-equations (coded in Matlab) in a fraction of a second. For comparative purposes, the SNM-histograms from the Spice simulations and the equations are shown in Fig. 3. The equations give us $\mu_{SNM} = 0.2092$ V and $\sigma_{SNM} = 0.0288$ V, as compared to MC simulations that showed $\mu_{SNM} = 0.2006$ V and $\sigma_{SNM} = 0.0276$ V. One could argue that the accuracy of the NM-equations would be improved if higher order MOS-models for i_D are used, however, that would diminish the advantage of our proposed compact closed-form analytic expressions.

As a first use-case, we used equations 31 and 32 to find the relationship of β_R to an inverter's SNM. Fig. 4 shows the results. For $\beta_R > 1$, the μ_{SNM} exhibits a marginal increase. However, the SNM-range ($\mu_{SNM} \pm 3 \times \sigma_{SNM}$) first shrinks

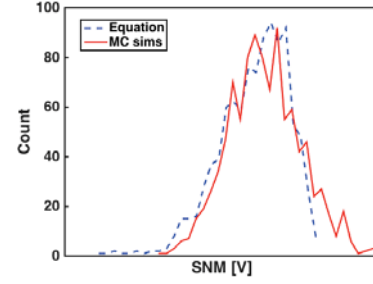


Fig. 3: Random variations in the SNM: histograms for 1000 Monte Carlo simulations and the equation-based results.

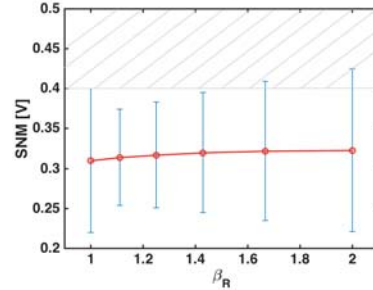


Fig. 4: Random variations in the SNM as a function of β_R (only W_{pmos} is varied) ($L_{pmos} = L_{nmos} = 22$ nm; $W_{nmos} = 44$ nm; $V_{DD} = 0.8$ V).

and then significantly widens as $\beta_R \rightarrow 2$. (The SNM values beyond the theoretical maximum of $0.5 \times V_{DD}$ are shaded in Fig. 4 and the following two figures).

The second use-case investigates the effect of channel length ($L_{mos} = L_{nmos} = L_{pmos}$) on SNM-variations (see Fig. 5). As L_{mos} is increased beyond the minimum ($L_{mos_min} = 22$ nm), we observe a large increase in μ_{SNM} . The range of SNM-variation drops as $L_{mos_min} \rightarrow 30$ nm.

The third use-case looks into the effect of V_{DD} on the SNM. In Fig. 6, we observe that μ_{SNM} is linearly related to V_{DD} (as expected). As V_{DD} drops below its nominal value of 0.8 V, the SNM-variations increase, thus aggravating the gate noise immunity; elevating the V_{DD} (> 0.8 V) shrinks the σ_{SNM} .

7. Conclusions

To study the effect of V_{th} variations on an inverter's NM, we can use the mathematical models in lieu of lengthy MC Spice simulations. The models due to their very nature are very time-efficient. This work has covered only the above- V_{th} operation of an inverter. Development of similar models for the sub- V_{th} operation is in progress. We are also planning to create above- and below- V_{th} NM-variation models of other common logic gates.

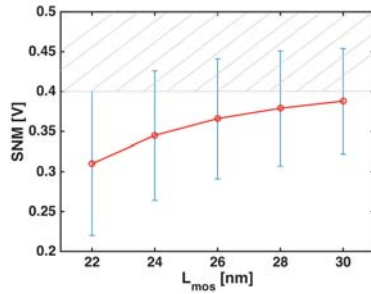


Fig. 5: Random variations in the SNM as a function of channel length ($L_{mos} = L_{pmos} = L_{nmos}$; $W_{pmos} = 66$ nm; $W_{nmos} = 44$ nm; $V_{DD} = 0.8$ V).

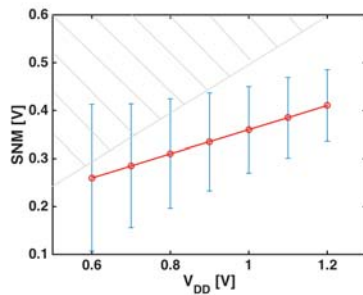


Fig. 6: Random variations in the SNM as a function of V_{DD} ($L_{pmos} = L_{nmos} = 22$ nm; $W_{pmos} = 66$ nm; $W_{nmos} = 44$ nm).

8. Acknowledgment

This work is partially supported by ADEC Award for Research Excellence (A²RE) 2015.

References

- [1] E. Nussbaum, E. A. Irland, and C. E. Young, "Statistical Analysis of Logic Circuit Performance in Digital Systems," *Proc. IRE*, vol. 49, no. 1, pp. 236–244, jan 1961.
- [2] C. F. Hill, "Definitions of noise margin in logic systems," *Mullard Tech. Communications*, no. 89, pp. 239–245, 1967.
- [3] J. Lohstroh, E. Seevinck, and J. de Groot, "Worst-case static noise margin criteria for logic circuits and their mathematical equivalence," *IEEE J. Solid-State Circuits*, vol. 18, no. 6, pp. 803–807, dec 1983.
- [4] J. Hauser, "Noise margin criteria for digital logic circuits," *IEEE Trans. Educ.*, vol. 36, no. 4, pp. 363–368, 1993.
- [5] E. Taylor and J. Fortes, "Device variability impact on logic gate failure rates," in *16th IEEE Int. Conf. Appl. Syst., Arch. Process. (ASAP 2005)*, 2005, pp. 247–253.
- [6] M. R. Choudhury and K. Mohanram, "Accurate and scalable reliability analysis of logic circuits," *2007 Des. Autom. Test Eur. Conf. Exhib.*, pp. 1–6, apr 2007.
- [7] S. Keller, S. S. Bhargav, C. Moore, and A. J. Martin, "Reliable Minimum Energy CMOS Circuit Design," in *2nd Eur. Work. C. Var.*, Grenoble, France, 2011, pp. 1–6.
- [8] J. L. Merino, S. A. Bota, R. Picos, and J. Segura, "Alternate characterization technique for static random-access memory static noise margin determination," *Int. J. Circuit Theory Appl.*, vol. 41, pp. 1085–1096, 2013.
- [9] A. Beg and A. Elchouemi, "Enhancing static noise margin while reducing power consumption," in *2013 IEEE 56th Int. Midwest Symp. Circuits Syst.* Columbus, OH, USA: IEEE, 2013, pp. 348–351.
- [10] A. Beg, "Automating the sizing of transistors in CMOS gates for low-power and high-noise margin operation," *Int. J. Circuit Theory Appl.*, pp. 1–14, 2014.
- [11] R. Jaeger and T. Blalock, *Microelectronic Circuit Design*. New York, NY, USA: McGraw-Hill, Inc., 2010.
- [12] "Berkeley Short-channel IGFET Model," 2013. [Online]. Available: http://www-device.eecs.berkeley.edu/bsim/?page=BSIM4_LR
- [13] "Predictive Technology Model," 2016. [Online]. Available: <http://ptm.asu.edu/>
- [14] P. Gupta, A. B. Kahng, P. Sharma, and D. Sylvester, "Gate-length biasing for runtime-leakage control," *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 25, no. 8, pp. 1475–1485, 2006.
- [15] A. Asenov, A. R. Brown, J. H. Davies, S. Kaya, and G. Slavcheva, "Simulation of intrinsic parameter fluctuations in decanometer and nanometer-scale MOSFETs," *IEEE Trans. Electron Devices*, vol. 50, no. 9, pp. 1837–1852, 2003.

Forces characterization from trajectory equations in virtual reality simulations for industrial applications

Marwene Kechiche¹, Mohamed-Amine Abidi¹, Patrick Baert¹, and Rosario Toscano¹

¹LTDS UMR 5513, Nationaly School of Engineering of Saint-Étienne, Saint-Étienne, Auvergne-Rhône-Alpes, France

Abstract—Recently, Virtual Reality (VR) is coupled with applications of Flow Production's (FP) compute. Several architectures and technics are developed to enable the communication between virtual reality and FP simulators. These latter are particularly important since they support the part of FP and VR applications. Nowadays, The FP simulations with virtual reality exist in two forms. On the one hand, the first form is the immersive visualization where the operator can only view the results computed through the simulation software. On the other hand, the 2nd form is presented under an interactive immersive visualization mode where the operator (a human factor) can act during the gallop of simulations. Some actions made by the operator require an assessment of muscular forces to predict musculoskeletal disorders (MSDs). Through the following paper, we will automate this step (compute muscular forces exerted) by putting aside any human intervention used in this production simulation, accurately in the simulation of forces. The forces will be computed from the equations of motion of the 3D object. In this stage, this method is only applicable with objects on movement.

Keywords: Virtual Reality, Flow Production, Industrial Simulation

1. Introduction

In an industrial environment, the workflow simulation enables to simulate the production within the parameters of input and output of each machine or group of machines. The workflow calculation is possible with a modelling of the technical environment and input-output of each machine. We can also perform many tests with variation of each parameter in order to optimize the production process. There are several workflow simulation software, most of these software do not give visual feedback. Consequently, only the experts are able to understand the results. In this context, virtual reality will be coupled with the FP software to have concrete visual results. The theoretical simulation workflow (software simulation) will be combined with a virtual presentation (virtual reality). Therefore, the observation of results will be viewed in real time. Either a simplified presentation of results or a detailed one can be used. This aspect depends on the operator's needs. With the integration of flow simulation and thanks to the development of virtual reality techniques,

a virtual flow simulation could be run side by side with the simulation flow (real one); in this stage, we don't parallelize the execution tasks. The flow simulation such coupled with a virtual reality simulation where the operator can intervene. Among the virtual reality simulations in which the operator intervention is required, there are simulations with a quantization of forces. The latter are used by the ergonomic evaluation applications and accurately in the force evaluation. To quantify the muscular forces in virtual reality, a physical engine and the haptic device are necessary [1]. Sometimes haptic devices cannot reach the estimated values of forces. There is a technique to replace the hardware limitations of the haptic devices; it consists into using the pseudo-haptic [2]. Our goal is to integrate the prediction of Musculo-Skeletal Disorders (MSD) in the flow simulation operation. This integration is done in three steps:

- * Implementation of the method of calculation
- * Automation of the calculation procedure
- * Get results and send them to the simulator

To achieve this, the exploitation of 3D moving objects with a dynamic behaviour is necessary to compute the forces for this simulation.

2. RELATED WORKS

Several flow simulation tools are developed. The given results by these developed tools are only understood by the experts in the flow simulation field. The technics of coupling FP with virtual reality and more precisely with a visual 3D rendering are developed to make understandable and interpretable results to everyone (those who at least have little knowledge in the field of flow simulation). The purpose of the use of virtual reality is to provide also a simple presentation of complex results. The research work of [3] shows how to transform complex results on 3D visualization system. Studies done by [4] show that the industrial models of virtual reality can be used as a reference for viewing during the modelling stage and the construction of the chain of production system. By exploiting this method, a virtual validation of the model is possible even before the implantation of the production chain. This work is done only for the 3D visualisation. However, the work of [5] allows the design of a virtual reality system coupled with a discrete event simulator. Generally, this tool is a means for control

and design of production processes. Therefore, a possibility of optimizing the production system is based on this tool. This approach allows the development of an abstract layer between the flow simulation and the concerned actors with a simple presentation of complex results (based on a 3D visualization) from this simulation. Abstract layers are not visible. The simulation tool encapsulates the simulation and the complex result in the visualization layer. The visualizing layer is changed to be immersive. It also allows the Integration of 3D visualization devices like headset @Oculus Rift or @CAVE. The immersion in the scene presentation enables the operator to visualize the results of calculations generated by the simulation software. This immersion is also an important factor to have a rapid and complete understanding of the set of results, which are generated. It gives the opportunity to suggest new ideas and solve problems. Moreover, the operator can intervene and interact with the scene and assess certain criteria. To perform this evaluation: the main process is stopped, local interactive simulation begins, and the results will be generated in a final report. Then they will be sent to the main module for interpretation. Finally the software uses the information and continues its execution.

In this paper, our contribution is based on the fact that we can improve the FP simulation on two parts: The first component is to integrate a computational method and automated simulations of a well-defined operation. This task (operation) cannot be simulated by flow simulation software (e.g. a transport simulation of a cart from a position to another with a calculation of muscular forces generated to perform this operation). The forces will be calculated using the estimated trajectories by the tool between two points (sampling of steps). The second component allows the integration of an interactive virtual reality simulation with an operator. Thanks to this component, we have the ability to compute an estimation of forces exerted on the mobile to move on the path personalized by the operator. So, the objective is to add in the flow simulation process an additional layer used to evaluate DMS generated by the pushing of cart. We should evaluate the applicable forces to transport it from one zone to another zone. Ideally, this assessment must be carried out automatically after semantic modelling workflows. This model interacts with a database created to calculate the correspondences between the postures and the exerted forces.

3. WORK CONTEXT

3.1 General context

As part of a workflow simulation in an automobile industry, flow simulation must incorporate these results into the simulation of cart manipulation. Indeed, this industrial uses carts to displace parts that are already assembled or not assembled. For unassembled parts, the operator can assemble parts on the cart. The carts are used to transport

parts with fixed or variable masses (if assembly is done on the cart). The aim is to treat this issue and integrate it into the virtual reality simulation in order to predict DMS. The carts are moved from position A_i to position A_f either directly or through other positions A_n (intermediate steps). The manipulation of carts and the planning of their trajectories become a problem to be solved. We will couple the flow simulation with VR including human character simulations. The simulated character is the muscular forces. These simulations take a multitude of possibilities. The most important two parameters are the path and the charge of the cart that have a direct influence on the muscular forces. Thus, the authorized forces are chosen from defined threshold. The authorized forces of carts manipulation will be stored in a database. A real time comparison was done between computed forces and authorized ones. For that, we have done acquisitions of forces and positions.

3.2 Process of coupling VR with flow simulation

3.2.1 Flow simulation tool

The flow simulation's tools are generally used to design and optimise industrial production systems. Software ARENA [10] SLAM [11] and APOLLO [12] are a good illustration of this type of tools. Processes used by flow simulation's tools are represented by tasks' sequences that produce and consumed resources. It also affects the characteristics of modelled and assembled parts. The execution time of each task is calculated. The basic tasks can be designed and modelled by the coloured Petri net model. There are various software allowing workflow's modelling. These software autonomously manage inputs and outputs of each task or group of tasks.

3.2.2 Semantic modelling

Several frameworks are developed for the semantic modelling. [6] The principle of High level Architecture (HLA) is to give a tool that interacts with other simulations, through interface layer called Run Time Infrastructure (RTI). Other architectures are based on XML and Web Services (WS) systems [7]. These systems use the WS architecture and interact with WS layer using Remote Procedure Call (RPC) mechanism. The messages are transported on XML messages. The SCIVE framework [8] allows the development of intelligent and interactive virtual environments. The connection between heterogeneous modules is possible. SCIVE supports also the maintainability, modularity, and interoperability of VR applications. The MASCARET framework [9] provides a logical connection between the virtual reality domain and the system engineering. It introduces an abstract layer between the components of VE and the concepts of the domain model. It allows building a semantic representation of the industrial system by using the SysML language.

MASCARET is implemented in various 3D engines, namely Ogre or Unity®.

4. EXPERIMENTAL STUDY

The aim is to generalize the treatment of all types of carts; by integrating its physical characteristics in the simulation. Physical characteristics are the mass of the empty cart and his coefficient of friction, which generates the frictional force constraining the movements of this later. Once these parameters are integrated into the tool. It must be capable to planning optimal paths. These paths are generated by the muscular forces applied by operators. The forces must respect the norms imposed by the standard of MSD. To study the carts, a series of measurement are performed to compare between the real and virtual computed forces. In this section, we will detail the protocols of acquisitions to compute the exerted forces.

4.1 Force Sensors

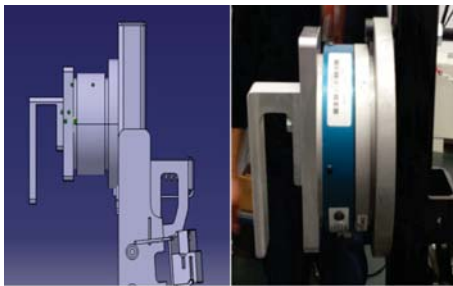


Fig. 1: Modelling and installing of platinum to attach force sensor

The force sensor can acquire the forces exerted by the operator. It must imperatively be installed on the point of application to acquire the exerted muscular forces. There are several types of forces sensors, which enable the acquisition of the exerted forces. The problem of the sensor choosing is usually associated with forces range. The forces exerted depending on the weight of the carriage and the charge of this latter. Another problem is related to the installation of the sensors on the carriage. It is necessary that the sensor be installed on the carriage in the impact point and far from the charge. In this context, we conducted a study on the carriage and we have designed platinum to install the sensor.

4.2 Motion capture

There are several kinds of the motion sensors. We chose to use the infrared motion sensors because we have an ®ART motions capture system with 10 cameras that cover an area of 7.5 m^2 . The order of precision on this system is the millimetre. After a calibration operation the position and orientation of the reference object is retained. The provided

data are the 3D positions of the object in the real world with a reference position and orientation parameters of this object in the same space. After calibration the retained data are the 3D positions and orientation of the cart on the space (with personalized marker for cart tracking). We can also adjust the frequency of acquisition (send or registration data). Note that the maximum frequency allowed by the system is 60Hz (60 values per second). Customizing markers is also possible. In our case, we can assign markers to any item after a calibration and integration procedure into the tracking system.

4.3 Protocols and experiences

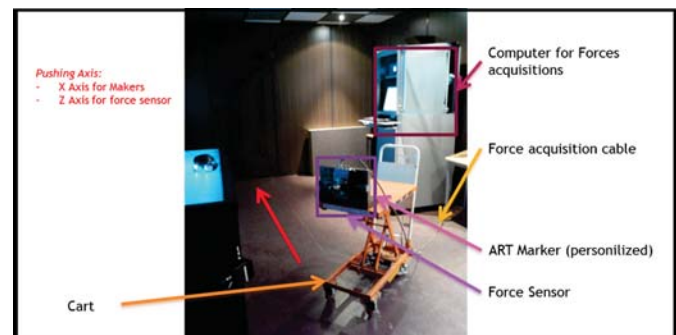


Fig. 2: Laboratory environment for forces and motion capture acquisitions.



Fig. 3: Another type of cart for tests

The primary objective of this experiment is to compare the forces computed by the equations of the trajectory with the forces exerted on the point of impact on the cart. A protocol of acquisition is defined, which allows us to study the forces. We will work on two fundamental criteria; the charge and the velocity of the cart. Other parameters are necessary for the DMS's prediction as the sensor position relative to the hand's position. In this paper we will ignore those settings. We are only interested in the variations of force. The tested parameters:

- * Distance: 1m, 2m, 3m
- * Speed: slow, medium, fast
- * Operator: Men / Women
- * Charge: 20kg, 40kg, 60kg
- * Acquisition frequency for position and orientations: 50 Hz, 25Hz, 10 Hz, 5 Hz
- * Acquisition frequency for the forces: 1000Hz, 100Hz, 50Hz, 25Hz, 10hz, 5hz

4.4 Data Collection

After installing the force sensor, the setting of acquisition protocols, and the initiation of markers for 3D tracking. The recorded data will be spread over two files. The first file contains information about the exerted muscular forces to move the cart. The second file retrieves the positions and orientations of each custom marker. In the first gallop of tests, we have customized a single marker to track only the positions of the carriage. In the remainder of this chapter the following frequencies are laid down: 100Hz, 50Hz, 25Hz, 10 Hz and 5 Hz. These frequencies are the retained frequencies for the various acquisitions.

5. Data Interpretation

5.1 Fundamental principle of dynamics

theoretical forces, or more precisely the forces computed from the equation of motion of the cart. To extract the motion equations we use these definitions:

$$\vec{V} = \frac{d\vec{X}}{dt}$$

$$\vec{A} = \frac{d\vec{V}}{dt}; \vec{A} = \frac{d^2\vec{X}}{dt^2}$$

For force compute we use fundamental principle of dynamics :

$$\sum \vec{F}_{ext} = m * \vec{A}$$

The acquired data represents the 3D coordinates of the cart. In a first time the calculation is performed based on the magnitude of the position and magnitude of forces. We have found that the main element is the component of displacement so we eliminated the calculation of the magnitude and we retained only the displacement component. (i.e: displacement component presents the displacement axis).

The first two curves that show the position data are not too noisy. With the precision of our tracking system a smooth curve of position is obtained. After the first derivative of the curve positions we got the curve of velocity that is slightly noisy. During the derivation of the velocity curve (for the acceleration curve) we had a totally noisy curve and the acceleration values are invaluable from this curve.

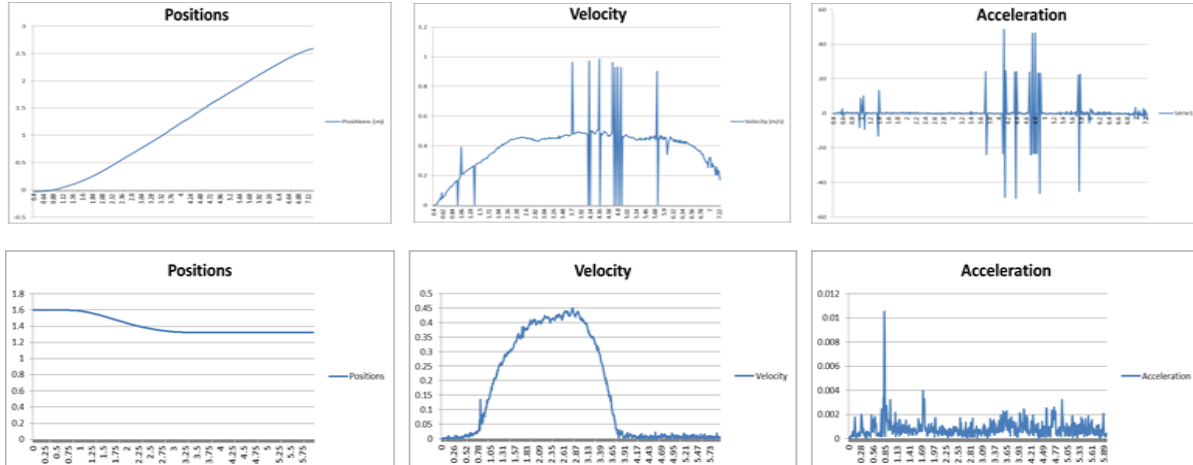


Fig. 4: curves of positions, velocities, and accelerations

Once the data are available, an interpretation of these latter is necessary. The trajectory modelling and kinematic parameters of the carriage are required. We will compute the instant velocity and acceleration. With this calculation we can deduce the equations of motion, the trajectory, and predict the performed forces on the trajectory. These parameters will be used directly in the calculation of the

It is possible to see the difference between the two curves. The curve forces theoretically computed (from position values) is too noisy and has no overall look. This curve does not follow the curve acquired by the sensor. Noise reduction is required. To reduce the noise frequency variation is made, this variation greatly reduces noise but it did not allow the complete removal of the latter.

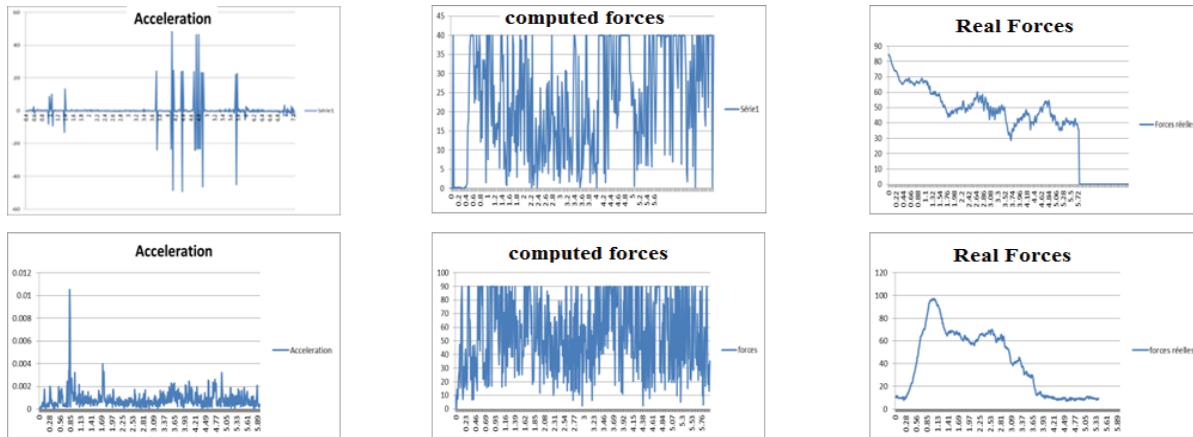


Fig. 5: Comparison between forces computed from accelerations and real forces

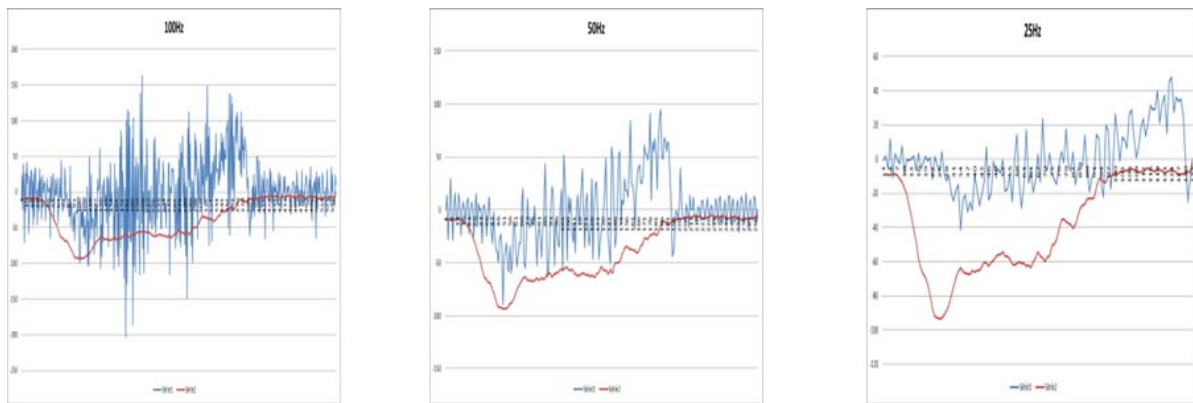


Fig. 6: changing of frequencies

5.2 Polynomial approximation

To smooth the curve we will use the principle of a polynomial approximation that can filter and smooth the data curve. Our goal is to define a trajectory equation based on the following polynomial form:

$$at^2 + bt + c = y$$

this form is a second order polynomial. This polynomial is easily differentiable and more stable in the interpretation. Our goal is to calculate the coefficients of the polynomial from the series of acquisition. These coefficients represent: acceleration, velocity, and position. The approximation method is based on the principle of Means square.

5.3 result

When using raw data or merging data to calculate the acceleration by the second derivative of the value of the position we have a noisy results that are non-exploitable for the estimation of applied forces to the cart. A small improvement is observed on the curve when we lowered the acquisition frequency in other words we have lowered the noise levels. Finally and in ideal conditions we can have an accurate estimate of the force exerted on the cart. This estimate is obtained for testing a linear movement in a single direction, with forces exerted only on one axis. The shift obtained on the force curve shows the experimental value of the frictional forces.

6. USING RESULTS IN VR APPLICATIONS

In the previous sections, we cited coupling techniques, the method of interacting with the virtual environments, and the

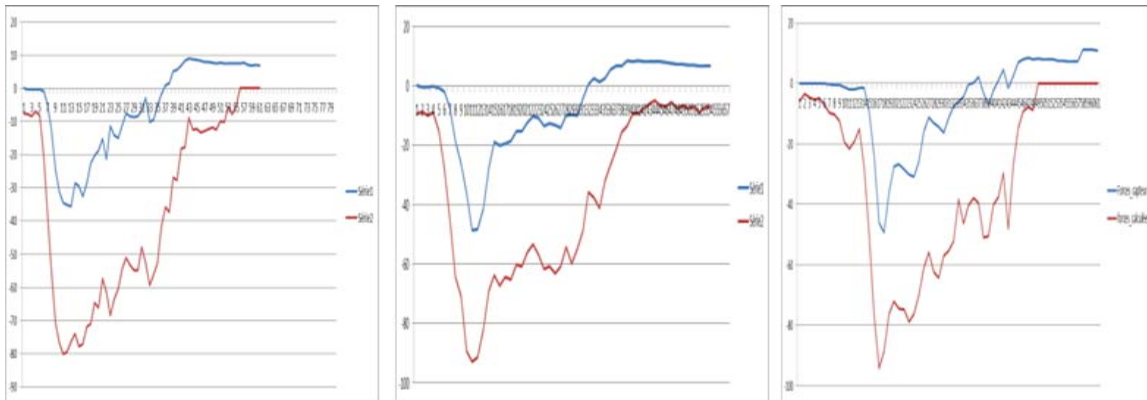


Fig. 7: Results obtained by polynomial approximation

method of computing forces based on motion capture. In this section, we describe the exploitation of these methods in a virtual reality application. This application combines between these modules. Figure 8 presents the architecture of this application. The obtained result presents an estimation of muscular forces exerted in the industrial context. This estimation can be computed automatically by trying several combinations of trajectories. The start and the end positions are fixed and within the constraints imposed on the simulator offer different trajectories. The module of creation of Virtual Environment designed the zones of the cart circulation. A first ste of tests is done with this path. The path decomposition Virtual Reality Module Simulation (VRS) allows the compute of muscular forces. During the simulation, this VRS sends in real time the information to the 3D visualization module. That's allows the visualization of path personalizing operation. If the forces measured do not generate the MSD, an optimization is performed. The system retains the paths and varied velocities. Once speeds are retained, the system performs other tests to check the maximum acceptable charge.

Table 1: forces computing steps.

Steps	Jobs	Results
Step1	Trying different paths with constant speed	Retain the paths and the velocity that do not generate MSD
Step2	Trying different velocities	Retain the threshold of velocity
Step3	Changing charges On different zones	Retain the maximum accepted charges

After this estimation the characteristics of the transaction will be retained and sent to flow simulation software to continue its simulations. This simulation is not parallelized so far but in the future a parallelization of tasks can be

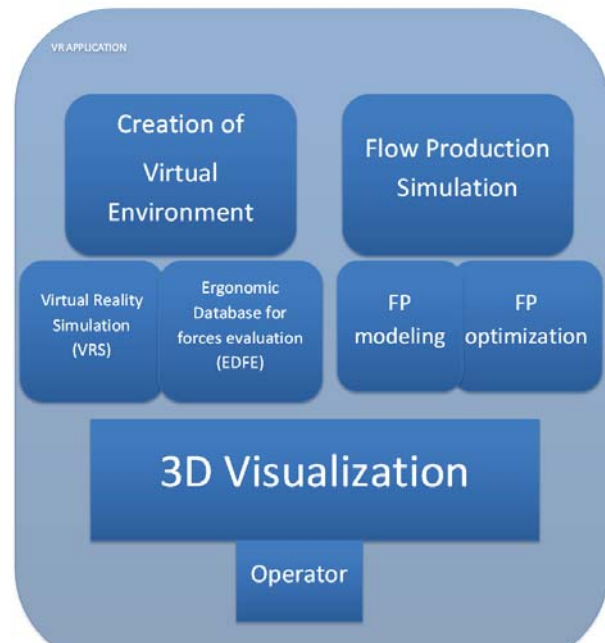


Fig. 8: Proposed architecture for VR application

envisaged. In another application, a flow simulation is coupled with a training environment where the operator must perform the exercise. This is the case of the interactive and immersive virtual reality. In this case the compute of interpretation is performed in real time with the motion capture and the results will be validated with ergonomists who work directly with the operator interface. The advantage of use the calculation of the previous section in this case is manifest in the ability to prevent the operator. A training

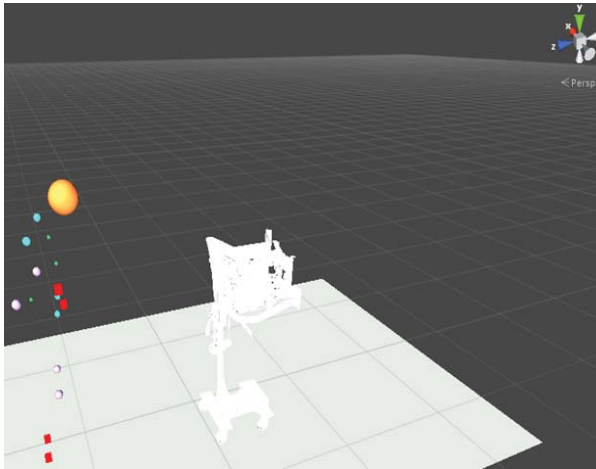


Fig. 9: Reproduce of the simulation in virtual environment

phase allows the prediction of the trajectory at time $T + 1$ as a function of time $T-1$ and T . In this case the application of the principle of "Dead reckoning" for the prediction and for the warning of a possibility of DMS if the operator keeps its behaviour.

7. Conclusions

In this document we presented the FP simulators of production process in industrial context and their complex results. The research has been conducted in order to develop communication, or more precisely, the coupling between the flow simulator and virtual reality. There are techniques that allow coupling the results with the 3D visualization of results and the interaction with the virtual environment. In this document we detailed the evaluation aspect and predicted the forces exerted on a cart. These forces will be used, later, by an ergonomic assessment software. The flow simulator will take into consideration the forces in its workflow optimization. In this document we tried to properly estimate the muscular exerted forces needed for the integration of ergonomic assessment in the FP process. This estimation can be defined either automatically with a proposed trajectory or manually with the direct intervention of the operator for a virtual reality simulation.

8. ACKNOWLEDGMENTS

At the end of this paper I would like to thank all those who participated in this work one way or another to perform the experiments and get the results. Special thanks to:
 IFSTAR (LYON)
 Coralie BAUVENT
 Mariem AMRI
 Leila HNIA

References

- [1] M. Kechiche, M-A. Abidi, P. Baert, R. Toscano, *Using Haptic Forces Feedback for Immersive and Interactive Simulation in Industrial Context*. Augmented and Virtual Reality. Volume 9254 of the series Lecture Notes in Computer Science pp 301-313 Lecce
- [2] A. Lecuyer, S. Coquillart, A. Kheddar, P. Richard, P. Coiffet, *Pseudo-haptic feedback: can isometric input devices simulate force feedback?*. Virtual Reality, 2000. Proceedings. IEEE. 18 Mar 2000-22 Mar 2000
- [3] W. Dangelmaier, M. Fischer, J. Gausemeier, M. Grafe, C. Matysczok, B. Mueck, *Virtual and augmented reality support for discrete manufacturing system simulation*, Computers in Industry 56 (4) (2005).
- [4] E. Lindskog, J. Berglund, J. Vallhagen, B. Johansson, *Visualization support for virtual redesign of manufacturing systems*, Forty Sixth CIRP Conference on Manufacturing Systems 7 (2013).
- [5] M-A. Abidi, P. Chevalier, B. Lyonnet, M. Kechiche, P. Baert, R. Toscano, *How to Create a New Generation of Industrial Processes Simulation by Coupling the Simulation Tools with VR Platforms*, 28th International Conference on Computer Applications in Industry and Engineering (CAINE-2015)
- [6] K. Frederick, W. Richard, D. Judith, *Creating computer simulation systems: an introduction to the high level architecture*, Prentice Hall PTR Upper Saddle River.
- [7] S. Chandrasekaran, G. Silver, J. Miller, J. Cardoso, A. Sheth, *Xml-based modeling and simulation: web service technologies and their synergy with simulation*, Proceedings of the 34th conference on Winter simulation 12(2002).
- [8] M. Latoschik, C. Frohlich, *Semantic reaction for intelligent virtual environments*, Virtual Reality Conference, 2007. VR '07. IEEE (2007) 305-306
- [9] P. Chevaillier, T.-H. Trinh, M. Barange, P. De Loor, F. Devillers, J. Soler, R. Querrec, *Semantic modeling of virtual environments using mascaret*, Proceedings of SEARIS'12 12 (2012).
- [10] C.D. Pegden, R.E. Shannon, R.P. Sadowski, *Introduction to simulation using SIMAN*, McGraw Hill, New York, NY, 1990
- [11] [Pritsker 1986] A.A.B. Pritsker, *Introduction to simulation and SLAM II*, Halsted Press, New York, NY, 3rd edition, 1986
- [12] G. Habchi, C. Berchet, *A Model for Manufacturing Systems Simulation with a Control Dimension*, SIMPRA, Editions Elsevier, Pays-Bas, vol. 11, number 1, 2003, pp. 21-44.

Advil: A Visualization Language for Dynamic Visualization

T. Cerrah¹, and H.-P. Bischof^{1,2}

¹Department of Computer Science, Rochester Institute of Technology Rochester, NY, USA

²Center for Computational Relativity and Gravitation, Rochester Institute of Technology Rochester, NY, USA

Abstract—*Visualization of scientific data can help to analyze and explore the data in ways, which cannot be achieved with analytical methods. Most visualization programs are typically implemented using a data flow approach. A visualization programs consist of a set of components connected via directed graph, and the data flows through the program and this process creates images, which are later assembled into a movie. We often need to change the properties of the components dynamically during the visualization process in order to create the best possible movie. We propose a method to use the visualization program as an interpreter for a dynamic visualization program, which allows making these changes without rewriting the visualization program. This method allows us to focus on a particular visual after the visualization program has been written. This method allows us to create significantly more interesting visualization movies.*

Keywords: Visualization Languages, Data Flow, Dynamic Visualizations

1. Introduction

A visualization of data can produce one image or a movie, meaning many images. This paper concerns only the type of visualizations, which generates more than one image for a data set. Examples for these kind of data sets are simulations of Black Hole mergers [2], or measurements of fracture strains[7] etc. These data sets have one common property: one value changes during the simulation or the experiment, but not necessarily in a linear fashion. In a Black Hole merger simulation this property is time, which moves forward in a linear fashion; in a fracture strain experiment/simulation it could be force, or gauge which changes cannot be described with a linear function.

Most visualization environments are using a data flow framework, which was first described by Foulser[4]. A visualization program can be modified and executed as often as needed creating individual images, which are then mounted to a movie.

In principle, a visualization program consists of components, which are connected via a directed graph. A component has n input channels and k output channels, which are connected, which create the directed graph. The same graph can also be achieved by calling methods in a particular order. A component's functionality can typically be fine-tuned using component-specific arguments. These

arguments specifying individual properties, like line width, the color or transparency of a visualized object, position of the viewpoint, look-at position etc. In most visualization systems these properties cannot be modified during the execution of the visualization process, or it is very difficult to do so.

We typically create many versions of a movie from the same data set because we are not satisfied with the final result. For example, the camera movement starts to late/early and is to fast/slow, or the camera speed for two different movements needs to be identical. We decide to change when and by how much an object becomes translucent often, because the desired effect has not been achieved yet. Light positions needed to be changed dynamically because the shadow of an object hides what should be visible. Using this technique allows us to experiment with much different visualization until we find the best fitting one.

This paper describes a visualization environment where all these modifications can be made without rewriting the visualization program. This is not a new idea. The \LaTeX typesetting framework follows a similar idea for typesetting text. The text is written in a document including formatting ideas like *new paragraph*, *this is a bulleted lists*, *heading*, *sub-heading* etc.

The paper describes a use case in detail followed by a discussion of related work, the Spiegel visualization framework[5], and A Visualization Language for dynamic visualizations (*advil*).

2. Dynamic Visualization

Dynamic visualizations allow changing the properties of components during the visualization process. An example will help to illustrate this. Let's assume we visualize the simulation of a black hole's merger. One purpose of this visualization is to show the trajectory of the black hole's position over time. The number of past positions, also called the length of the trajectory, must decrease over time otherwise the trajectories will be on top of each other. The left part of Figure 1 shows the simulation at the beginning stage and the right part shows the state of the simulation close to the merger. The decreasing length of the trajectory cannot be described with a linear function, because the distance between black hole's during the merger is not linear.

Another example would be to move the viewpoint from position a to position b and then to position c . The viewpoint

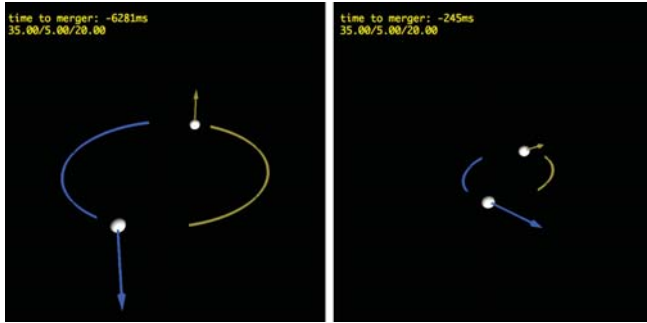


Fig. 1: Trajectory Length over Time.

position can be moved in a linear way between the anchor points, or on a spline curve. Moving the viewpoint along a linear function will cause a kink, if a , b , and c are not linearly aligned; moving them along a spline curve forces one to have less control over the exact movement because of the nature of splines[8]. We will later explain how these kind of problems could be addressed using Spiegel and *advl*.

3. Related Work

Many wonderful visualization systems have been developed. Most visualization systems focus on the quality of the images, being able to read a wide range of data formats, and particular visualization algorithms for very specific visualization challenges. We selected two highly used visualization systems here and a novel approach by Forbes.

The main focus of Spiegel is to create movies. Therefore it was logical to look into the production problems of cinematography[9]. We realized the world of cinematography and visualizations is so different that their approaches are not applicable for your problem.

3.1 behaviorism: a framework for dynamic data visualization

Forbes[1] created a framework which provides flexibility for visualizations of dynamic data. The framework is based on three connected graphs, and operators for each graph. The scene graph is used for rendering, a data graph is used for accessing the data, and a time graph to connect the two.

The framework provides a range of flexibility and aims to help visualization developer to focus on the visualization and not on the behavior. The paper provides little about how the behavior is controlled, it is more focused on the implementation and the design. Therefore it is very difficult to tell how it is used, but best to our understanding the behavior modifications are very limited.

3.2 yt

Yt[3] is a cross-code visualization tool that works with a number of astrophysical simulation codes, and is therefore very well suited for astrophysical visualizations. Yt is one

out of four visualization systems supported by the Blue Waters Sustain Petascale Computing Center[10]. Python was the language of choice for the developer. Only the parts, which require high performance computing, have been implemented in C. Yt supports around 20 different data types, numerous algorithms to examine, and visualize the data, and MPI support for distributed visualization programs.

The code snippet in Listing 1 shows how a camera is created, rotated, and moved to a position. Lines 1-12 create the camera object. A camera rotation is shown in line 15, and a movement to a position in line 16. The first argument of *move_to* is the final position, and the second argument defines in how many steps the final position will be reached.

This example gives a glimpse of how yt is used. It is fair to say that dynamic programming is extremely difficult to achieve in yt. It can be done, but requires a rewrite of the visualization program. This is extremely time consuming and therefore not advisable.

Listing 1: Creating and moving a camera in yt.

```

1 center = [0, 0, 0]
2 normalV = [1, 1, 1]
3 width = 1.0
4 xPixels = 512
5 yPixels = 512
6 transF = yt.ColorTransferFunction(...)
7
8 northV = [0., 0., 1.]
9
10 cam = ds.camera(c, normalV, width,
11                (xPixels, yPixels),
12                transF, northV=northV)
13
14 theta=0.2
15 cam.rotation(theta)
16 cam.move_to([0, 1, 2], 10)
```

3.3 ParaView

ParaView[6] is a visualization tool supporting C++, Python and JavaScript. ParaView is one out of four supported visualization systems supported by the Blue Waters Sustain Petascale Computing Center[10]. The user guide for ParaView is 230 pages long. ParaView, differently to yt is a more general visualization tool. ParaView supports around 11 most commonly used data formats. A graphical editor can be used to create a visualization program, and it is also possible to script a visualization program.

A simple Python paraView script is shown in Listing 2 to give a glimpse of how it is used. Lines 1-4 define a sphere; Line 7 creates a renderer, which is connected with the view in line 9. Lines 11-13 shrink the sphere by a factor of 2 and are rendered in line 15.

Listing 2: ParaView Code Snippet.

```

1 >>> from paraview.simple import *
2 >>> sphereObject = Sphere ()
3 >>> sphereInstance.Radius = 1.0
4 >>> sphereInstance.Center [1] = 2.0
5
6 >>> sphereDisplay = Show(sphereInstance)
7 >>> view = Render ()
8
9 >>> Render (view)
10
11 >>> shrinkInstance =
12     Shrink (Input=sphereInstance ,
13           ShrinkFactor = 2.0)
14 >>> shrinkDispl
15 >>> Render ()

```

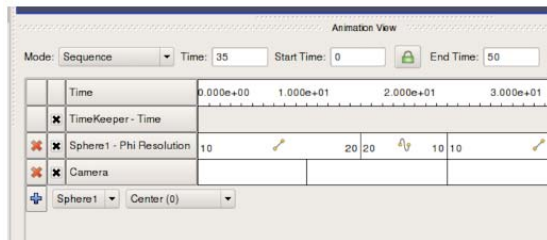


Fig. 2: ParaView Animation View [11].

ParaView supports the creation of animations using key frames. This is described in ParaView[6] guide on the pages 119-123. The key frames can be defined using the *Animation View*. The animation view is shown in Figure 2.

Only very simple animations can be created using the Animation view, like modifying a scalar, creating and modifying a camera path. Anything sophisticated cannot be done within this framework.

4. Spiegel

Spiegel[5] is a visualization framework written in Java. A program in Spiegel is a directed graph connecting individual components. An interpreter executes the program. A Spiegel program is most often implemented by using a graphical editor, but can also be implemented using a text editor. The language is type safe; this means only connections of connectors of the same type can be made. The graphical editor uses reflection[13] to ensure this property. The Spiegel language is simple, but it allows creating functions to create more complex components using simpler components or functions.

Figure 3 shows a very simple, but complete, *Hello World* program. The data flows from the component named *Stars*

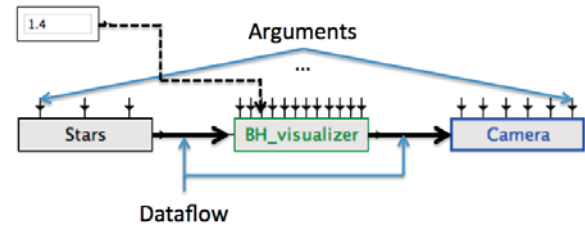


Fig. 3: Hello World.

to the *BH_visualizer* component, and finally to the *Camera*. The size of the black hole is set via an argument to be 1.4.

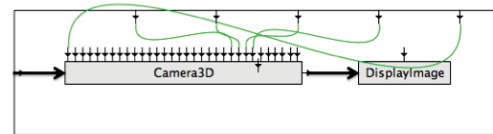


Fig. 4: A Graphical Representation of a Function inSpiegel.

Figure 4 shows a representation of the camera function used in Figure 3. As can be seen, some arguments from the components inside the function are not accessible within the *Camera* component. This encapsulating can be done with input and output channels. Encapsulation and using functions reduces the complexity of a creating a program with a graphical editor significantly.

A selection of the available Spiegel components and their categories are:

- Visuals: for visualizing Black Holes, Stars, Gas, Mesh
- Extractors: for extracting data from different data formats and origins (disk/network)
- Filters: For finding intersections, extracting positions
- Inputs: for data types like double, int, point
- Light: for point light, ambiguous light
- Util: for advl, orbiter, linear value supplier

5. Dynamic Visualization and Spiegel

Simulations or experiments, which produce the data sets, have one common property. A value changes, which drives the simulation or the experiment. In most simulations this variable is a scalar, like time, temperature, pressure, or light intensity. This value is typically used to determine which part of the data set will be used for an individual image, and which data set will be used for the next image. We will use this property to drive the programming of the dynamic visualization.

We will explain this with the help of Figure 5. The goal is to move the view point at the times 0, 3, 7, 8, and 9 to the positions outlined in Figure 5. The points are called anchor points. The positions of the view point locations in between

the known positions can be interpolated. The dotted line represents a linear and the dashed line represents a spline interpolation.

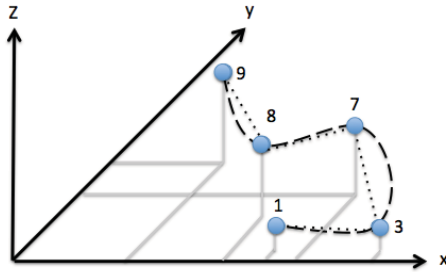


Fig. 5: Linear vs. Spline Interpolation.

6. Spiegel and Advl

Figure 6 illustrates how the advl program is used within the Spiegel framework. An interpreter component, *output Interpolator*, reads the program and provides for every stream connection. This connection, in this example *location of the camera*, is connected to one or more components. The input for the *Interpolator* is provided by the *Time* component. This component produces a series of values beginning at $t_{\text{beginning}}$ to t_{end} with intervals of t_{δ} and therefore the camera moves along the defined path.

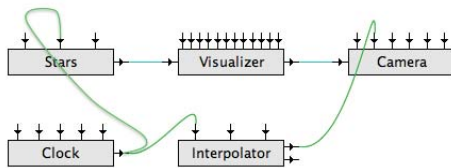


Fig. 6: Interpolating between Anchorpoints.

One component in the Spiegel framework can be programmed in advl to achieve the interpolation between points and send out the values between the anchor points. Figure 6 depicts the graphical version of the program. The component *Interpreter* interprets the advl program seen in Listing 3. Lines 1-4 define two constant values. Lines 6-16 define the output stream *viewP*. Line 8 defines the type of the output stream. The type of interpolator used to calculate the position between two anchor points is a TCP spline interpolator is defined in line 9. Lines 12-16 define the output stream *simulationTime*. The interpolator for this output stream is

linear as defined in line 15. Lines 18-25 define the anchor points and the position of the viewpoints.

The *Clock* component is programmed to send out values from 0 to 9 with a δ of 0.1. The *simulationTime* component output of the *Interpolator* will send out exactly the same values through the *simulationTime* output, because the line 24-25 specifies the *simulationTime* equal to the *Clock* time. This means the camera moves along on a TCB spline[12] path in 0.1 time units. The data set is accessed for the same time units. This visualization program will create 90 frames.

Listing 3: Interpolating between Anchor Points.

```

1  var {
2      const double startClock = 0;
3      const double endClock = 1;
4  }
5
6  stream {
7      viewP {
8          type vector;
9          interpolator TCB;
10     }
11
12     simulationTime {
13         type double;
14         interpolator Linear;
15     } }
16
17
18 startClock { viewP = (3, 1, 1);
19 }
20 3.0 { viewP = (5, 1, 1); }
21 7.0 { viewP = (3.5, 2, 4); }
22 8.0 { viewP = (2.5, 1.5, 3); }
23 end { viewP = (1, 1, 5);
24     simulationTime =
25     endClock; }
```

6.1 Spiegel and Slow-Motion

We now would like to change this program to achieve a different kind of visualization. First, changing *TCBhpEdMSV06* to *Linear* will move camera along a linear path.

The data between simulation time, 1 and 2, might be very interesting and therefore we would like to show this part in slow motion. This means we must generate more visuals for this time period versus the other time periods. One way to achieve this is to move the clock time faster forward than the simulation time. As a result, more images will be generated and therefore a slow-motion effect will be created.

The modified advl code is shown in listing 4. We added a few constants to make code easier to modify. Line 12 was changed to move the camera on a linear path. Lines 23-26 will produce 40 clock ticks. This means instead of 10, 30 images will be generated. Lines 30-33 are needed so such

the of the visualization produces for every 0.1 time unit one image.

Listing 4: Slow Motion.

```

1  var {
2    const double startClock = 0;
3    const double endClock   = 9;
4    const double slowStart  = 1;
5    const double slowDelta  = 1;
6    const double clockDelta = 5;
7  }
8
9  stream {
10   viewP {
11     type vector;
12     interpolator Linear;
13   }
14
15   stream simulationTime {
17     type double;
18     interpolator Linear;
19   } }
20
21 startClock { viewP = (3, 1, 1)
22             }
23 slowStart {
24   simulationTime = slowStartÖ }
25 slowStart + clockDelta {
26   simulationTime += slowDelta; }
27 3.0 { viewP = (5, 1, 1);      }
28 7.0 { viewP = (3.5, 2, 4);   }
29 8.0 { viewP = (2.5, 1.5, 3); }
30 end + clockDelta {
31   viewP = (1, 1, 5)
32   simulationTime = endClock +
33     slowDelta;      }
```

7. Advanced Advl Program

A more complicated example is show in Lisiting 5. Lines 1-8 define variables; lines 9-14 create the camera position stream. A function, *moveCam* , is defined in 15-21. The anchor point, line 23-27, defines the variables *x* and *y*. The scope of these variables is this block. The lines 23-34 move the camera position to a given point and back. The value of the built in variable *time* is equal to *deltaTime* after line 27 has been interpreted. It is worth to point out that the speed of camera is identical for both movements. A modification of *deltaTime* would change the speed for the camera movements for both segments. Lines 35-39 moves the camera in the time along a varying *x* value.

Listing 5: Advl and Spiegel in Concert.

```

1  var {
2    const double deltaTime = 42;
3    const double z = 2;
4    const double radiusC = 10.0;
5    const double middleC = (1, 2, 2);
6    double midX = 10.0;
7    double midY = 20.0;
8  }
9  stream {
10   cameraPos {
11     type point;
12     interpolator TCB;
13   }
14 }
15 point moveCam(double x) {
16   double r = radiusC ^ 2;
17   double xComp = (x - midX) ^ 2;
18   double y = (sqrt(r-xComp))+midY;
19
20   return (x, y, z);
21 }
22
23 0.0 {
24   double x = 90.0;
25   double y = 20.0;
26   cameraPos = (x, y, z);
27 }
28
29 0 + deltaTime {
30   cameraPos = (x + delta , y+delta , z)
31 }
32 time + deltaTime {
33   cameraPos = (x y, z)
34 }
35 time + deltaTime {
36   for (i = 1 : 20) {
37     cameraPos = moveCam((x - 10) + i);
38   }
39 }
```

Changing of the variable *deltaTime* (line 2) would change the speed of the camera movement, but not the path of the camera movment.

8. Conclusion

Advl is a language, which allows controlling the behavior of visualization systems effortlessly. It would be relatively easy to add this framework to yt, or ParaView, which would allow developers to use and control very sophisticated visualization with the same language. Using small, domain specific languages allows for an ease of use which can not be achieved general purpose languages.

9. Future Work

Future work will include to add the functionality to yt, and ParaView. Spiegel and advl do not support much user interaction during the execution of the visualization program. Domain specific programming languages drive the complete visualization process. It might be useful to allow user interaction during the visualization process to change the visualization process if *interesting* things can be seen. It might be useful to add an AI component, which can direct the visualization process to direct the visualization process instead of advl.

10. Acknowledgements

The authors would like to thank all members of *The Center for Computational Relativity and Gravitation at RIT*. Their visualization needs drove much of the development of advl.

11. Appendix: Advl

This section describes the syntax of *advl*.

Listing 6: advl Syntax.

```

prog: vars? streams func* anchor+

vars: 'var' '{' varDecl* '}'

streams: 'stream' '{' stream+ '}'

varDecl: 'const'? basic ID
        ( '=' expr )? ';'

stream: ID '{' 'type' basic ';'
        ('interpolator' interp ';')? '}'

func: type ID '(' params? ')' block

block: '{' stmt* ('return' expr ';')? '}'

params: basic ID (',' basic ID)*

anchor: DOUBLE ('+' ID)? block

stmt: ID ('='|'+='|'-=') expr ';'
      | ID ('++'|'--') ';'
      | ID '(' args? ')' ';'
      | varDecl
      | ifBlock elifBlock* elseBlock?
      | 'while' '(' expr ')' block
      | 'for' '(' ID '=' expr ':' expr ')'
        block

ifBlock: 'if' '(' expr ')' block

```

```
elifBlock: 'else if' '(' expr ')' block
```

```
elseBlock: 'else' block
```

```

expr: ID '(' args? ')'
      | expr '==' expr
      | expr '!=' expr
      | expr '<=' expr
      | expr '>=' expr
      | expr '>' expr
      | expr '<' expr
      | expr '&&' expr
      | expr '||' expr
      | expr '*' expr
      | expr '/' expr
      | expr '+' expr
      | expr '-' expr
      | expr '%' expr
      | expr '^' expr
      | '-' expr
      | 'sin' expr
      | 'cos' expr
      | 'tan' expr
      | 'sqrt' expr
      | 'abs' expr
      | '(' expr ')'
      | bool
      | INT
      | DOUBLE
      | point
      | bool

```

```
args: expr (',' expr)*
```

```
interp: 'Linear' | 'TCB'
```

```

type: 'int' | 'double' |
      'point' | 'bool' | 'void'

```

```

basic: 'int' | 'double' |
       'point' | 'bool'

```

```

point: '(' doubleValue ','
        doubleValue ',' doubleValue ')'

```

```
bool: 'True' | 'False'
```

```
doubleValue: DOUBLE | id
```

```
ID: ID_LETTER (ID_LETTER | DIGIT)*
```

```
INT: '-'? ('0' | NZD DIGIT*)
```

```

DOUBLE: '-'? ('0' | NZD DIGIT*)?
        DOT DIGIT*

```

```
ID_LETTER: 'a'..'z'|'A'..'Z'
```

```
DIGIT: '0'..'9'
```

```
NZD: '1'..'9'
```

COMMA: ', '

DOT: ', .'

LINE_COMMENT: '// '.*? '\n'

COMMENT: '/*' .*? '*/'

WS: [\t\r\n]+

References

- [1] A., Forbes, T., Hoellerer, and G. Legrady, "behaviorism: a framework for dynamic data visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, issue 6, October, 2010, pages 1164-1171.
- [2] C., Lousto, and J. Healy, "Flip-Flopping Binary Black Holes," *Phys. Rev. Lett.*, vol. 114, April, 2015.
- [3] M. J. Turk, et al., "yt: A Multi-code Analysis Toolkit for Astrophysical Simulation Data," *arXiv*, vol. 192, Jan. 2011.
- [4] D., Foulser, "IRIS Explorer: a framework for investigation," *ACM SIGGRAPH Computer Graphics - Special focus: modular visualization environments (MVEs)*, vol. 29, Issue 2, pp. 13-16, Nov. 1995.
- [5] H.-P. Bischof, E. Dale, and T. Peterson, "Spiegel - a visualization framework for large and small scale systems," in *Proc. MSV'06*, 2006, paper, p. 199-205.
- [6] C. Quammen, "Scientific Data Analysis and Visualization with Python, VTK, and ParaView," in *Proc. Python Conference'15*, 2015, paper, p. 32-39.
- [7] G. Yuang, B. Yan, and H. Zhu, "Measurement of Fracture Strains for Advanced High Strength Steels (AHSS) Using Digital Image Correlation," in *Proc. SAE'09*, 2009, paper, p. 482-486.
- [8] Donald House.(April/2016). Chapter 14: Spline Curves. [Online]. Available: <https://people.cs.clemson.edu/~dhouse/courses/405/notes/splines.pdf>
- [9] H. Moore: Production Problems: Cinematography. (Journal of the University Film Producers Association, 1(1), 2009. (April/2016) [Online]. Available: <http://www.jstor.org/stable>
- [10] Blue Waters - National Center for Supercomputing Applications. University of Illinois of Urbana-Champaign. (April/2016) [Online]. Available: <http://www.ncsa.illinois.edu/enabling/bluwaters>
- [11] Paraview. (April/2016) [Online]. Available: <http://http://www.paraview.org/>
- [12] David Eberly. (April/2016). Kochanek-Bartels Cubic Splines (TCB Splines). [Online]. Available: <http://www.geometrictools.com/Documentation/KBSplines.pdf>
- [13] Ira R. Forman, and Nate Forman. "Java Reflection in Action." Manning Publishing Company, 2004.

SESSION
POSTER PAPERS

Chair(s)

TBA

A Previsualization Method using BRDFs for 3D Printing

Seung-Woo Nam¹, In-Su Jang¹, Jin-Seo Kim¹ and Sung-Il Chien²

¹SW Content Research Lab., ETRI, Daejeon, Korea

²School of Electronics Engineering, Kyungpook National University, Daegu, Korea

Abstract – This paper presents a shading method to previsualize a result of 3D printing object by using measured BRDFs (bidirectional reflectance distribution functions). Visual artifacts often occur when a texture is generated from a measured BRDF for real-time rendering because of sampling error from the compressed texture. We newly generate the texture as continuously changed values in the coordinate frame based on the angles with respect to the half vector. Though the texture is compressed from 33.3Mbyte to a 274Kbyte, the visual artifacts are reduced in the rendering results by using the compressed texture for shading.

Keywords: BRDF, 3D printing, rendering, shading, texture

1 Introduction

A 3D model to be printed is need to previsualize the real-time rendering result to allow us to predict the colors and materials of the 3D printing result because 3D printing time is about ten hours to print a 10 cm-tall sized object. In this paper we present a real-time rendering method by using BRDFs to previsualize colors and materials of the 3D model to be printed. Addy Ngan et al. have proposed analytical BRDF driven by illumination of the image to allow user to navigate BRDFs [1]. Matusik et al. have proposed a data-driven reflectance model and introduced BRDF set of measurements [2][3]. We use the measured BRDF model [2] of MERL database [3] for the experiment instead of the analytical model because the reflectance of a 3D printing material can be measured and BRDF database can be constructed by measurement for many materials of the 3D printing.

2 Proposed Method

2.1 Data-driven BRDF model

Matusik et al. have used a coordinate frame based on the angles with respect to the half-vector instead of the standard coordinate frame as shown in Fig. 1(a), because sampling density is smaller near the specular highlight than far away from the specular reflection [1]. We also use the same coordinate system based on the half-vector as shown in Fig. 1(b) for the experiment. The BRDF function base on the half-vector [1] is defined as follows:

$$f(\phi_d, \theta_d, \theta_h) = f_r(\phi_i, \theta_i, \phi_o, \theta_o), \quad (1)$$

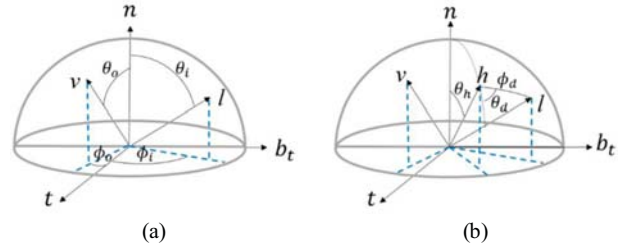


Fig. 1. (a) Standard and (b) half-vector based coordinate frames

where, ϕ_i and θ_i represent incoming light direction in spherical coordinates. ϕ_o and θ_o represent outgoing reflected direction in spherical coordinates as shown in Fig. 1(a).

2.2 Shading and texture construction

For rendering in real-time, we use OpenGL shading Language (GLSL) [4][5] and generate a texture from the measured BRDF [3]. We can find a 2D texture coordinates (u, v) for a given angles (ϕ_d, θ_d , and θ_h) with respect to the half-vector. Outgoing reflectance for the incoming light direction is defined by angles of ϕ_d, θ_d , and θ_h which are calculated in shading as follows:

$$\phi_d = \cos^{-1}(n_h \cdot l_h), \quad (2)$$

$$\theta_h = \cos^{-1}(h \cdot n), \quad (3)$$

$$\theta_d = \cos^{-1}(l \cdot h), \quad (4)$$

where $n_h = \cos(n \cdot h)h - n$ and $l_h = l - \cos(n \cdot h)h$. We generate a texture as the (u, v) coordinates of the column-based texture are corresponding to $(\phi_d, \theta_d \times \theta_h)$ as shown in Fig. 2(a). A visual artifact is observable for the generated texture as shown in Fig. 2(b).

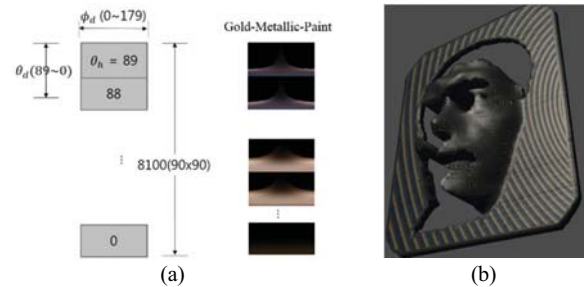


Fig. 2. (a) A structure of the generated texture and an example texture using a BRDF (Gold-Metallic-Paint) and (b) rendering result

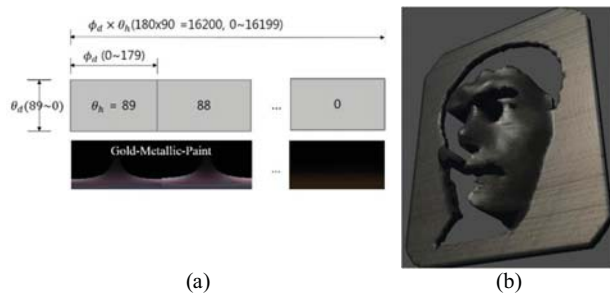


Fig. 3. (a) Proposed structure of the texture and an example texture using a BRDF (Gold-Metallic-Paint) and (b) rendering result

To reduce this artifact, We generate the texture as (u, v) coordinates of the row-based texture are corresponding to $(\phi_d \times \theta_h, \theta_d)$ as shown in Fig. 3(a). The artifact in Fig. 2(b) is reduced by using the texture as shown in Fig. 3(b) because the reflectance value of the texture is continuously changed to the angles with respect to the half-vector.

2.3 Experimental results

We have used the measured BRDFs of MERL database [3] for the experiment and implement the shader code using GLSL for real-time rendering. For Cherry-235 in BRDF database [3], we applied shading of the BRDF to a ball 3D model and compared rendering image using column-based texture with using row-based texture as shown in Fig. 4. A visual artifact is also significantly reduced in the result (Fig. 4(d)) by using the proposed texture structure. Additionally, we also show that the previzualized rendering result using Pearl-Paint BRDF in the database [3] and a real 3D printed object using FDM (fused deposition modeling) as shown in Fig 5.

3 Conclusions

We proposed a shading method using measured BRDF to visualize a model in real-time. The shading result allows us to predict the materials of the model before processing in the 3D printer. In the future work, we can measure full color materials of the 3D printer and then visualize previously and predict them without printing directly.

4 Acknowledgement

This research is supported by Ministry of Culture, Sports and Tourism (MCST) and Korea Creative Content Agency (KOCCA) in the Culture Technology (CT) Research & Development Program 2014.

5 References

[1] Addy Ngan et al. "Image-driven navigation of analytical BRDF Models," Eurographics Symposium on Rendering, pp. 399-407, 2006.

[2] Wojciech Matusik et al. "A Data-Driven Reflectance Model," ACM Transactions on Graphics, Vol. 22, Issue 3, pp. 759-769, Jul 2003.

[3] <http://people.csail.mit.edu/wojciech/BRDFDatabase/>

[4] Tomas Akenine-Möller, Eric Haines and Naty Hoffman, "Real-Time rendering". A K Peters Publishing Company, 2008.

[5] Randi J. Rost and Bill Licea-Kane, "OpenGL Shading Language". Addison-Wesley Publishing Company, 2009.

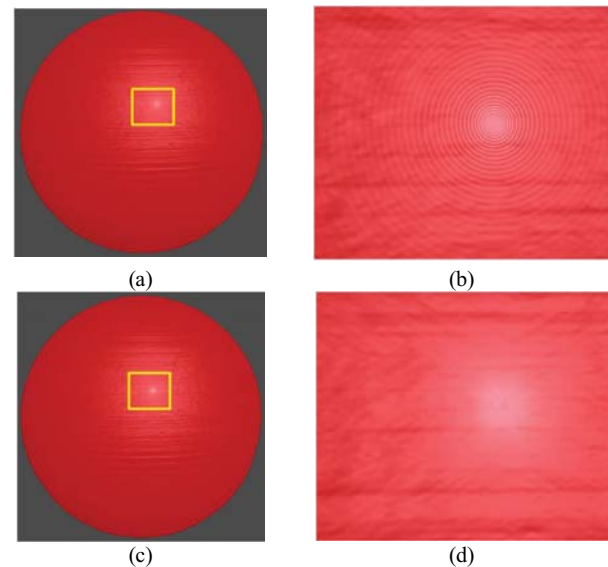


Fig. 4. (a) Result using column-based texture and (b) enlarged image corresponding the rectangle, (c) result using row-based texture (proposed) and (d) enlarged image corresponding the rectangle (proposed)

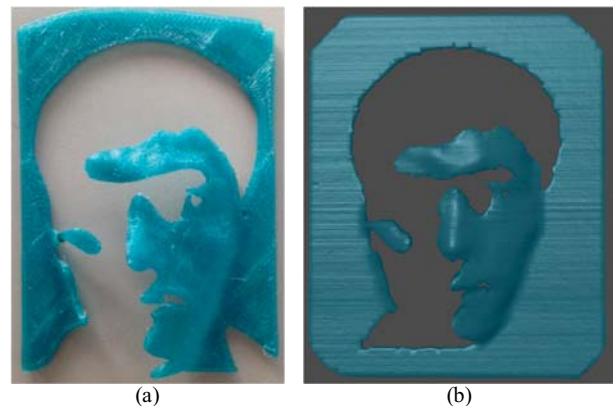


Fig. 5. (a) Real 3D printing result using FDM (fused deposition modeling) and (b) rendering result of row-based texture (proposed) using Pearl-Paint BRDF.

Interactive 3D Deformable Objects in Virtual Reality

Wook Song¹, Nak-Jun Sung², Min Hong²

¹Department of Computer Science, Soonchunhyang University,

²Department of Computer Software Engineering, Soonchunhyang University,
Asan-si, Chungcheongnam-do, South Korea

(wook2735@sch.ac.kr, njsung@sch.ac.kr, mhong@sch.ac.kr)

Extended Abstract/Poster Paper

Abstract – Recently, according to the advanced hardware technology, the price of sensors in HMD (Head Mounted Display) is getting cheaper and calculation speed in smartphone is fast enough to express a plausible virtual reality. Users wear the HMD device on their head and it shows the virtual scenes in front of users. Users can enjoy a wide display without any restrictions in space. Based on this device, users can now enjoy more realistic and plausible virtual reality contents in real-time. Although the use of HMD device keeps increasing rapidly, HMD related contents are insufficient for users to enjoy. In this paper, we proposed realistic 3D deformable object simulation based contents in VR.

Keywords: HMD, Virtual Reality, Deformable Object, Contents, Gear VR

1. Introduction

According to the survey result by Tractica[1], the virtual reality market with a combination of virtual reality contents and HMD devices in the world will be approximatively reached more than \$20,000 in 2020. For this reason, virtual reality contents and HMD devices in the IT industry have been under the spotlight.

Even though many 3D contents have been presented under various forms such as PC, console, and mobile phone, these 3D contents should be rendered in general 2D flat-panel displays. To represent the three dimensional effects based virtual reality in 2D flat-panel display, the binocular parallax based methods have been deeply studied[2]. This binocular parallax method can be classified as stereoscopic approach with additional glass and auto-stereoscopic approach without any devices which are shown in Figure 1.



Figure 1. Devices for Stereoscopic and auto-stereoscopic approach

In this paper, we propose the HMD based 3D deformable object with Samsung Gear VR. Users wear HMD device on their head and the HMD device tracks the change of user's point of view, and it can provide a plausible 3D environments. Recently, HMD device has been supplied by Sony, Oculus, Samsung, and several other companies. For traditional virtual reality systems, the background and objects should be created by designer and it can reduce the reality of 3D VR system. Therefore, the proposed system utilizes a 360 degree camera to generate 3D background environments for VR content.

Although previous 360 degree cameras have been used to record the dynamics of outdoor sports, they are recently applied in various fields with advance VR technology. IT companies have been developed some 360 degree cameras that can be connected with their own VR devices as shown in Figure 2.



Figure 2. Examples of 360 degree camera

Instead of creating all of virtual backgrounds and objects with computer graphics based technology for VR, the 360 degree camera can readily increase the reality with photographic quality. This technique can generate two kinds of VR: Static or dynamic virtual world. In static VR, only point of view for the background can be change from the fixed location. However, user can freely walk around the virtual world with unconstrained point of view in dynamic VR.

2. 3D Deformable Object Simulation in VR

The proposed VR system is generated with realistic background images which are achieved using the 360 degree camera, and the created virtual 3D deformable objects are inserted in dynamic VR system. Therefore, it can provide the immersive and realistic VR 3D contents in real-time. To reflect the change of background image according to user movement, other background image which is taken in the

same position should be updated in 3D VR system. The main process of the proposed interactive VR system is shown in the Figure 3.

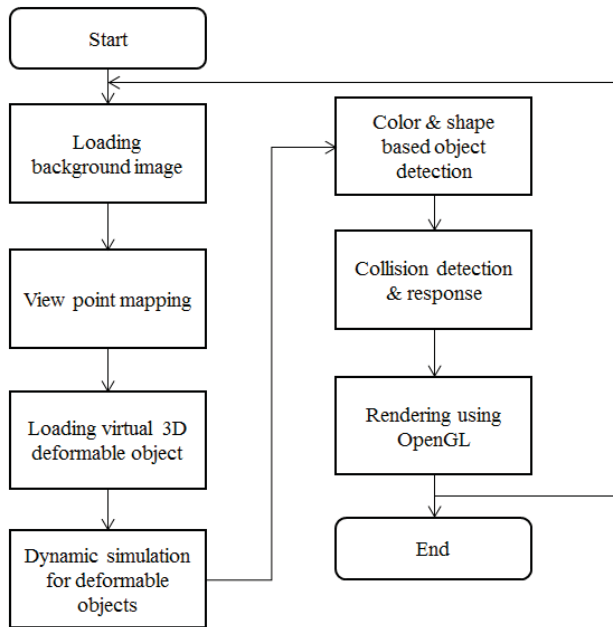


Figure 3. Flowchart of the proposed interactive 3D VR system with deformable objects

The interactive simulation of 3D deformable object for representing physically natural motions of virtual object is essential for this system. Unlike the traditional AR system which requires the collision handling between virtual objects and user which is shown in Figure 4[3], the proposed VR system is focused on the collision handling between virtual objects for dynamic interaction with photographic background environments. In addition, view point mapping technique between virtual 3D objects and background image are critical issue to combine them seamlessly.

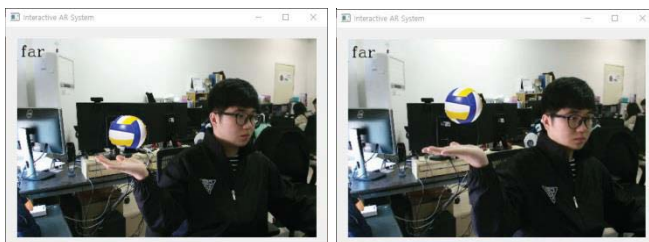


Figure 4. Interactive simulation between deformable object and hand in AR system

In this research, Euler integration method is used to estimate the next state of virtual 3D deformable object and OpenGL is applied to set up the viewing camera for perspective effect and to realistically render the virtual 3D objects with texture mapping. For dynamic simulation, 3D virtual deformable objects are created with tetrahedrons by Tetgen program[4].

For plausible simulation of 3D deformable object, collision contact detection and response are very important and it can be implemented with markerless approach for convenience. Therefore, computer vision based object detection and tracking technique are necessary for interactive simulation. In this research, color and shape based object detection algorithm and Meanshift algorithm for object movement tracking are designed to implement using OpenCV. The proposed VR system is implemented under Oculus Mobile SDK.

3. Conclusion

In this paper, we proposed the interactive 3D deformable object simulation with VR environment. The proposed VR system can provide realistic point of view for user and plausible interaction with virtual objects under photographic quality of background using 360 degree camera. We believe that the proposed system can provide more immersive contents in VR and can be a positive impact on various VR fields such as education, game, sport, medical content, rehabilitation, and simulation.

4. References

- [1] Tractica. "Virtual reality market to reach \$22 billion by 2020". <http://www.hypergridbusiness.com/2015/07/virtual-reality-market-to-reach-22-billion-by-2020>, 2015.
- [2] Steuer, Jonathan. "Defining virtual reality: Dimensions determining telepresence." *Journal of communication* 42.4, 73-93, 1992.
- [3] Hyo-Sum Yum, Min Hong, "Interactive Augmented Reality System with Kinect", CGI 2015(Computer Graphics International 2015), 2015.
- [4] Tetgen, <http://wias-berlin.de/software/tetgen/>

A Sensitivity Analysis for Deriving Dynamic and Evolutionary Rules in an Artificial Immune System-Cellular Automata Model

B. Curtis¹, C. Willy¹, and J. Bischoff¹

¹Dept. of Eng. Management & Systems Engineering, George Washington University, DC, USA

Abstract - *There have been several studies advocating the need for, and the feasibility of, using advanced techniques to support decision makers in urban planning and resource monitoring. One such advanced technique includes a framework that leverages the use of remote sensing and geospatial information systems (GIS) in conjunction with cellular automata (CA) to monitor land use / land change phenomena like urban sprawling. However, little research has been performed to analyze these frameworks' sensitivity to the input data (e.g. imagery). New technology is promising better data more frequently; all with an associated price tag. Understanding sensitivity provides decision-makers and analysts the necessary information to procure just the right amount, and type, of data. Our research focuses on arming analysts and decision makers with this information.*

Keywords: Sensitivity, GIS, Remote Sensing, Cellular Automata

1 Introduction

Urban environments are complex systems presenting dynamic spatial and temporal features along with emergent and non-linear growth behaviors. Understanding and predicting these dynamic phenomena can be difficult, however useful. One example of a concrete problem at the nexus of engineering and land use are the issues associated with urban sprawl – the migration, or expansion, of our populations away from city centers outward towards low density, residential, and usually highly automobile-dependent areas. The research of Ewing, et al. demonstrated a connection between urban sprawl and an epidemic affecting the United States today, obesity, stating that "residents of more compact counties have lower BMIs and lower probabilities of obesity and chronic diseases" [1]. The CDC also published a report [2] attributing this epidemic to local policies and our physical environments in which we live, including the lack of physical activity due to residential zoning strategies requiring people to drive, vice walk, to work and school simply because it is too far. Ultimately, according to research conducted by Ogden, et al. [3], more than one-third of adults and 17% of youth are considered obese in the United States today.

As a result of these findings, several studies have been published advocating the need for, and the feasibility of, using advanced techniques to model and simulate urban sprawl and provide decision makers the tools necessary for urban planning and resource monitoring. One such advanced technique includes a framework that leverages the use of remote sensing and geospatial information systems (GIS) in conjunction with a cellular automata (CA) to monitor land use / land change phenomenon like urban sprawling. While this technique has been shown to be a viable solution to simulate the complex nature of urban sprawl, little research has been conducted analyzing the sensitivity to input data (i.e. imagery). Therefore, this research seeks to analyze the relationship between a GIS-CA model and the imagery which feeds it; looking at frequency, imagery resolution, and fragmentation (i.e. partial coverage). A case study simulating urban sprawl for the city of Albuquerque, New Mexico will be used to analyze the relationship between GIS-CA and the data which feeds it.

2 Methodology

The use of *geospatial information systems (GIS)* as we know them today has been around for decades; over 180 years in its purest form of spatial analysis. In the simplest of terms, GIS is a system used to manipulate, analyze, and visualize all different types of geospatial information (e.g. land use, roads/streets, bodies of waters, elevation information, etc.). When used in raster form, GIS information is stored as a layered grid of data; each layer is a two-dimension matrix representing a specific piece of information as it relates to the geographic area being studied.

The use of cellular automata (CA) to simulate the evolution of complex systems, spatially and temporally, has been widely applied to urban sprawl research. CA-based models use a 'bottoms up' approach where a simple set of transition rules govern the interactions between cells. They have the ability to represent non-linear, spatially dependent, stochastic processes and simulate the evolution of the complex systems [4]. According to Liu, et al. [5], "this 'bottoms-up' approach coincides with complexity theories stating that a complex system comes from the interactions of simple subsystems." The fact that a CA-model is cell based – or a

two-dimensional matrix – makes it perfect to couple with raster GIS and remote sensing data for studying the complex nature of urban sprawl.

One method for deriving the dynamic transition rules of a CA-model takes its inspiration from nature; an intelligent computational technique capable of solving complex problems known as artificial immune systems (AIS). Liu et al. [5] first used an AIS-based CA model to determine policy impacts on land use and found its ability to adapt, learn, organize, and memorize new information was extremely promising for complex geographical problems. Much like natural systems, AIS uses the concept of ‘antigens’ and ‘antibodies’ to derive the transition ‘rules’ for the CA-model. More specifically, cells needing to be classified are the ‘antigens’ and the classifiers which will assign the proper land use (e.g. urban) to such cells are the ‘antibodies’. He, et al. [6] proposed a process to calculate the evolution probability for the urban CA-model through a standard recycling process: defining antigens, generating an initial set of antibodies, calculating antibody/antigen affinities, clonal selection, and the mutation and updating of the antibodies. The high-level process flow of the AIS-based CA model is described in Figure 1. The crux of our research focuses on this model’s ability to be repeatedly updated when new remotely sensed imagery becomes available; dynamically adapting and learning to the introduction of new antigens (i.e. information) into the simulation.

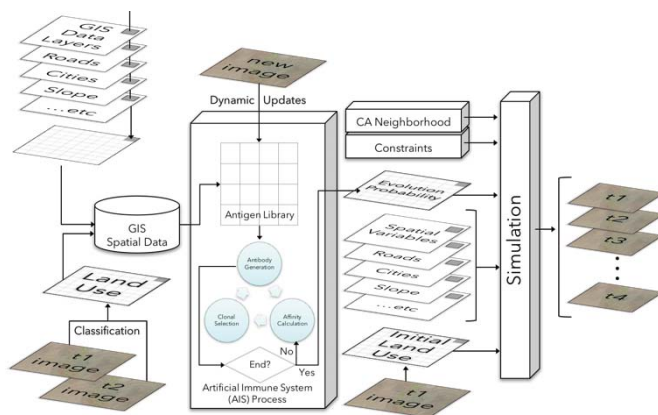


Figure 1. Generating dynamic transition rules with an AIS-based CA model

Today, companies (e.g. PlanetLabs) are delivering new technology which is promising better data more frequently all with an associated price tag. Understanding the sensitivity of this model provides decision-makers and analysts the necessary information to procure just the right amount, and type, of data. Our research focuses on arming analysts and decision makers with this information. We intend to do this by incorporating a sensitivity analysis into a proven urban land-use model, AIS-CA. By analyzing variables such as temporal frequency, image resolution, and segmentation, our hope is to increase the understanding of the relationship between input data (i.e. imagery) and the model’s ability to predict land use (i.e. accuracy).

A hypothetical example for our temporal frequency based scenario would include using historical data to seed the antigen library and then build the antibody library. Year sets could be spaced by 10 years (e.g. 1990, 2000, 2010), 5 years (1990, 1995, ..., 2010), and 1 year or less (depending on available data). Once the transition rules have been obtained, a simulation forward to the most recent available imagery (e.g. 2015) could be used to measure the accuracy of each year set. To measure the findings between simulation results and actual situations, we use a cell-level comparison analysis (i.e. pixel by pixel) and apply a ‘figure of merit’ (FoM) metric [7]. This ‘FoM’ is a ratio by which we measure the number of cells correctly simulated as urbanized cells (numerator) divided by the total number of instances.

3 Conclusions

New technologies are promising to deliver a radical change in access to information. However, our research indicates a lack of understanding into how the complex and dynamic models used to monitor sprawl can leverage this new technology. Therefore, this research will answer the questions if more, finer, and partial remote sensing data can be used to improve an AIS-based CA model’s accuracy.

4 References

- [1] R. Ewing, G. Meakins, S. Hamidi, and A. C. Nelson, “Relationship between urban sprawl and physical activity, obesity, and morbidity - Update and refinement,” *Heal. Place*, vol. 26, pp. 118–126, 2014.
- [2] D. Keener, K. Goodman, A. Lowry, S. Zaro, and L. K. Khan, “Recommend Community Strategies and Measurements to Prevent Obesity in the United States: Implementation and Measurement Guide,” 2009.
- [3] C. L. Ogden, M. D. Carroll, B. K. Kit, and K. M. Flegal, “Prevalence of childhood and adult obesity in the United States, 2011–2012,” 2014.
- [4] C. He, N. Okada, Q. Zhang, P. Shi, and J. Zhang, “Modeling urban expansion scenarios by coupling cellular automata model and system dynamic model in Beijing, China,” *Appl. Geogr.*, vol. 26, no. 3–4, pp. 323–345, 2006.
- [5] X. Liu, X. Li, X. Shi, X. Zhang, and Y. Chen, “Simulating land-use dynamics under planning policies by integrating artificial immune systems with cellular automata,” *Int. J. Geogr. Inf. Sci.*, vol. 24, no. 5, pp. 783–802, 2010.
- [6] Y. He, B. Ai, Y. Yao, and F. Zhong, “Deriving urban dynamic evolution rules from self-adaptive cellular automata with multi-temporal remote sensing images,” *Int. J. Appl. Earth Obs. Geoinf.*, vol. 38, pp. 164–174, 2015.
- [7] R. G. Pontius, W. Boersma, J. C. Castella, K. Clarke, T. Nijs, C. Dietzel, Z. Duan, E. Fotsing, N. Goldstein, K. Kok, E. Koomen, C. D. Lippitt, W. McConnell, A. Mohd Sood, B. Pijanowski, S. Pithadia, S. Sweeney, T. N. Trung, A. T. Veldkamp, and P. H. Verburg, “Comparing the input, output, and validation maps for several models of land change,” *Ann. Reg. Sci.*, vol. 42, no. 1, pp. 11–37, 2008.

Development of CSEA and TSEA software for predicting high-frequency dynamic responses in complex plate structure

Hyeonmin Yang¹, Young-Ho Park²

¹Department of Eco-friendly Offshore FEED Engineering, Changwon National University, Changwon, Korea

²Department of Naval Architecture & Marine Engineering, Changwon National University, Changwon, Korea

Abstract - Built-up structures such as ships, cars, and aircrafts can have noise and vibration problems in high-frequency ranges. The developed software in this paper deals with the acoustic and vibrational energetics of built-up structures composed of acoustic cavities and plates, and is implemented using MATLAB language. Finally, to validate the developed software, simple applications to a vessel's superstructure were successfully performed.

Keywords: SEA, TSEA, Built-up structure, acoustic and vibrational responses, MATLAB

1 Introduction

The built-up structures require a higher level of performance in terms of noise and vibration than existing structures due to the high development cost of projects and the working environment characteristics of high wage earners. Therefore, a reliable prediction technique of noise and vibration performance in the early stage of the design of built-up structures is necessary. In this paper, in order to analyze the effective broadband noise and vibration of the built-up structure and the numerical analysis for some ideal offshore plant structures was performed. For the analysis result, the validity was verified by comparing with the result of the commercial SEA program VAOne.

2 Theory

2.1 Classical statistical energy analysis

Classical statistical energy analysis (CSEA), the representative analytic method of statistical approaches, can effectively predict the space- and frequency-averaged behavior of built-up structures at high frequencies where the modal overlap of structural components is high.³ In the fundamental principles of SEA, the averaged power flow between two coupled groups (subsystems) of dynamical modes is proportional to the difference in the averaged modal energies. Power flows out of a subsystem through dissipation or by transmission to another subsystem. Power flows into a subsystem either by transmission from another subsystem or from an external source of excitation.

2.2 Transient Statistical Energy Analysis

The Transient Statistical Energy Analysis (TSEA) can predict the transient responses of a structure. TSEA is based on the basic concept of the SEA. Energy should establish the equilibrium in the same time area, and the power flow between the subsystems must also consider the effect of the given time interval. In addition, TSEA is a very useful tool in simulating a decay rate measurement in order to verify the damping levels used in a model.³

3 Developed Program

3.1 Composition of Program

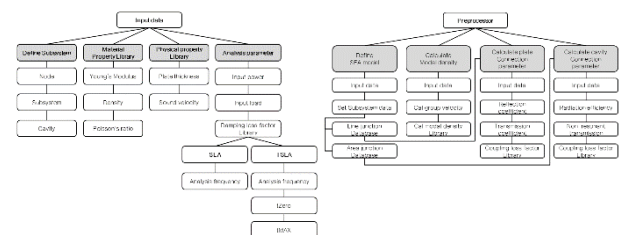


Fig. 1 Data structure of Input data (left) and Flowchart of Preprocessor (right) in program

The data structure of this program is effectively structured for the database construction of the SEA parameters of various flooring, including the internal damping loss factor, the modal density, and the coupling loss factor of the offshore plant structure. A database can be constructed from the input data. In the preprocessor, these databases needed for the main processor are constructed by referencing the database needed for the input data. In this program, radiation efficiency can be calculated by using Maidanik's radiation efficiency equation.⁴ Also, a non-resonant transmission coefficient can be calculated using Beranek's equation.¹ These calculated dates can be used to calculate coupling loss factor and added to the coupling loss factor library. In this library, plates can be defined by the type of propagated wave between subsystems. Also, cavities can be defined as a library by its cavities.

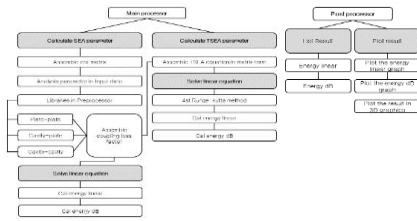


Fig. 2 Flowchart of Main processor(left) and Post processor(right) in program

The main process can then be solved. If the CSEA linear equation is configured in the matrix form based on the database constructed in the above process and then solved, the energy according to each subsystem can be obtained. The TSEA power balance equation is configured in the matrix form based on the shared coupling loss factor library in CSEA.

3.2 Composition of Program (GUI)

Developed program in this paper consists of the following figure 3.

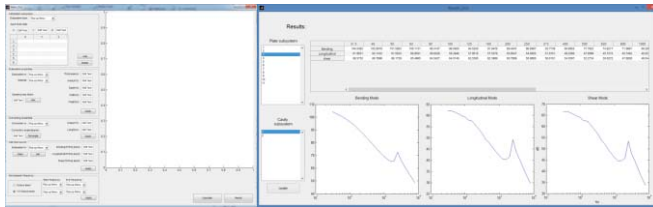


Fig. 3 Main GUI (left) and result GUI (right) in program

4 Verification

The analysis results for the cabin structure were compared with the commercial SEA program VAone.

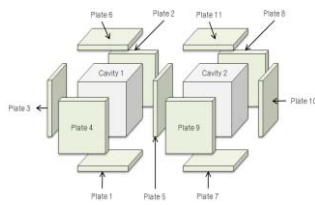


Fig. 4 SEA model(main noise source: cavity 1, 1W)

Table 1 Material and physical properties in cavity

Fluid	Density (kg/m ³)	Volume (m ³)	Surface (m ²)	Perimeter length(m)
Air	1.21	8	24	24

Table 2 Material and physical properties of plate

Material	Dimensions (m)	E(N/m ²)	Density (kg/m ³)	Poisson's ratio
Steel	2*2*0.005	2.1*10 ¹¹	7800	0.3125

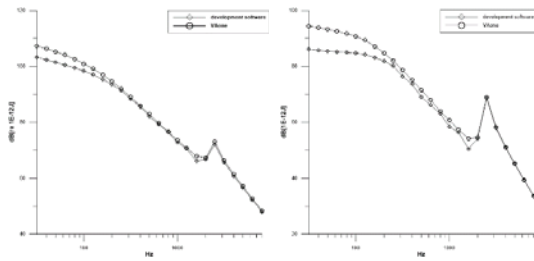


Fig. 5 Comparison between developed software results and VAone results (left: flexural energy in plate 5, right: acoustic energy in cavity 2)

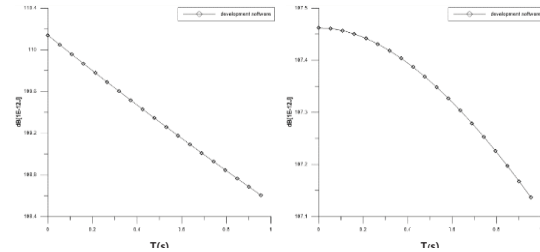


Fig. 6 developed software results (left: flexural energy in plate 1, right: acoustic energy in cavity 5)

5 Conclusions

In this study, the authors developed a statistical energy analysis program for the prediction of the effective noise and vibration response of a system in which the plates and the acoustic cavities are coupled as the built-up structure. They compared the results of the program with the results of the actual commercialized VAone in order to verify the reliability of the program. In general, it was shown that the tendency according to the frequency and the noise and vibration energy level agrees well with those of the VAone.

6 Acknowledgements

The authors of this paper were partly supported by the Brain Korea21Plus Projects

7 References

- [1] Leo. L. Beranek, István L. Vér. "Noise and Vibration Control Engineering". John Wiley & Sons, 1992
- [2] Richard. H. Lyon. "Shock spectra for statistically modelled structures"; Shock and Vibration Bulletin, Vol. No.40, 17-23, 1969
- [3] Richard. H. Lyon, Richard. G. DeJong. "Theory and application of statistical energy analysis second edition". Butterworth-Heinemann, 1995
- [4] Richard. H. Lyon, Gideon. Maidanik. "Power flow between linearly coupled oscillators"; Acoustical Society of America, Vol. No.34, Issue No.5, 623-639, 1962

SESSION
LATE BREAKING PAPERS

Chair(s)

TBA

SCATTERED DATA MODELING USING A GPU: A CASE STUDY

B. Cai, Y. Xiao, T. O'Neil, Z. Duan

Department of Computer Science, University of Akron, Akron, Ohio, USA

Abstract - This paper presents a case study on how to use GPUs to accelerate scattered data modeling in a two-step approach of scattered data visualization. Measurements were made to correlate GPU performance with various modeling parameters. The following results were obtained: (1) Adjusting internal modeling parameters belonging to the modeling function has no impact on computing time. (2) Modeling time is linearly proportional to the size of the intermediate grid. (3) Speedup by the GPU increases as grid size increases. (4) Efficiency of GPU utilization increases as the grid size increases. (5) Data communication between the GPU and the host hinders the efficiency of GPU utilization. But, its relative impact decreases as grid size increases. (6) Among the two Block -Oriented Localized Data Modeling methods, the Dynamic Local Block Data Modeling method consumes more time than the Static Local Block Data Modeling method. Future work includes GPU speedup of other modeling functions and accuracy of the intermediate grid.

Keywords: GPU, scattered data, modeling, interpolation, visualization

1 Introduction

Data visualization is a communication tool for people to present, analyze and understand data. It has been widely used in many disciplines to help scientists and engineers to study all kinds of data; for example, traffic data in intelligent transportation systems [1], health data for wellness monitoring [2], grazing-incidence X-ray scattering data for crystal structure analysis [3], soil bacteria susceptibility for environmental studies [4], city data for urban planning [5], chemistry data for chemical information modeling [6], marine forecast data for oceanic studies [7], and big data for spatial analysis [8].

When the data to be visualized do not fill the volume of interest completely, as in the case of many real world applications where sample data are measured values at suspected areas in the volume of interest, data modeling becomes an inevitable part of data visualization. This is especially true for scattered data.

Scattered data are data unevenly distributed or randomly spread over the volume of interest. Examples of such data can be found in environmental studies, oil exploration and mining. Each sample data point consists of three values for the position (x,y,z) and one value for the attribute (v). The

attribute is the data to be visualized, which could be, for example, the concentration of a chemical compound in a polluted field. Quick interactive visualization of scattered data is in demand. A commonly used approach for scattered data visualization consists of two steps [9] (Figure 1). The first step involves converting the scattered sample data into a 3D uniform grid, the intermediate grid, after which the intermediate grid is rendered using grid-based visualization techniques such as Marching Cubes [10]. The purpose of the first step is to model the data onto the volume of interest based on the input sample data. The second step is to render the modeled volume into graphics for visualization. Both interpolation and finite element method have been used in the modeling step [11-15]. Constraining methods have also been added to increase the accuracy of modeling [16]. One of the constraining methods is localization [16], in which only nearby sample points are selected to model the data value on a given grid node.

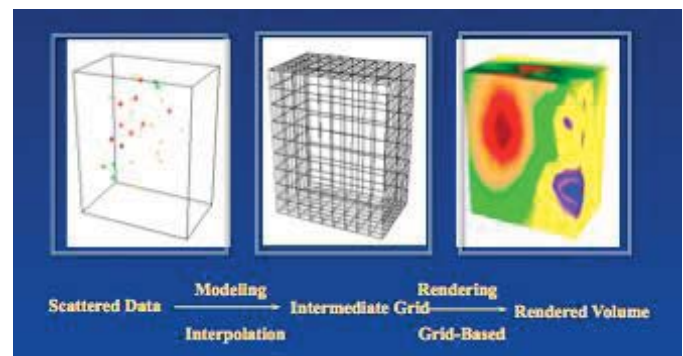


Figure 1. Two-Step Approach to Scattered Data Visualization.

The advent of GPU based parallel processing has greatly improved the performance of many graphics-intensive applications, including visualization [17,18]. A premier programming API for GPUs is CUDA [19]. CUDA not only allows the use of a GPU to speed up graphics display but also allows the use of a GPU to speed up other parallelizable computing [19]. Since CUDA threads are grouped into blocks and grids, the grid based two-step approach to scattered data visualization makes it a good candidate for GPU based parallel processing. The intermediate grid helps to parallelize the code in both the modeling step as the output and the rendering step as the input.

This research investigates how GPUs can be used to improve the speed of scattered data modeling for the purpose of visualization, i.e., the performance enhancements of the

first step in the two-step approach to scattered data visualization. As an initial examination of the issue, we concentrate on interpolation based scattered data modeling.

2 System Implementation

To investigate how GPUs can be used to improve performance of interpolation based scattered data modeling for the purpose of visualization, we have built a system that can test the performance enhancements with various parameter values, various intermediate grid sizes and various localization techniques.

2.1 Interpolation Methods

Interpolation methods construct new data points on the intermediate grid with a discrete set of known input sample data points [20]. Given a set of n sample points,

$$P_i(x_i, y_i, z_i), i = 1, 2, \dots, n, \quad (1)$$

with a sample data value at each point

$$v_i, i = 1, 2, \dots, n, \quad (2)$$

we construct an interpolation function $f(x, y, z)$ that is valid everywhere inside the domain of interest and satisfies the condition of

$$f(x_i, y_i, z_i) = v_i, i = 1, 2, \dots, n. \quad (3)$$

One of the commonly used interpolation methods is the Shepard method [21]. The mathematical expression of the method is:

$$f(x, y, z) = \sum_{i=1}^n v_i d_i^{-\alpha} / \sum_{i=1}^n d_i^{-\alpha} \quad (4)$$

Where, parameter α is a positive real number; d_i is the distance between sample point i and point $P(x, y, z)$. The inverse-distance weighted method is a special case of the Shepard's method with $\alpha = 1$. Changing the parameter alters the values being interpolated onto the intermediate grid.

2.2 Intermediate Grids

The intermediate grid is the bridge that connects the first and the second step of the two-step approach. The modeling step generates the intermediate grid with data values on the grid nodes interpolated from the original sample data. The rendering step uses the intermediate grid as the input representing the original sample data in the volume of interest and renders the intermediate grid onto the graphics display. Traditional grid-based visualization techniques [22] are used in the second step. The intermediate grid is a 3D grid of dimensions (n_x, n_y, n_z) . The size of the dimensions affects the accuracy of the grid, subsequently the accuracy of the rendering. A grid of larger size takes more time to generate at

the modeling step and more time to render at the rendering step.

2.3 Localization

One of the challenges of using interpolation methods to model scattered data is the accuracy dilemma [23]: Even though the interpolated data values are constrained by Equation 3 to be 100% accurate at the sample data points, the interpolated values of the grid nodes of the intermediate grid vary with different interpolation methods and even with the same method but different parameters. Studies have been conducted to measure the errors in the intermediate grid in representing the original sample data [24]. One way to reduce such error is to localize the interpolation methods [16].

Localized interpolation methods use only nearby sample points to interpolate a grid node value. The nearby input sample points for each grid node can be selected by (a) number, where a specific number of the nearest sample points to the grid node are selected; or (b) region, where only the sample points within a local region are selected. The latter is further divided into two: Range-Oriented Localized Data Modeling (ROLDM) and Block-Oriented Localized Data Modeling (BOLDM).

ROLDM is a distance-based localized data modeling method. Each time we interpolate the data value onto a node of the intermediate grid, we draw a sphere using this grid node as the center, and only use the sample points within the sphere to compute the data value at the grid node. The radius of the sphere is a modifiable parameter. If the radius is large enough to contain all original sample points, the interpolated result will be the same as that of the original data modeling method without localization. BOLDM is similar to ROLDM except that we use a cube centered around the grid node instead of a sphere to define the boundary for selecting the local sample data. Only sample points within the cube are used to compute the data value at the grid node.

2.4 GPU Based Interpolation

To take advantage of the GPU, we implement the Shepard's interpolation method as a CUDA kernel and let each core processor of the GPU run a kernel thread for each grid node. Since the interpolation of each grid node is independent of other nodes, the parallelization of the interpolation based modeling code is relatively easy. After each core processor is assigned to a grid node, all core processors will run the interpolation method simultaneously. The speedup will be proportional to the number of core processors in the GPU.

An important part of GPU based computation is to transfer input from the host to the GPU and transfer the output from the GPU to the host. In terms of scattered data modeling, that is to transfer the input sample data from the

host to the GPU and transfer the output intermediate grid data from the GPU to the host.

CUDA groups threads hierarchically [17] into grids of blocks, with each block being formed as a grid of threads. There are two types of memories on the GPU: global and shared. The global memory is for the whole GPU. Each block has a faster memory shared only by the core processors within the block. The data in the shared memory of a block are copied in and out to the global memory of the GPU.

Block-Oriented Localized Data Modeling (BOLDM) fits well with the block structures of CUDA threads and memory. Since each thread is assigned to a grid node, we can load the sample points within the constraining block to the shared memory of the thread block. Below is the pseudo code for BOLDM.

- 1) Define the size of the intermediate grid.
- 2) Allocate arrays for input sample points and intermediate grid points on the host.
- 3) Read sample data points into the host arrays.
- 4) Allocate arrays for sample points and intermediate grid points on the GPU.
- 5) Calculate GPU block and GPU grid dimensions according to the size of the intermediate grid.
- 6) Divide sample points into blocks according to the GPU grid dimension.
- 7) Copy the input sample points from the host to the GPU.
- 8) Invoke the GPU kernel function by passing the block dimension, grid dimension, the pointers to the sample data array and intermediate grid array.
- 9) Allocate shared memory.
- 10) Each kernel thread performs the following steps:
 - a) Load this kernel's corresponding block of input sample data from the global memory to the shared memory.
 - b) Synchronize with other threads and wait until all input sample data are load into the shared memory.
 - c) Interpolate this thread's corresponding intermediate grid node data value by using the corresponding block of input sample data.
 - d) Write the interpolated data value into the intermediate grid array on the GPU.
- 11) Copy the interpolated intermediate grid data values from GPU to the host.
- 12) Free GPU memories.

3 Results and Analysis

As a case study, we used the scattered data modeling system implemented above to model a set of sample data collected in the real world at a polluted chemical plant [23]. The data value (v) at each sample point is the concentration in ppb (parts per billion) of a toxic chemical agent at a given location (x,y,z). We collected performance measurements of

the modeling system with different modeling parameters, different intermediate grid sizes and different localization methods.

3.1 GPU and Performance Measurement Tool

The GPU that we used was an NVidia GeForce GT 525M on a Dell laptop computer. The following are the specifications of the GPU:

CUDA Driver Version / Runtime Version: 5.5/5.5
 CUDA Capability Major/Minor version number: 2.1
 Total Number of CUDA Cores: 96
 Total amount of global memory: 1024 Mbytes
 Total amount of shared memory per block: 49152 bytes

We used the NVIDIA Visual Profiler [25] to measure the performance of the system. The NVIDIA Visual Profiler is a cross-platform performance-profiling tool that delivers vital feedback for optimizing CUDA applications.

3.2 Data Communication Speed

In order to use a GUP, we need to send input data to the GPU from the host and get the output data from the GPU back to the host. Data communication between the GPU and its host is an inevitable overhead for GPU based computing. This overhead hinders the performance of the overall system.

Table 1 shows the data size, time, and speed of copying input data from the host to the GPU (device) and the output data from the GPU to the host with various intermediate grid sizes. The kernel function implements the Shepard's interpolation method without localization. All input sample data were copied to the GPU and shared by all GPU cores. Each GPU core runs the kernel function for one grid node and computes the interpolated data value for the grid node. The computed data values on the grid nodes were then copied back from the GPU to the host.

Table 1. Measurements of Data Communication between the GPU and Host with Various Intermediate Grid Sizes (grid sizes measured in $n_x \times n_y \times n_z$, time measured in ms).

Grid Size	Data Copy Host to Device data size	Data Copy Host to Device time	Data Copy Host to Device speed	Data Copy Device to Host data size	Data Copy Device to Host time	Data Copy Device to Host speed
1*1*1	2156KB	0.672	3.06GB/s	4bytes	0.002	1.66MB/s
2*2*2	2156KB	0.672	3.06GB/s	32 bytes	0.002	14.67MB/s
4*4*4	2156KB	0.672	3.06GB/s	256 bytes	0.002	117.38MB/s
8*8*8	2156KB	0.640	3.21GB/s	2 KB	0.002	847.71MB/s
16*16*16	2156KB	0.704	2.92GB/s	16KB	0.004	3.29GB/s
32*32*32	2156KB	0.704	2.92GB/s	128KB	0.021	5.56GB/s
64*64*64	2156KB	0.672	3.06GB/s	1MB	0.160	6.09GB/s
128*128*128	2156KB	0.672	3.06GB/s	8MB	2.100	2.82GB/s

We can make the following observations from the data in Table 1.

- a) Since the size of the input sample data does not change with the output intermediate grid size, the time and speed of copying the input sample data from the host to the device stay nearly constant: ~ 0.672 ms and ~ 3.06 GB/s.
- b) The size of the output data is the number of grid nodes multiplied by the output data size per grid node: $(n_x \times n_y \times n_z) \times \text{sizeof}(v)$. We used floating point computation for v , $\text{sizeof}(v) = 4$ bytes. When grid size is less than or equal to $8 \times 8 \times 8$ the time needed to copy data from the device to the host is less than 0.002 ms. Because the smallest time unit of NVIDIA Visual Profiler is 0.002 ms, all time measurements that were smaller than 0.002 ms are shown as 0.002 ms in the table.
- c) The time needed to copy output data from the device to the host increases as the grid size increases. This is because the output data size increases as the grid size increases.
- d) The speed of copying from the device to the host increases as the grid size, hence data size increases. This is due to larger data size filling the output pipeline fuller. But the speed peaks at 6.09 GB/s when the data size is 1 MB and the grid size is $64 \times 64 \times 64$.

3.3 GPU Computation and Communication Time

Once the input sample data are copied to the GPU, the GPU cores run the kernel function to model the data onto the intermediate grid nodes, one core per node. When the number of nodes is larger than the number of cores (96 for the NVidia GeForce GT 525M), each core is used repeatedly once for each group of 96 nodes. The Kernel Compute Runtime measures the computing time for each core to finish all nodes assigned to it. Table 2 shows the Kernel Compute Runtime for various grid sizes along with the total Runtime and total Data Communication Time for each core, where

$$\text{GPU Runtime} = \text{Kernel Compute Runtime} + \text{Data Communication Time} \quad (5)$$

$$\begin{aligned} \text{Data Communication Time} = & \\ & \text{Host to Device Data Copy Time} + \\ & \text{Device to Host Data Copy Time} + \\ & \text{Device Memory Malloc Time} \end{aligned} \quad (6)$$

We can make the following observations from the data in Table 2.

- a) The Kernel Compute Runtime is too small for the NVIDIA Visual Profiler to measure when the grid size is less than or equal to $8 \times 8 \times 8$ and is displayed as the minimal unit of 0.002 ms.

Table 2. Computation and Communication Time (in ms).

Grid Size	GPU Runtime	GPU Kernel Compute Runtime	Data Copy Host to Device time	Data Copy Device to Host time	Malloc Memory time	Data Communication time
1*1*1	50.2	0.002	0.672	0.002	50	51
2*2*2	51.2	0.002	0.672	0.002	51	52
4*4*4	51.2	0.002	0.672	0.002	51	52
8*8*8	54.2	0.002	0.640	0.002	54	55
16*16*16	55.9	0.009	0.704	0.004	55	57
32*32*32	65.804	6.804	0.704	0.021	59	60
64*64*64	115.702	53.702	0.672	0.160	62	65
128*128*128	485.898	428.898	0.672	2.100	57	60

- b) When the intermediate grid size is larger than or equal to $32 \times 32 \times 32$, the Kernel Compute Runtime increases linearly with the intermediate grid size (approximately 8 times from $32 \times 32 \times 32$ to $64 \times 64 \times 64$ and 8 times again from $64 \times 64 \times 64$ to $128 \times 128 \times 128$.)
- c) When the intermediate grid size is in between $8 \times 8 \times 8$ and $32 \times 32 \times 32$, the Kernel Compute Runtime increases nonlinearly with the intermediate grid size.
- d) Memory Malloc Time measures the time needed to allocate memories on the GPU. In table 1, the minimum memory allocation time is 50 ms. It increases as the size of the intermediate grid increase, but not much. It increased about 20% when the size of the intermediate grid increased from $1 \times 1 \times 1$ to $128 \times 128 \times 128$.
- e) We count Device Memory Malloc Time as part of the Data Communication Time since it is part of the non-computational overhead to get the data into the GPU. The time it takes to allocate memory on the GPU is much longer than copying data into and out off the GPU.
- f) As the size of the intermediate grid increases the computation time increases. The ratio of Data Communication Time over Kernel Compute Runtime decreases (see Table 3). The ratio changed from 8.818 for grid size $32 \times 32 \times 32$ to 0.140 for grid size $128 \times 128 \times 128$. Thus, the efficiency of the GPU based modeling program increases as the intermediate grid size increases.

Table 3. Ratio of Communication and Computation Time.

Grid Size	GPU Kernel Compute Runtime (ms)	Data Communication Time (ms)	Ratio
32*32*32	6.804	60	8.818
64*64*64	53.702	65	1.210
128*128*128	428.898	60	0.140

In addition, we tested the system with different α values of the Shepard method. The tests shown no changes in GPU performance (GPU Runtime and Data Communication Time) if we only change the α value without changing other parameters. The reason for the lack of GPU performance change is because α is an internal modeling parameter belonging to the modeling function. Its value change will not result in an increase of memory or an increase of computing steps.

3.4 GPU Speedup Factor and GPU Usage Efficiency

To further investigate the efficiency of the GPU based modeling program, we implemented a CPU based sequential modeling program of the same interpolation method. By comparing the performance of the GPU based program against the CPU based program, we can measure the GPU speedup factor and GPU usage efficiency. They are defined below.

$$\text{Speedup Factor} = \frac{\text{Runtime of CPU Program}}{\text{Runtime of GPU Program}} \quad (7)$$

$$\text{Efficiency} = \frac{\text{Runtime of CPU Program}}{(\text{Runtime of GPU Program} \times \text{Number of GPU Cores})} \quad (8)$$

We define the speedup factor as the ratio of the GPU Runtime and the CPU Runtime. It measures the relative benefit of using the GPU. Efficiency is the speedup factor divided by the number of GPU core processors. It measures how efficiently we use the GPU core processors.

Various sizes of the intermediate grid have been chosen to compare the CPU and GPU programs. Table 4 and Figure 2 show the running times in milliseconds (ms) and each value is the average of ten experimental measurements. Both speedup factor and efficiency increase as the size of the intermediate grid increases, which is due to the reduction of the data communication overhead relative to the overall GPU runtime.

Table 4. CPU and GPU Program Comparison.

Grid Size	CPU Runtime	GPU Runtime	Speed Up Factor	Efficiency
1*1*1	0.003	50.2	0.00018	<0.0001
2*2*2	0.030	51.2	0.00080	<0.0001
4*4*4	0.253	51.2	0.00494	<0.0001
8*8*8	1.931	54.2	0.03562	0.0004
16*16*16	17.341	55.9	0.31021	0.0032
32*32*32	140.123	65.804	2.12765	0.0221
64*64*64	1014.421	115.702	8.76768	0.0914
128*128*128	13508.452	485.898	27.81003	0.2552

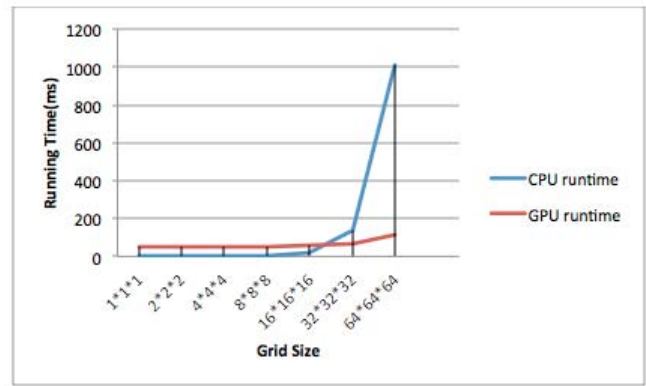


Figure 2. CPU and GPU Program Comparison.

3.5 Localization Methods

Block-Oriented Localized Data Modeling (BOLDM) uses a cube centered at an intermediate grid node to select local sample points. Only sample points within the cube are considered local and are used to compute the data value at the grid node. Since the center of the cube is at the grid node to be computed and each GPU core is assigned to a different grid node, GPU cores within the same GPU block may use different sets of local sample points. We term this method Dynamic Local Block Data Modeling as compare to the Static Local Block Data Modeling, where we statically divide the sample data volume into small blocks and copy the sample data points in each small block into the shared memory of a GPU block. So each small block of sample points resides in their own shared memory and will not change at runtime. During modeling, each GPU core only uses the sample points in the shared memory of its residing GPU block. Each grid node has its own block ID and its data value is interpolated by using the sample points that have the same block ID.

Comparing with this static assigning of local blocks, the Dynamic Local Block Data Modeling Method may need to fetch sample data from the global memory because each GPU core in a GPU block uses a different set of local sample points, and some of the local sample points may have not been copied into the shared memory of the block from the global memory. So, in Static Local Block Data Modeling, a GPU core reads all local sample data from its own shared memory while in Dynamic Local Block Data Modeling it reads some of the local sample data from its own shared memory and the other local sample data from the global memory. The measurements of GPU Runtime for both methods are shown in Table 5.

Table 5. Comparing Runtime of Static Local Block Data Modeling and Dynamic Local Block Data Modeling.

Grid Size	Static Local Block Data Modeling Method Runtime	Dynamic Local Block Data Modeling Method Runtime
20*20*20	39.724	75.963
30*30*30	41.823	93.517
40*40*40	48.132	99.341
50*50*50	73.213	108.324
60*60*60	91.772	116.648
70*70*70	130.341	194.425
80*80*80	208.321	380.175
90*90*90	250.324	530.934
100*100*100	367.512	714.532
110*110*110	460.425	898.353
120*120*120	586.339	1059.693

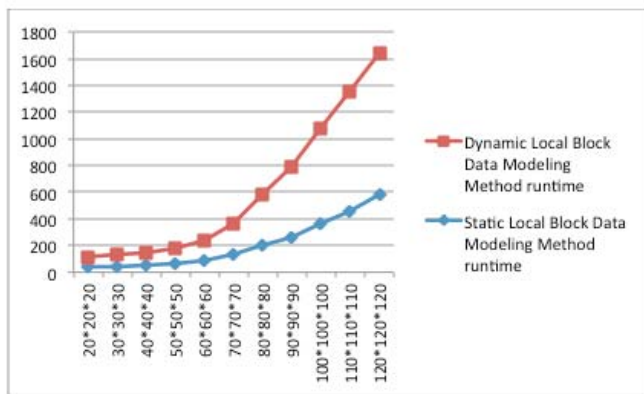


Figure 3. Comparing Runtime of Static Local Block Data Modeling and Dynamic Local Block Data Modeling.

We can see that Dynamic Local Block Data Modeling Method consumes more time than the Static Local Block Data Modeling Method, almost doubling the runtime length for the same grid size. This is due to the Static Local Block Data Modeling method reading all needed sample data from faster shared memory while Dynamic Local Block Data Modeling method reads needed sample data from both faster shared memory and slower global memory.

4 Conclusions

We have built a GPU accelerated scattered data modeling system and conducted a case study with a real-world dataset using the system. For comparisons, we also built a CPU based system with the same modeling function. We measured the performance of the systems with various modeling parameters. The results show how modeling parameters affect the performance of such modeling systems. The results also reveal the bottlenecks of such systems and reveal the areas where further researches are needed.

- 1) Adjusting internal modeling parameters belonging to the modeling function, such as α in the Shepard method, has no impact on computing time.

- 2) Time needed to model the input sample data onto the intermediate grid is linear compared to the total number of grid nodes in the intermediate grid, i.e., the size of the intermediate grid ($n_x \times n_y \times n_z$). The linear dependency is not clear when the size is small (less than $32 \times 32 \times 32$ for the test case.)
- 3) Speedup by the GPU as compared to the CPU increases as the grid size increases. The speedup reached 27 with grid size of $128 \times 128 \times 128$ for the test case using an NVidia GeForce GT 525M GPU.
- 4) Efficiency of GPU utilization also increases as the grid size increases. It reached 0.2764 with a grid size of $128 \times 128 \times 128$ for the test case using an NVidia GeForce GT 525M GPU. This is still far from the theoretical limit of 1.0. There is room for our GPU program to improve.
- 5) Data communication between the GPU and the host is a non-computational overhead that impacts the efficiency of GPU utilization. However, the ratio of Data Communication Time over Kernel Compute Runtime decreases as grid size increases. In our test case, the ratio changed from 8.818 for grid size $32 \times 32 \times 32$ to 0.140 for grid size $128 \times 128 \times 128$. Thus, we need to find other ways to improve GPU utilization at large grid sizes in addition to reducing data communication overhead.
- 6) Among the two Block-Oriented Localized Data Modeling methods, the Dynamic Local Block Data Modeling method consumes more time than the Static Local Block Data Modeling method. This is due to the Static Local Block Data Modeling Method reading all needed sample data from faster shared memory while the Dynamic Local Block Data Modeling method reads from both faster shared memory and slower global memory.

In future studies we plan to investigate GPU enhancement of other interpolation methods, such as volume spline, thin-plate-spline and multi-quadrics, in scattered data modeling. Another important area that needs attention is the accuracy of scattered data modeling, i.e., how accurately the intermediate grid can represent the original sample data.

5 References

- [1] W. Chen, F. Guo and F. Wang. Survey of traffic data visualization. *IEEE Transactions on Intelligent Transportation Systems*, 16(6), June 2015, 2970 – 2984, DOI: 10.1109/TITS.2015.2436897.
- [2] A. Ledesma, M. Al-Musawi and H. Nieminen. Health figures: an open source JavaScript library for health data

visualization. *BMC Medical Informatics & Decision Making*. Published online on 03/22/2016, <http://bmcmmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-016-0275-6>
DOI: 10.1186/s12911-016-0275-6.

[3] Z. Jiang. GIXSGUI: a MATLAB toolbox for grazing-incidence X-ray scattering data visualization and reduction, and indexing of buried three-dimensional periodic nanostructured films. *Journal of Applied Crystallography*, **48**, 917-926, 2015, [doi:10.1107/S1600576715004434](https://doi.org/10.1107/S1600576715004434).

[4] R. Liu, Y. Ge, P. A. Holden and Y. Cohen. Analysis of soil bacteria susceptibility to manufactured nanoparticles via data visualization. *Beilstein J. Nanotechnol.* **6**, 2015, 1635–1651.

[5] X. Liua, Y. Song, K. Wu, J. Wang, D. Lie and Y. Long. Understanding urban China with open data. *Cities*, **47**, September 2015, 53–61.

[6] L. Contreras, C. I. Font, P. Morillo and D. Vallejo. DataWarrior: An open-source program for chemistry aware data visualization and analysis. *J. Chem. Inf. Model.* January 2015, **55**(2), 460–473, DOI: 10.1021/ci500588j.

[7] M. Zhang, J. Yao, S. Wang and S. Zhu. Ocean surface approximation from scattered numerical marine forecast data. *Proceedings of IEEE International Conference on Information and Automation*, August 2015, 755 – 760.

[8] A. Eldawy, M. F. Mokbel and C. Jonathan. HadoopViz: A MapReduce framework for extensible visualization of big spatial data. *Proceedings of IEEE International Conference on Data Engineering*, Helsinki, Finland, May 16 to 20, 2016.

[9] A. T. Foley and A. D. Lane. Visualization of irregular multivariate data. *Proceedings of the First IEEE Conference on Visualization*, San Francisco, CA, 1990, 247-254.

[10] W. E. Lorensen and H. E. Cline. Marching Cubes: A high resolution 3D surface reconstruction algorithm. *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, 1987.

[11] G. M. Nielson and J. Tvedt. Comparing methods of interpolation for scattered volumetric data. In R. A. Earnshaw and D. F. Rogers (Eds.), *State of the Art in Computer Graphics*, Springer, 1993, 67-86.

[12] M. Lai and C. Meile. Scattered data interpolation with nonnegative preservation using bivariate splines and its application. *Computer Aided Geometric Design*, **34**, March 2015, 37–49.

[13] G. J. Streletza, G. Gebbieb, O. Kreylosa, B. Hamanna, L. H. Kellogg and H. J. Speroc. Interpolating sparse scattered data using flow information. *Journal of*

Computational Science, in press. Available online 30 April 2016, Elsevier.

[14] G. R. Joldes, H. A. Chowdhury, A. Wittek, B. Doyle and K. Miller. Modified moving least squares with polynomial bases for scattered data approximation. *Applied Mathematics and Computation* **266**(1), September 2015, 893–902.

[15] Y. Xiao and J. Ziebarth. FEM-based scattered data modeling and visualization. *Computers and Graphics*, **24**(5), 2000, 775-789.

[16] Y. Xiao and C. Woodbury. Constraining global interpolation methods for sparse data volume visualization. *International Journal of Computers and Applications*, **21**(2), 1999, 56-64.

[17] D. Kirk and W. Hwu. *Programming Massively Parallel Processors: A Hands-on Approach*. MK Publications, 2013.

[18] D. Weiskopf. *GPU-Based Interactive Visualization Techniques*. Springer, 2007.

[19] NVIDIA, CUDA C – Programming Guide. <http://www.geforce.com/hardware/technology/cuda>

[20] G. M. Nielson. Scattered data modeling. *IEEE Computer Graphics & Applications*, **13**(1), 1993, 60-70.

[21] D. Shepard. A two-dimensional interpolation function for irregularly spaced data. *Proceedings of ACM National Conference*, 1968, 517-524.

[22] W. Schroeder, K. Martin, and B. Lorensen. *The visualization toolkit - an object-oriented approach to 3D graphics*, Kitware Publishing, 2006.

[23] Y. Xiao, J. P. Ziebarth, C. Woodbury, E. Bayer, B. Rundell and J. van der Zijp. The challenges of visualizing and modeling environmental data. *IEEE Visualization 96 Conference Proceedings*, San Francisco, California, October 27 – November 1, 1996, 413- 416.

[24] Y. Xiao, J. Tian and H. Sun. Error analysis in sparse data volume visualization. *Proceedings of International Conference on Imaging Science, Systems, and Technology*, Las Vegas, June 24-27, 2002, 813-818.

[25] NVidia, *Visual Profiler*, <https://developer.nvidia.com/nvidia-visual-profiler>

Visualizing Competence Models and Individual Learning Paths

Michael D. Kickmeier-Rust
Graz University of Technology
Knowledge Technologies Institute
8010 Graz, Austria
+43 316 873 30636
michael.kickmeier-rust@tugraz.at

Dietrich Albert
Graz University of Technology
Knowledge Technologies Institute
8010 Graz, Austria
+43 316 873 30640
dietrich.albert@tugraz.at

ABSTRACT

Learning analytics means gathering a broad range of data, bringing the various sources together, and analyzing them. However, to draw educational insights from the results of the analyses, these results must be visualized and presented to the educators and learners. This task is often accomplished by using dashboards equipped with conventional and often simple visualizations such as bar charts or traffic lights. In this paper we want to introduce a method for utilizing the strengths of directed graphs, namely Hasse diagrams, and a competence-oriented approach of structuring knowledge and learning domains. After a brief theoretical introduction, this paper highlights and discusses potential advantages and gives an outlook to recent challenges for research.

Keywords

Learning analytics, data visualization, Hasse diagram, Competence-based Knowledge Space Theory.

1. INTRODUCTION

Using methods and tools from Learning Analytics (LA) can be considered best practice and is a key factor for making education more personalized, adaptive, and effective. Analyzing a variety of available data to uncover learning processes, strengths and weaknesses, competence gaps undoubtedly is a prerequisite for a formatively-inspired guidance, for changing and adjusting educational measures and teaching, and not least for disclosing and negotiating learner models [4]. Usually, the benefits are seen in the potential to reduce attrition through early risk identification, improve learning performance and achievement levels, enable a more effective use of teaching time, and improve learning design and instructional design [10]. On the basis of available data, ideally large scale data sets, smart tools and systems are being developed to provide teachers with effective, intuitive, and easy to understand aggregations of data and the related visualizations. There is a substantial amount of work going on this particular field; visualization techniques and dashboards are broadly available (cf. [2,4,7]), ranging from simple meter/gauge-based techniques (e.g., in form of traffic lights, smiley, or bar charts) to more sophisticated activity and network illustrations (e.g., radar charts or hyperbolic network trees).

However, LA operates in a delicate and complex area. On the one hand, facing today's classroom realities, we often find technology-lean environments, which do not easily allow or support recording the necessary data. Also, from a socio-pedagogical perspective, learning must be seen as a process of social interaction that not always occurs in front of some electronic. Thus, LA must be based on fewer data. On the other

hand, it is rather easy to visualize learning on a superficial level using perhaps the aforementioned traffic lights or bar charts. The added value to the teachers is likely of limited utility to them. To provide a deeper and more formative insight into the learning history and the current state of a learner (beyond the degree to which a teacher might know it intuitively) requires finding and presenting complex data aggregations. This, most often, bears the significant downside that it is hard to understand. Challenges for LA and its visualizations, for example, are to illustrate learning progress (including learning paths) and - beyond the retrospective view - to display the next meaningful learning steps/topics.

In this paper we introduce the method of directed graphs, the so-called Hasse diagrams, for structuring learning domains and for visualizing the progress of a learner through this domain.

2. HASSE DIAGRAMS AND COMPETENCE-BASED KNOWLEDGE SPACES

A Hasse diagram is a strict mathematical representation of a so-called semi-order in form of a directed graph that reads from bottom to top. A semi-order is a type of mathematical ordering of a set of items with numerical values by identifying two items as equal or comparable if the values are within a given interval of error or noise. Semi-orders were introduced in mathematical psychology by Duncan Luce in 1956 [8] in human decision research without the assumption that indifference is transitive. This approach is also crucial for handling human learning and the resulting performance that is prone to all sorts of errors and peripheral aspects (perhaps failing in a test although the learner holds the knowledge due to being tired). A Hasse diagram is one way of displaying such ordering – in our case competences or competency states (which is to be explained in the following section). The technique was invented in the 60s of the last century by Helmut Hasse. The diagram exists of entities (the nodes), which are connected by relationships (indicated by edges).

The mathematical properties of a semi-order and the Hasse diagrams are (i) reflexivity, (ii) anti-symmetry, and (iii) transitivity. Reflexivity refers to the view that an item, perhaps a competency, references itself in a cause/effect sense. Anti-symmetry demands that if one entity is a prerequisite of another, this relationship is not invertible; as an example, if competency x is a prerequisite to develop competency y , y cannot be the prerequisite of competency x . Finally, transitivity means that whenever an element x is related to an element y , and y is in turn related to an element z , then x is also related to z . In principle, the direction of a graph is given by arrows of the edges; by convention however, the representation is simplified by avoiding the arrow heads, whereby the direction reads from bottom to top. In addition, the arrows from one element to itself (reflexivity

property), as well as all arrows indicating transitivity are not shown in Hasse diagrams. The following image (Figure 1) illustrates such a diagram. Hasse diagrams enable a complete view to (often huge) structures. Insofar, they appear to be ideal for capturing the large competence or learning spaces occurring in the context of assessment and learning recommendations (for example, all the competencies involved in the math curriculum for a specific age).

In an educational context, a Hasse diagram can display the non-linear path through a learning domain starting from an origin at the beginning of an educational episode (which may be a single school lesson but could also be the entire semester). Moreover, the elements in the diagram may refer to (latent) competencies, to learning objects or test items. Figure 1 illustrates the simple example of typical learning objects in a certain domain. The beginning of a learning episode is usually shown as $\{ \}$ (the empty set) at the bottom of the diagram. Now a learner might attend three learning objects (K, P, H), which is indicated by the edges; this, in essence, establishes three possible learning paths. After H, as an example, this learner might attend K, or H but not T yet, which in turn opens further three branches for the learning path until reaching the final state, within which all learning objects have been attended.

As claimed initially, in the context of formative LA, a competence-oriented approach is necessary. Thus, a Hasse diagram can be used to identify and display the latent competencies of a learner in the form of so-called competence states. An elaborated theoretical approach to do so is Competence-based Knowledge Space Theory (CbKST). The approach originates from Jean-Paul Doignon and Jean-Claude Falmagne [5, 6] and is a mathematical psychological, set-theoretic framework for addressing the relations among problems (e.g., test items). It provides a basis for structuring a domain of knowledge and for representing the knowledge based on prerequisite relations. While the original Knowledge Space Theory focuses only on performance (the behavior; for example, solving a test item), its extension CbKST [1] introduces a separation of observable performance and latent, unobservable competencies, which determine the performance [1]. This is a psychological learning-theoretical approach, which highlights that competencies (e.g., the ability to add two integers) are unobservable latent constructs and which can only be observed or assessed indirectly.

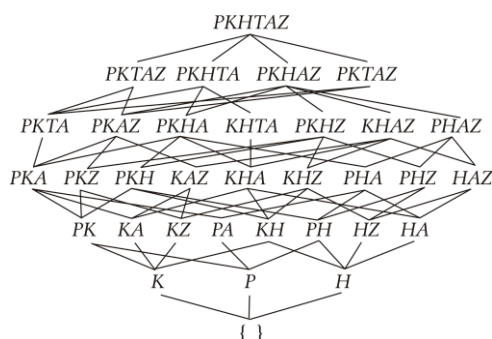


Figure 1. A simple Hasse diagram.

We interpret the performance of a learner (e.g., mastering an addition task) in terms of holding or not holding the respective

competency. In addition, recent developments of the approach are based on a probabilistic view of having or lacking certain competencies. In our example, mastering one specific addition task allows the conclusion that the person is able to add two numbers (to hold this competency) only to a certain degree or probability. When thinking of a multiple-choice item with two alternatives, as another example, mastering this item allows only to 50 percent that the person has the required competencies/knowledge.

On the basis of these fundamental views, CbKST is looking for the involved entities of aptitude (the competencies) and a natural structure, a natural course of learning in a given domain. For example, it is reasonable to start with the basics (e.g., the competency to add numbers) and increasingly advance in the learning domain (to subtraction, multiplication, division, etc.). As indicated above, this natural course is not necessary linear, which bears significant advantages over other learning and test theories.

As a result we have a set of competencies in a domain and potential relationships between them. In terms of learning, the relationships define the course of learning and thus which competencies are learned before others. In CbKST such relationships are called prerequisite relations or precedence relations. On the basis of competencies and relationships, in a next step, we can obtain a so-called competence space, the ordered set of all meaningful competence states a learner can be in. As an example, a learner might have none of the competencies, or might be able to add and subtract numbers; other states, in turn, are not included in this space, for example it is not reasonable to assume that a learner holds the competency to multiply numbers but not to add them. By the logic of CbKST, each learner is, with certain likelihood, in one of the competence states.

3. VISUALIZING COMPETENCE SPACES

As claimed, Hasse diagrams are capable of holding a number of important information for an educator to evaluate the learning progress and also to make recommendations. In this paper we want to highlight such advantages.

3.1 Competence States and Levels

As outlined, a competency space is the collection of meaningful states a learner can be in. Depending on the domain, the amount of possible states might be huge. The big advantage, however, is that depending on the degree of structure in the domain, by far not all possible combinations of competencies are reasonable and thus part of the space. When zooming into the diagram, a teacher can exactly identify the set of competencies that is most likely for the learner, by zooming out color-coding can illustrate the most likely locations of a learner within the space. When looking at the entire space, it is obvious at first site at which completion level a learner is approximately (rather at the beginning or almost finished). These zoom levels are shown in Figure 2. Technically, there is a variety of options to achieve the coding, for example, bolding, greying, or color coding, whereas likely states are displayed more distinctly than such with low probability.

Equal to individual states, Hasse diagrams can represent group distributions. Defined by a certain confidence interval of probabilities those states and areas can be made more salient that hold the highest percentage of learners of a group. By this means,

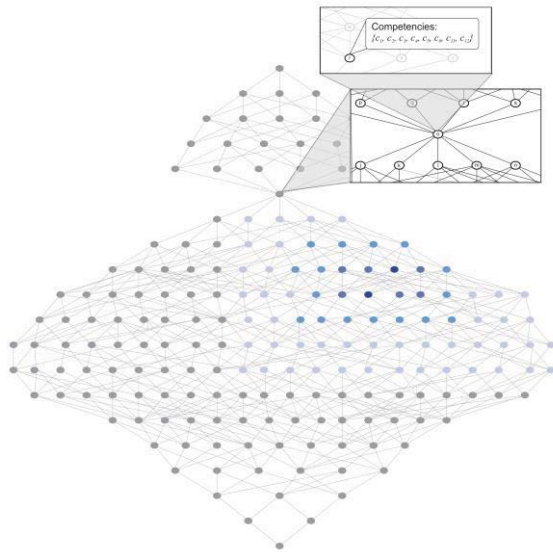


Figure 2. Hasse diagram illustrating the probability distribution over a competence space on three zoom levels.

specific areas in the competence space become apparent within which the most learners are and, in contrast also positive or negative outliers pop out the diagram. A different method was suggested by [9], who altered the size of the nodes to represent the groups' sizes; the larger a node the more learners hold a particular state.

3.2 Learning Paths

In addition to having insight into groups' and individuals' current states of learning, the learning history, the so-called learning paths, are of interested for educators; on the one hand for planning future activities, on the other hand, for negotiation and documenting the achievements of a learning episode (e.g., a semester). Learning paths can be simply displayed by highlighting the edges between the most likely state(s) over time. As for the states, various probable paths can be realized by making more likely paths more intensive (by color coding or line thickness). Figure 3 shows a simple example. A key strength of presenting learning paths, as indicated, is opening up the learner model to the learners (perhaps parents) themselves [9] – to explain where they started at the beginning of a course and how they proceeded

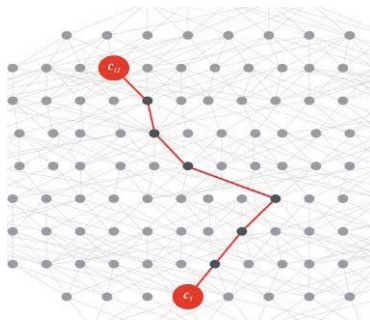


Figure 3. Learning Path. The cutout is part of the structure shown in Figure 2.

during the course and which competencies they hold today. This perhaps can be complemented with comparisons to others or groups. Not least, learning paths can unveil information about the effectiveness and impact of certain learning activities, materials, or the teacher herself.

3.3 Tests and Recommendations

Hasse diagram offers information about two very distinct concepts, the inner and outer fringes. The inner fringe indicates what a learner can do / knows at the moment. Mathematically it refers to all sets of competencies, which hold all competencies of the current state but one. This inner fringe is a clear hypothesis of which test/assessment items this learner can master within the margins of a certain probability. Such information may be used to generate effective and individualized tests. The test generation can be complemented with group information. If an educator has very clear information in which competency areas of the space most of the learners are, she can generate or select test item covering exactly those competencies. The big advantage of such approach is the effectiveness of a test for identifying competency states or for ranking the learners can be maximized while the efforts for this evaluation (e.g., the number of test items) can be minimized. And of course the test can be optimized to differentiate different learners and the individual capabilities.

On the other hand, the outer fringes determine which competencies should be addressed in a next educational step. Mathematically it refers to all states which include all the competencies of the current state plus one. These fringes provide a clear set of recommendations about the most effective learning activities for a specific individual or a specific group of learners. Moreover, outer fringes, together with learning paths, allow specifically planning the most effective ways of reaching a specific learning goal (which not necessarily is the final stage of the competence space, the full set, and which is not necessarily the same goal for all individual learners).

3.4 Costs and Pace

When supporting teachers with information about learning processes, the concept of costs or learning pace (sometimes referred to as learning trajectories) is of distinct importance. Cost and pace can be considered as the time or any other measure of effort it takes to proceed from one competence state to another. In a Hasse diagram this information can be displayed by varying the length of the edges accordingly. If an educational leap requires a lot of efforts or time the edges are displayed proportionally longer than such that happens rather quickly. This method was introduced initially by [9]; an example is shown in Figure 4. Such information unveils criteria for the effectiveness of certain learning materials or acts of teaching. Particular outliers obviously pop out of the diagram and call educators to action to adapt teaching or teaching materials for a specific individual or a group.

3.5 Subordinate Concepts and General Notions of Achievement, Bottlenecks

A further important aspect in the context of LA is aligning the rather fine grained and low level approach to view competencies on a deeper level of granularity to more general concepts or rather superordinate notions of achievement. A general concept can be considered a higher level cluster of competencies; for example, sub-dividing mathematics into clusters like linear equations, non-linear equations, and vector arithmetic. Lower level competencies can be linked to one or more of those 'chapters'. Equally, one

might view learning processes in a domain in terms of maturity. For example, writing skills can be on a low level of maturity, involving certain competencies and abilities, and on a higher one. Such approach is given, for example, in the CEFR language skills (cf. http://en.wikipedia.org/wiki/Common_European_Framework_of_Reference_for_Languages). Finally, teaching might involve the achievement of certain milestones, which should be reached step by step. Hasse diagrams allow identifying such milestones even if they were unclear or unknown initially. Considering that milestones as bottlenecks, i.e. unique competence states, each learning must pass, such bottlenecks immediately pop out in of the diagram. In a formative sense, it is easy for an educator to located their learners in their approach to or exceeding of such milestones (cf. Figure 2). A slightly different variant was introduced by [9] who used additional graphical elements (e.g., intersecting lines) to separate certain levels of maturity (whereas these authors used the CMMI¹ method; cf. Figure 5).

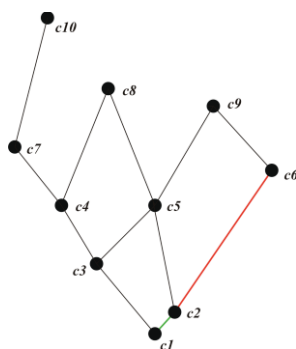


Figure 4. Illustrating learning efforts (as costs or pace). The longer the more efforts/time it took to acquire a further competency.

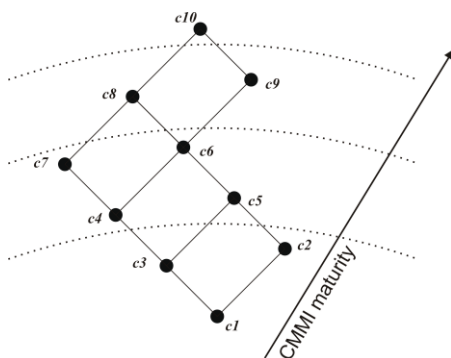


Figure 5. Illustrating maturity levels.

¹ CMMI refers to the so-called *Capability Maturity Model Integration* approach which models development processes (e.g., in production) on different predefined levels [3].

4. WHERE DO DATA COME FROM?

The features of Hasse diagrams and the arising advantages for LA appear all well and good. However, the key question is, where do they data for computing the probabilities of competence states come from. And everything stands or falls with this question. As for all techniques of LA, it depends on a data rich approach to education, the more and the better data exist, the better is the quality of LA conclusions. CbKST and Hasse diagrams are no exception to that. However, the approach of separating latent competencies, which more or less develop and exist in the black box 'human brain', and the performance they determine, bears particular advantages. On the one hand, performance, e.g. test scores, classroom participation, homework, etc., is not only determined by competencies or aptitude; there is a variety of aspects contributing to a certain performance, e.g., motivation, daily constitution, tiredness, external distractors, nutrition, health status, etc. On the other hand, CbKST-ish competence spaces are rather stable, once set up and validated properly. The advantage lays in the fact that performance such as test results, behaviors, achievements, etc. is considered as probability-based indicators for certain competencies. Mathematically this relationship is established in form of interpretation and representation functions [1], which links an arbitrary set of performances/behaviors to one or more competencies, either in an increasing or in a decreasing sense. This, in the end, allows linking all available and perhaps changing data sources to one and the same competence space. It's not about a single test, it's about all available information we can gather, even it is considered being of little importance, all sorts of information may contribute to strengthen the model, the view of the learner. In case the amount or quality of data is weak, CbKST allows conservative interpretations, based on the arising probability distributions, in case there is a richer data basis, the probability distributions are more reliable, valid, and robust. For the educator, and this is important, the uncertainty is mirrored in the degree of likelihood. On a weak data basis, the probabilities of competence states differ substantially less than on the basis of richer data. Such information, however, can change the educator's view and evaluation of a student's achievements. In the end, this approach supports a fairer and more substantiated approach to grading or providing formatively inspired feedback.

5. CONCLUSIONS AND OUTLOOK

There is little doubt that frameworks, techniques, and tools for LA will increasingly be part of a teacher's professional life in the near future. The benefits are convincing – using the (partly massive) amount of available data from the students in a smart, automated, and effective way, supported by intelligent systems in order to have all the relevant information available just in time and at first sight. The ultimate goal is to formatively evaluate individual achievements and competencies and provide the learners with the best possible individual support and teaching. Great. The idea of formative assessment and educational data mining is not new but the hype over recent years resulted in scientific sound and robust approaches becoming available, and usable software products appeared. However, when surveying the educational landscape, at least that of the EU, the educational daily routines are different. We face technology-lean classrooms and schools, we face a lack of proper teacher education in using ICT in schools – not mentioning of using techniques of LA in schools. We face a certain aloofness to use breaking educational technologies and a well-founded pedagogical view that learning ideally is analogous and socially embedded and doesn't occur in front of some kind of

electronic device. These are all experiences and results of a large scale European research project named Next-Tell (www.next-tell.eu) that was looking into educationally practices across Europe and that intended to support teachers where exactly they are today with suitable ICT as effective and as appropriately as possible.

The framework of CbKST offers a rigorously competence-based, probabilistic, and multi-source approach that accounts for the latent and holistic abilities of learners and therefore accounts for the recent conceptual change in Europe's educational systems towards a more competence-oriented education including multi-subject competencies and superordinate 21st century (soft) skills.

No matter if data are rich or lean, a teacher is supported to the best possible degree and with a variety of important information about individual and group-based learning processes and performances and not least about the performance of learners and about the educator's own performance. The probabilistic dimension allows teachers to have a more cautious view of individual achievements – it might well be that a learner has a competency but fails in a test; vice versa, a student might luckily guess an answer.

From an application perspective, in the context of European projects we developed and evaluated tools that cover the techniques and approaches described in this paper. In the Next-Tell project, for example, we developed a software tool named ProNIFA, which allowed linking multiple sources of evidence of learning and building CbKST-based learner models. We piloted various school studies and gathered feedback from teachers. In the end, and this can be considered an outlook for future developments, we had to find out that the 'massive' Hasse diagrams are overburdening teachers' understanding and mental models about individual and class-based learning. Moreover, in order to understand the classical Hasse diagrams, it required (too) massive efforts in training teachers to fully utilize the potentials of those diagrams. Large scale surveys yielded that most educators still prefer simple but information-wise shallow visualizations such as traffic lights or bar charts significantly over more information-rich approaches such as Hasse diagrams or, just to mention another interesting approach, parallel coordinates.

Therefore, recent efforts, e.g., in the LEA's BOX (www.lea-box.eu) project, seek to adjust and advance the classical Hasse diagrams to such visualizations that are intuitively understood by educators and, at the same time, hold the same density of information. In particular, focus of research is on an advancement of Hasse diagrams towards specific mental models teachers may hold, such as a starry night sky or organic, biological structures such as cells of a living being. Also, abstraction and simplification techniques are investigated, e.g., fisheye lenses or streamgraphs.

In conclusion, the utility of CbKST-ish approaches to LA, involving a separation of latent competencies and observable behaviors/performance, as well as having a conservative, probabilistic, multi-source approach appears to be a striking classroom-oriented, next-level contribution to LA, learner modelling, and model negotiations.

6. ACKNOWLEDGMENTS

This work is based on the finalized project Next-Tell, which was supported by the European Commission (EC) under the Information Society Technology priority of the 7th Framework Programme for research and development as well as the running LEA's BOX project, contracted under number 619762, of the 7th Framework Programme. This document does not represent the opinion of the EC and the EC is not responsible for any use that might be made of its content.

7. REFERENCES

- [1] Albert, D., & Lukas, J. 1999. Knowledge Spaces: Theories, Empirical Research, and Applications. Mahwah, NJ: Lawrence Erlbaum Associates.
- [2] Ferguson, R., and Buckingham Shum, S. 2012. Social Learning Analytics: Five Approaches. In Proceedings of the 2nd International Conference on Learning Analytics & Knowledge, 29 Apr - 02 May 2012, Vancouver, British Columbia, Canada.
- [3] Forrester, E. C., Buteau, B. L., and Shrum, S. 2009. CMMI for Services. Guidelines for Superior Service. Addison-Wesley.
- [4] Dimitrova, V., McCalla, G. and Bull, S. 2007. Open Learner Models: Future Research Directions (Special Issue of IJAIED Part 2), International Journal of Artificial Intelligence in Education 17(3), 217-226.
- [5] Doignon, J., & Falmagne, J. 1985. Spaces for the assessment of knowledge. International Journal of Man-Machine Studies, 23, 175-196.
- [6] Doignon, J., & Falmagne, J. 1999. Knowledge Spaces. Berlin: Springer.
- [7] Duval, E., 2011. Attention Please! Learning Analytics for Visualization and Re-recommendation. In Proceedings of the 1st International Conference on Learning Analytics & Knowledge, 27 Feb - 1 March 2011, Banff, Alberta, Canada.
- [8] Luce, R. D. 1956. Semiorders and a theory of utility discrimination. Econometrica, 24, 178-191.
- [9] Nakamura, Y., Tsuji, H., Seta, K., Hashimoto, K., and Albert, D. 2011. Visualization of Learner's State and Learning Paths with Knowledge Structures. In A. König et al. (Eds.), KES 2011, Part IV. Lecture Notes in Artificial Intelligence 6884, pp. 261-270. Berlin: Springer.
- [10] Siemens, G., Gasevic, D., Haythornthwaite, C., Dawson, S., Buckingham Shum, S., Ferguson, R., Duval, E., Verbert, K., and Baker, R.S.J.D. 2011. Open Learning Analytics: an integrated & modularized platform: Proposal to design, implement and evaluate an open platform to integrate heterogeneous learning analytics techniques. Available online at <http://solaresearch.org/OpenLearningAnalytics.pdf>