SESSION

BIOINFORMATICS AND RELATED ISSUES

Chair(s)

TBA

Shannon's Entropy of The Stochastic Context-Free Grammar and an Application to RNA Secondary Structure Modeling

Amirhossein Manzourolajdad
National Center for Biotechnology Information (NCBI), NIH, Bethesda, USA

Abstract— *Stochastic context-free grammars (SCFG) have been used in RNA Secondary structure modeling. An SCFG consists of a set of grammar rules with probability for each. Given a grammar design, finding the best set of probabilities that yield optimum performance can be challenging. Although current Expectation Maximization (EM) Maximum-Likelihood (ML)-based model training approaches have been effective, there is no guarantee that they provide parameter sets for the grammar to have optimum performance. In this work, An analytical measure of the SCFG space, denoted here as Grammar Space (GS) entropy, is introduced and calculated for various SCFG models in the literature. It is shown that more accurate models have lower GS entropy. Finally, based on the GS entropy, a novel RNA structure model training method is proposed.*

Keywords: SCFG, RNA secondary structure, Shannon's entropy

1. Introduction

[1] initially formalized Context-Free Grammars (CFG) as an attempt to model languages. Since then, they have been used broadly in a variety of computer science applications. CFGs have a recursive nature and are composed of sets of rules that can drive strings of alphabets, also referred to as words. Words generated by the grammar is referred to as the language of that grammar. A grammar is said to be unambiguous if there is a one-to-one correspondence between the words and the derivations. Stochastic CFGs (SCFG) assign probabilities to each rule, which in turn assigns a likelihood value to each word by multiplying the probabilities of the rules used to derive that word [2], [3]. [4], [5], [6], [7] and others have investigated important properties of CFGs and SCFGs, such as convergence of average length of derivations, capacity, or the Shannon's entropy [8], using generating-functions techniques. RNA secondary structure was first formally described by [9]. Combinatorial features of RNA secondary structures were then explored by [10] and subsequently studied by [11], [12], [13], [14], [15] and others using generating functions and probability-generating functions.

Stochastic Context-free Grammars have had great impact in RNA secondary structure studies [16], [17], [18], [19]. RNA secondary structure modeling helps us understand structural functions of a group of non-protein-coding RNAs (ncRNA). Typically, RNA secondary structure prediction is done by minimization of the folding energy under a thermodynamic-based folding model such as the Boltzmann ensemble [20]. An SCFG assigns likelihood values to all possible folding scenarios of an RNA sequence. The structure having Maximum Likelihood (ML) can then be found by the Cocke-Younger-Kasami (CYK) algorithm. Similar to energy minimization, the CYK algorithm is implemented through dynamic programming. The goal of model design is to obtain an ML estimation that is as similar as possible to a known RNA secondary structure, thus making assignment of probabilities to rules a critical and challenging task. In this work, the Shannon's entropy of the SCFG, denoted here as grammar space (GS) entropy, is analytically calculated and introduced as a potential grammar feature of RNA secondary structure modeling. Presented formulations are consistent with the general form of grammars entropy known as *derivational* entropy, and can be found in [4] and [7].

2. A Shannon's Entropy for the SCFG Space

As previously mentioned, any SCFG refers to an infinitely large ensemble of derivations that can be generated by applying the grammar rules in a recursive fashion. In the case of unambiguous grammars, such derivations are unique and discrete random events each having their associated probability. Here, an attempt is made to compute the closed form Shannon's entropy of the infinitely large probabilistic space that contains all possible derivations emanating from a given grammar. Our focus here is structurally unambiguous grammars that are designed and trained for RNA secondary structure prediction.

2.1 An Intuitive Example

Stochastic context-free grammars consist of terminals and non-terminals. Terminals denote string alphabet, while non-terminals denote the a new substructure. Consider the following unambiguous grammar, denoted as Ex1, with a single alphabet, $\Sigma = \{a\}$, and the following two production rules:

$$S \to a \ \ (\alpha), S \to aS \ \ (\beta)$$

where α and β are terminal and non-terminal rule probabilities, respectively, and $\alpha + \beta = 1$. Ex1 produces an infinite

number of strings s, $s \in \Sigma^*$, each of which corresponding to a unique derivation or parsing. Being exclusive events, the sum of all possible derivations converges to one in the limit of infinity, $\sum_{i=0}^{\infty} \alpha\beta^i = 1$, while their expected log-likelihood converges to the following:

$$\lim_{|\Sigma^*_{Ex1}|\to\infty} E\left[\log \Sigma^*_{Ex1}\right] = -\sum_{i=0}^{\infty} \alpha\beta^i \log\left(\alpha\beta^i\right) = -\log\alpha - (\beta/\alpha)\log\beta \quad (1)$$

where the random variable Σ^*_{Ex1} refers to the space of derivations of grammar Ex1, and $E[X]$ denotes the expectation of random variable X; $E[X] = \sum_{x\in X} p(x) \times x$.

The derived expression in (1) is *the* Shannon's entropy of Ex1 grammar space in the limit of infinitely large number of derivations, since any ordering of the geometric summation in (1) results in the same value. This is because the summation here is absolutely convergent[1]. The Shannon's entropy for Ex1 is defined for all terminal probability values α within the interval $(0, 1]$. For the case of $\alpha = 0$, however, the entropy of Ex1 is not defined but tends towards infinity as α tends towards zero. This is in line with the intuition that as the probability of selecting a terminal, α, decreases, the expected length (i. e. log-likelihood) of derived strings increases.

2.2 Generalization to All Structurally Unambiguous Grammars

In this section, a general method is presented for computing the GS entropy of unambiguous SCFGs regardless of number of non-terminals, rules, and probability distributions of the rules. In doing so, various axioms applicable to the Shannon's entropy were used to avoid analytical complications.

Let us use notation $H(\Pi Y|n, G, \Theta)$ for the Shannon's entropy of derivations emanating from non-terminal n in grammar space (G, Θ) in the limit of infinite number of derivations:

$$H(\Pi Y|n, G, \Theta) \equiv \lim_{|\Pi Y_{(n,G,\Theta)}|\to\infty} E\left[\log \Pi Y|n, G, \Theta\right]$$

where $\Pi Y_{(n,G,\Theta)}$ is the structural space of all derivations $(\pi, y) \in \Pi Y_{(n,G,\Theta)}$, each having probability $p(\pi, y|n, G, \Theta) = p(n \Rightarrow^*_\pi y)$. Expression n is the starting non-terminal for the generation of (π, y), y is the derived sequence, π denotes the derivation tree that produces y, and operator $\Rightarrow^*_\pi$ describes the derivation tree π. Expression $H(\Pi Y|n)$ is used instead of $H(\Pi Y|n, G, \Theta)$ from hereon for convenience, since G and Θ are constant throughout the GS entropy calculation. The total probability of derivation trees emanating from nonterminal n, $p(\Pi Y|n)$, can be expressed in terms of other non-terminals of the grammar by the summing over the probabilities of all grammar rules that emanate from non-terminal n:

$$P(\Pi Y|n) = \sum_{i\in n\to\omega} p_i \times P(\Pi Y|i_1) \times P(\Pi Y|i_2), \quad \forall n \in N \quad (2)$$

where $n \to \omega$ is any grammar rule with non-terminal n at its left-hand side, N is the number of non-terminals in the grammar, and p_i is the probability of rule i. Symbols i_1 and i_2 represent the first and second nonterminals on the right-hand side of rule i, respectively. Note, for the rules that have only one non-terminal on the right-hand side, $i_2 = \emptyset$, $P(\Pi Y|i_2)$ is set to one, and for rules having no non-terminals on their right-hand side, both $P(\Pi Y|i_1)$ and $P(\Pi Y|i_2)$ are set to one.

Expression $H(\Pi Y|n)$, can then be expressed in terms of the entropy other non-terminals by taking the entropy of the right side of (2):

$$H(\Pi Y|n) = H(\mathtt{P}_n) + \sum_{i\in n\to\omega} p_i[H(\Pi Y|i_1) + H(\Pi Y|i_2)], \forall n \in N \quad (3)$$

where $\mathtt{P}_n$ represents the corresponding probability vector of rules emanating from non-terminal n, $\mathtt{P}_n = \{p_i | i \in n \to \omega\}$, and $H(\mathtt{P}_n)$ is the Shannon's entropy of $\mathtt{P}_n$, since the sum of elements of $\mathtt{P}_n$ equals one: $H(\mathtt{P}_n) = -\sum_{i\in n\to\alpha} p_i \log p_i$. Note, for the rules that have only one non-terminal on the right side, $i_2 = \emptyset$, $H(\Pi Y|i_2)$ is set to zero and for rules having no non-terminals on their right-hand side, both $H(\Pi Y|i_1)$ and $H(\Pi Y|i_2)$ are set to zero.

As a result, the Shannon's entropy of non-terminals can be related to each other by N linear equations. The entropy of the SCFG space will then be equal to the entropy of the starting non-terminal S_0:

$$H(\Pi Y) = H(\Pi Y|S_0) \quad (4)$$

3. GS Entropy of RNA Folding Models

In this section, the GS entropy is calculated for various lightweight and heavyweight RNA folding models. The relationship between the GS entropy feature and model performance is then investigated. Lightweight RNA folding models were taken from [21]. The GS entropy figures of the four structurally unambiguous grammars were calculated. Computations were repeated for all three distinct choices of model parameters. The naming of the grammars G3, G4, G5, and G6 in the mentioned work are UNA, RUN, IVO, and BJK, respectively. Three RNA secondary-structure datasets by the names `mixed80`, `benchmark`, and `rfam5` were used to train each grammar model using the Conus software package [21]. Details about the grammars, the training datasets, software package, and the grammar rule probabilities are all available in [21]. The GS entropy values of the resulting twelve SCFGs were calculated according

[1]If the sum of the absolute value of summands of a series is finite, it is known as an absolutely convergent series. Absolutely convergent series converge to the same value regardless of order of summation.

to the presented procedure and are illustrated in Table 1. Symbol ∞ denotes cases where this is no convergance. All models trained via the `mixed80` dataset have higher entropy values than those of `benchmark` and `rfam5`. Also, all `benchmark`-trained models have higher entropy values than `rfam5`-trained models, regardless of choice of grammar.

The performance of an SCFG-based RNA folding model

Table 1: GS Entropy of Structurally unambiguous lightweight grammars under various parameter sets. Grammar description and parameters according to [21]. Column `Training Set` contains the name of the training set used to estimate model parameters. Logarithm was calculated in base 2.

Training Set	UNA	RUN	IVO	BJK
`mixed80`	∞	5448	5175	4198
`Benchmark`	767	907	982	732
`Rfam5`	467	533	598	452

is measured by its ability to predict the secondary structure of the given RNA sequence. The structure can be either observed in an experiment or inferred from structural homology. Sensitivity and specificity (or Predictive Positive Values (PPV)) are calculated by comparing base-pairs of the model prediction to those of the real structure. The performance is then assessed by a form of averaging individual sensitivity and specificity values across a collected test set. Various classes of RNAs have different folding features and functions, some with complicated structures difficult to predict given a simple model. Models trained on the `mixed80` dataset were, used in order to explore the possible relationship between the GS entropy of the model and its accuracy in predicting RNA secondary structure. The GS entropy of grammar models trained on the `mixed80` dataset were plotted against average sensitivity to structures in `Benchmark` and `Rfam5` test sets (Average sensitivity according to conus software [21]). GS entropy and corresponding average sensitivity values for the three BJK, RUN, and IVO grammars are shown in Figure 1 Top view. The UNA grammar was excluded, since its corresponding GS entropy was not defined as a real positive number (set to ∞). The BJK grammar has lower GS entropy while having higher average sensitivity values compared to the IVO and RUN grammars. Average Specificity values showed a similar pattern (data not shown).

An attempt was made to calculate the GS entropy of more recent grammars. [22] presented the TORNADO language with which heavyweight SCFGs can be described, enabling the user to parameterize various RNA structural features such as base-pair/coaxial stacking, dangles, helix/loop lengths, etc... The first grammar is denoted as g6, which is the same as the previously G6 or BJK model. The GS entropy of g6 was calculated using parameter sets provided by [22]. The second grammar that was selected was the `Basic Grammar` (BG). The rules of BG imitate a simplified version of a folding model that is at the the core of both the state-of-the-art thermodynamic model implemented in Vienna RNA [23] and complex nearest-neighbor RNA folding models used in [24]. (Please refer to [22] for further explanation)

Parameter sets for grammars g6 and BG are available for different training sets. Here, four training sets were considered. Names of training sets are shown in Table 2. Also, the TORNADO language enables applying different base-pairing constraints on the grammars. Two of these constraints were considered here. The first constraint describes a grammar that enforces Watson-Crick base-pairing, only. The second constraint, considers stacked base-pairs. In stacked base-pairing, the probability distribution of pairs varies depending on the context of surrounding pairing. Extensions `wcx` and `stk` were used to denote the first and second constraint on a given grammar, respectively. For instance, the stacking version of the BG grammar was denoted as BGstk. Parameter sets corresponding to different training sets and constraints were available in [22] and also provided through personal communication with Elena Rivas. GS entropy values are shown in Table 2. GS entropy values corresponding to the `TrA + 2 × TrB` training set are plotted against the best F measures of the models in Figure 1 Bottom view. Here, F measure is the harmonic mean of the sensitivity and PPV [25]. The best F measure is used as a measure of model accuracy. Values for model accuracy are according to [22, Table 1].

Table 2: GS Entropy of Structurally unambiguous non-stacking grammars under various parameter sets. Grammar description and parameters according to [22]. `Basic Grammar` denoted as BG. Extensions `wcx` and `stk` denote Watson-Crick base-pairing constraint and stacking versions, respectively. Column `Training Set` denotes the training set used to estimate model parameters. Logarithm was calculated in base 2.

Training Set	**BG**	**BGstk**	**BGwcx**	**g6**	**g6stk**	**g6wcx**
`TrA + 2 × TrB`	**37.160**	**37.079**	**35.214**	**378.740**	**377.086**	**357.630**
`TrA + TrB`	**47.056**	**46.978**	**44.680**	**412.892**	**411.477**	**390.547**
`TrA`	**73.114**	**73.026**	**69.660**	**469.959**	**468.933**	**446.016**
`TrB`	**14.889**	**14.904**	**13.991**	**243.074**	**240.575**	**228.354**

4. The Typical Set Criterion

As previously mentioned, current ML-based model training methods iteratively estimate rule probabilities from frequency of occurrence of corresponding rules in the predictions of RNA structures. In each step of the iteration, the predictions are in fact individual maximization of the probability of structure given the RNA sequence. In this

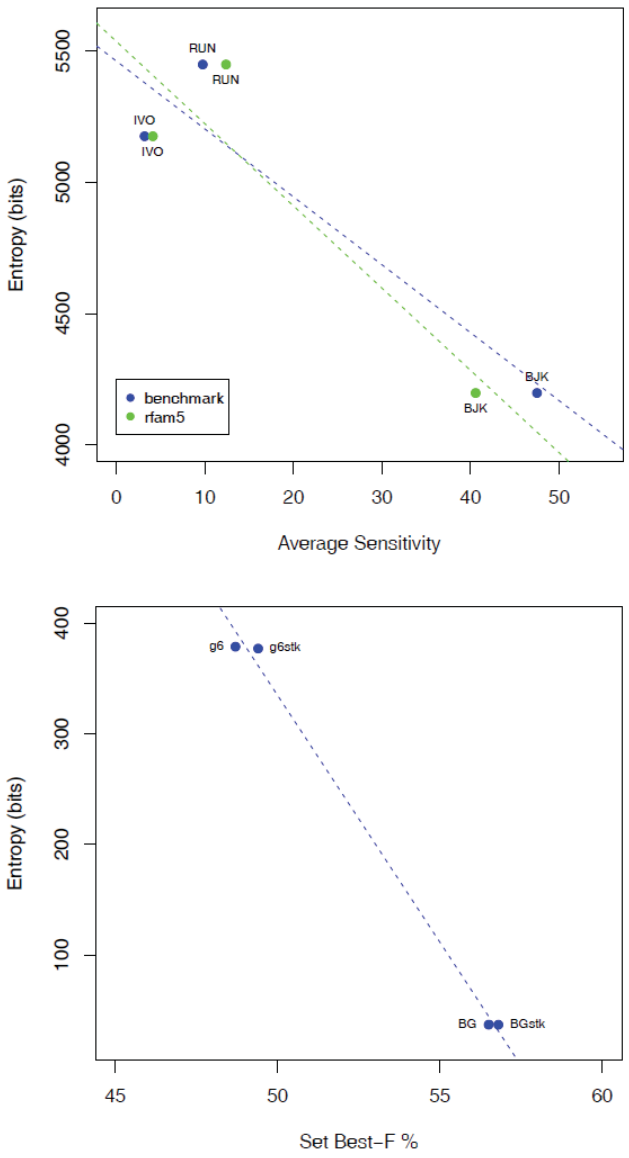


Fig. 1: Top view: The GS entropy of grammar models trained on the `mixed80` dataset against average sensitivity to structures in `Benchmark` and `Rfam5` test sets, separately. Sensitivity values according to conus software [21]. Grammar names and entropy values are according to Table 1. Trendlines for each test set are shown as dashed lines. Logarithm was calculated in base 2. Bottom view: The GS entropy of grammar models trained on the `TrA + 2 × TrB` dataset were plotted against best F measure values. Grammar names and entropy values are according to Table 2. The best F measure values were taken from [22, Table 1]. Trendline corresponding to entropy values is shown as a dashed line. Logarithm was calculated in base 2.

section, a criterion, denoted as the Typical Set Criterion (TSC) is presented to maximize the probabilities of the observed RNA secondary structures $\Pi Y_{obs.}$.

Satisfying the joint maximization of all observed RNA structures is done by tuning the grammar model such that the structures become a part of the typical set[2] of structures generated by the grammar: $\Pi Y_{obs.} \subset T_\epsilon(\Pi Y)$. The typical set of structures for grammar space (G, Θ) with parameter ϵ, can be defined as:

$$T_\epsilon(\Pi Y) = \{(\pi, y) : |\log p(\pi, y) - E[\log p(\Pi Y)]| \leq \epsilon E[\log p(\Pi Y)]\} \quad (5)$$

The following immediately implies for any structure in $T_\epsilon(\Pi Y)$:

$$(1-\epsilon)H(\Pi Y) \leq -\sum_i r_i(\pi, y)\log(p_i) \leq (1+\epsilon)H(\Pi Y), \quad \forall(\pi, y) \in T_\epsilon(\Pi Y) \quad (6)$$

where p_i is the probability of rule i in the model and $r_i(\pi, y)$ denotes the number of times rule i is deployed in structure (π, y). Note that the GS Entropy introduced in (4) is substituted for the expected log-likelihood of structures $H(\Pi Y)$.

All observed RNA secondary structures are constrained by (6), since they are to be a subset of the typical-set structures in the given grammar. Let's refer to this constraint as the Typical-Set Criterion (TSC) for RNA secondary structure model training. In order to satisfy (6) for all observed structures, the grammar rule probabilities $\hat{\Theta}$ is proposed, which yields from the following minimization:

$$\hat{\Theta} = \arg\min_{\Theta} \| - \mathbf{R}_G(\Pi Y_{obs.}) \times \log \Theta - H(\Pi Y|G, \Theta).\mathbf{1}\| \quad (7)$$

Where function $\mathbf{R}_G(\pi, y)$ maps the structure (π, y) to a $1 \times N$ vector whose elements are $r_i(\pi, y)$ with N being the total number of rules in the grammar G. $\mathbf{R}_G(\Pi Y_{obs.})$ is then an $M \times N$ matrix with M being the total number of observed RNA secondary structures. The $\log \Theta$ is a $N \times 1$ vector containing log-likelihoods of grammar rule probabilities which are subject to the proposed minimization.

5. Discussion and Conclusions

In this work, a procedure was presented for calculating the Shannon's entropy of SCFGs that model RNA folding. This measure of structural space was then used as a new constraint for RNA structure model training. The GS entropy values for certain SCFG-based RNA folding models were calculated. The GS entropy is more dependent on the training dataset than choice of grammar design (See Tables 1 and 2).

[2]The typical set is a set of sequences whose probability is close to one [26, pg. 62].

Lower GS entropy is generally associated with higher model accuracy. The GS entropy of the BJK model is lower than the other lightweight models while its accuracy to predict RNA secondary structure is higher (Figure 1 both Top view and Bottom view). Furthermore, it can be seen from Table 2 that the GS entropy of the `TrATrBTrB`-trained `Basic Grammar` is significantly lower than that of the `TrATrBTrB`-trained BJK model (g6). Accuracy of the `Basic Grammar` is approximately 10% higher than the BJK model (from 48.7 to 56.5 best F measure [22, Table 1]). The same argument holds for stacking versions of the above grammars. Although the association of higher model accuracy and lower uncertainty (here, Shannon's entropy) is somewhat intuitive, significantly low GS entropy may also indicate overfitting. The entropy of the mentioned BJK model is approximately ten times higher than `Basic Grammar` (378.7 to 37.2 in Table 2). This implies that the folding space of the BJK model with the `TrATrBTrB`-trained parameter set can be as significantly higher than the `Basic Grammar`. To get an idea of the significant difference between folding spaces, consider the following argument: An approximation or an upper bound for the largeness of the space of a probabilistic model S is $2^{H(S)}$. For the two BG and g6 folding models, consider GS entropy values corresponding to the `TrB` training set and Watson-Crick constraint. The ratio of folding space is in the order of $228.354 - 13.991 = 214.363$. This is actually the smallest difference between corresponding entropy values in Table 2. Such significant reduction in folding space ($\approx 2^{228}$) for approximately 10% increase in model accuracy, here best F measure, may not be desirable.

6. Acknowledgements

This work resulted from useful conversations in the RNA-informatics lab at the University of Georgia. Communication with Robin D. Dowell regarding the conus software was very helpful. Many thanks to Elena Rivas for helping me understand the TORNADO language. In addition, Dr. Rivas provided model parameters. This research was supported in part by the Intramural Research Program of the NIH, National Library of Medicine. I thank Dr. John L. Spouge for his support.

References

[1] N. Chomsky, "On certain formal properties of grammars," *Information and Control*, vol. 2, no. 2, pp. 137–167, 1959.

[2] T. L. Booth and R. A. Thompson, "Applying probability measures to abstract languages," *IEEE Transactions on Computers*, vol. C-22, pp. 442–50, May 1973.

[3] J. K. Baker, "Trainable grammars for speech recognition," *The Journal of the Acoustical Society of America*, vol. 65, pp. S132–S132, 1979.

[4] U. Grenander, *Syntax-controlled probabilities*, ser. Reports // BROWN UNIV, 1969.

[5] W. Kuich, "On the entropy of context-free languages," *Information and Control*, vol. 16, no. 2, pp. 173–200, 1970.

[6] S. E. Hutchins, "Moments of string and derivation lengths of stochastic context-free grammars," *Information Sciences*, vol. 4, no. 2, pp. 179–191, 1972.

[7] S. Soule, "Entropies of probabilistic grammars," *Information and Control*, vol. 25, no. 1, pp. 57–74, 1974.

[8] C. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, July-October 1948.

[9] T. Smith and M. Waterman, "RNA secondary structure: A complete mathematical analysis," *Math Biosci*, vol. 42, pp. 257–266, 1978.

[10] P. R. Stein and M. S. Waterman, "On some new sequences generalizing the catalan and motzkin numbers," *Discrete Mathematics*, vol. 26, no. 3, pp. 261–272, 1979.

[11] G. Viennot and M. Vauchaussade de Chaumont, *Enumeration of RNA Secondary Structures by Complexity*, ser. Lecture Notes in Biomathematics. Springer Berlin Heidelberg, 1985, vol. 57, ch. 50, pp. 360–365.

[12] I. Hofacker, P. Schuster, and P. Stadler, "Combinatorics of RNA secondary structures," *Discrete Appl. Math.*, vol. 88, no. 1-3, pp. 207–237, 1998.

[13] M. E. Nebel, "Combinatorial properties of RNA secondary structures," *J Comput Biol*, vol. 9, no. 3, pp. 541–73, 2002, using Smart Source Parsing.

[14] B. Liao and T. M. Wang, "General combinatorics of RNA secondary structure," *Math Biosci*, vol. 191, no. 1, pp. 69–81, 2004, liao, Bo Wang, Tian-ming Math Biosci. 2004 Sep;191(1):69-81.

[15] T. DoŽlic, D. Svrtan, and D. Veljan, "Enumerative aspects of secondary structures," *Discrete Mathematics*, vol. 285, no. 1-3, pp. 67–82, 2004.

[16] S. R. Eddy and R. Durbin, "RNA sequence analysis using covariance models," *Nucleic Acids Res*, vol. 22, pp. 2079–88, 1994.

[17] Y. Sakakibara, M. Brown, R. Hughey, I. S. Mian, K. Sjolander, R. C. Underwood, and D. Haussler, "Stochastic context-free grammars for tRNA modeling," *Nucleic Acids Res*, vol. 22, pp. 5112–20, 1994.

[18] Z. Yao, Z. Weinberg, and W. L. Ruzzo, "CMfinder–a covariance model based RNA motif finding algorithm," *Bioinformatics*, vol. 22, no. 4, pp. 445–52, 2006, using Smart Source Parsing Feb 15; Epub 2005 Dec 15.

[19] E. P. Nawrocki and S. R. Eddy, "Infernal 1.1: 100-fold faster RNA homology searches," *Bioinformatics*, vol. 29, no. 22, pp. 2933–5, 2013.

[20] M. Zuker and P. Stiegler, "Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information," *Nucleic Acids Res*, vol. 9, pp. 133–48, 1981.

[21] R. D. Dowell and S. R. Eddy, "Evaluation of several lightweight stochastic context-free grammars for RNA secondary structure prediction," *BMC Bioinformatics*, vol. 5, p. 71, 2004.

[22] E. Rivas, R. Lang, and S. R. Eddy, "A range of complex probabilistic models for RNA secondary structure prediction that includes the nearest-neighbor model and more," *RNA*, vol. 18, no. 2, pp. 193–212, 2012.

[23] I. L. Hofacker, "Vienna RNA secondary structure server," *Nucleic Acids Res*, vol. 31, no. 13, pp. 3429–31, 2003.

[24] C. B. Do, D. A. Woods, and S. Batzoglou, "CONTRAfold: RNA secondary structure prediction without physics-based models," *Bioinformatics*, vol. 22, no. 14, pp. e90–e98, 2006.

[25] C. van Rijsbergen, *Information Retrieval*, 2nd ed. London Butterworths, 1979.

[26] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. Wiley-Interscience, 2006.

Structure of the Intracellular Loop Domain in the $GABA_A$ Receptor α-Subunit

J.L. Mustard, J.B. Worley, and N.W. Seidler
Department of Biochemistry, Kansas City University of Medicine and Biosciences,
Kansas City, Missouri, USA

Abstract - *This crucial target for pharmacological intervention, $GABA_AR$, is a chloride channel that controls neuronal inhibition. The ligand-binding and transmembrane domains are highly-conserved, well-characterized and comprehensively studied. However, the intracellular loop domain (ILD) is strikingly variable, and its structure is still a mystery. Using computational approaches and bioinformatics, we garnered new insight into the structural orientation of this region. GAPDH, the "housekeeping" glycolytic enzyme, has binding sites in the ILD that may offer clues to the role of this protein in clustering the receptors at synaptic regions. We propose that the pattern of lysine residues attracts acidic phospholipids, recruiting GAPDH to this site.*

Keywords: $GABA_A$ receptor, intracellular loop domain, neuronal inhibition, GAPDH, chloride channel

1 Introduction

The highly ubiquitous and multi-functional protein, GAPDH (for, GlycerAldehyde-3-Phosphate DeHydrogenase) is known to interact with diverse cellular proteins [1], including the GABA (for, Gamma-AminoButyric Acid) receptor of the type A class (abbreviated as $GABA_AR$), which is a neurotransmitter-gated chloride ion channel that mediates inhibitory neurotransmission. Laschet and colleagues [2] observed that GAPDH phosphorylates $GABA_AR$, and thereby modulates neuronal inhibition. $GABA_AR$ is a member of the Cys-loop family of pentameric ligand-gated ion channels (often abbreviated pLGICs). Cys-loop receptors contain a characteristic disulfide bond between two highly conserved cysteine residues that are separated by thirteen residues creating a well-defined loop that is projected extracellularly. The pentameric nature of the channel refers to the annular arrangement of five distinct protein chains, or subunits, that are embedded in the plasma, or cell surface, membrane. $GABA_AR$ is heteromeric, meaning that it is made up of different subunits (Figure 1), and, the native ligand is the neurotransmitter GABA. The $GABA_AR$ is found throughout the nervous system and is essential for neural signaling. The modular arrangement of each of the five subunits consists of an extracellular ligand binding domain (LBD), four α-helical transmembrane domains (TMDs), designated TM1 to TM4, and a large intracellular loop domain (ILD).

As indicated in our study, there is a substantial heterogeneity among subunit subtypes in the region of the TM3-TM4 ILD. In a 1999 review article [3], the authors indicate that channel function is clearly affected by changes within this domain. Curiously, studies point to a linkage between a naturally-occurring substitution in the α6 ILD (i.e. P385S) and alcoholism [4] as well as reduced response to benzodiazepine agonists [5, 6], suggesting that ILD plays a key role in modulating receptor behavior. While Kash and colleagues [7] demonstrated the importance of the extracellular N-terminal domain and the TM2-TM3 extracellular domain of the α-subunits in the transduction of agonist binding, less is known about the intracellular region.

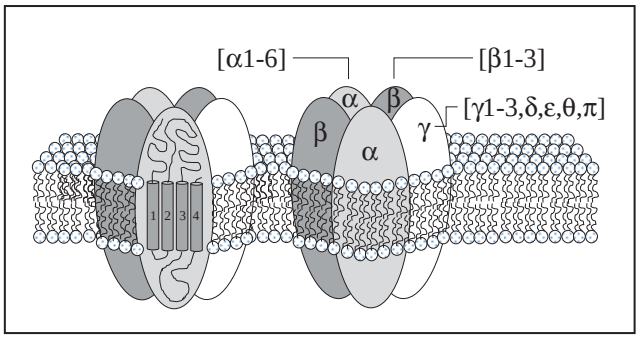


Figure 1: Illustration of the $GABA_AR$
The subunits of pLGICs share common structural features including an extracellular N-terminal domain, four TMDs and a large cytoplasmic loop between TM3 and TM4 that is the focus of the present study. The pentameric arrangement of the subunits is illustrated showing extra- and intracellular domains (*shown above and below the lipid bilayer, respectively*) as well as the conserved TMDs. There are 19 different genes (including the three rho, ρ, subunits that are omitted from this diagram). Combinations of different subunits result in a diverse group of $GABA_AR$ subtypes. Generally, there are two α and two β subunits together with one γ or δ subunit.

Fisher [8] reported that the TM3 region and/or the TM3-TM4 ILD influence the gating properties of this channel. In fact, the author compared α1 against α6, which contain differences in sequences in the area of GAPDH binding.

The TM3 domain appears crucial in the mechanism of anesthesia. The water-filled cavity that is formed by TM2 and TM3 acts as a binding pocket for ethanol and volatile anesthetics [9]. Interestingly, the regions directly adjacent to the TM3 and TM4 domains appear to be crucial for receptor function [10, 11]. Mutation of a conserved arginine (R427) at the junction between the ILD and TM4 domains affects the desensitization mechanism in pLGICs. This R427 aligns at the position of the asparagine (N414) at the α1 GAPDH consensus site (abbreviated GCS) that is identified as a sequence of five residues, -NXXS/TK-, where X is any residue followed by either an S (i.e. serine) or T (i.e. threonine), which are the targets of phosphorylation. O'Toole and Jenkins [10] identified regions adjacent to the TM3 and TM4 domains that are important in the regulation of channel function. The authors designated these regions as M3A and M4A with stretches of approximately 20 and 40 residues, respectively, noting the importance of charged residues particularly lysine residues. Of interest to our lab group, these regions contain GCSs, but only in some of the α subunits. In the absence of x-ray crystal structure of the ILD, we used computational/modeling strategies to propose a structure of this intracellular domain, including a functional role.

1.1 Methodological Approach

We used various databases to examine the features of the α subunits. Sequence information was obtained from www.uniprot.org, using the following accession numbers for human α1 to α6: P14867, P47869, P34903, P48169, P31644, and Q16445, respectively. Expression data was collected from www.biogps.org and the specific probeset number was noted. The protein BLAST program for comparing sequences that is available at www.ncbi.nlm.nih.gov was used to compare the six α subunits, namely the TM3 to TM4 region. The program for predicting protein regions of disorder was also employed; this was accessed at www.pondr.com. We additionally used secondary structure prediction programs (jpred and predictprotein). We then manually applied the Chou-Fasman predictive algorithm [12]. We tested a speculative structural model using the helical wheel in examining the possibility of surface attachment of ILD helices.

2 Results and Discussion

By conservative estimate, there are eleven distinct receptor subtypes that are abundant in brain [13]. The distribution of the various receptor subtypes is functionally divided into synaptic and non-synaptic regions, which regulate phasic and tonic neuronal inhibition, respectively. While not absolute, the α1-3 distribute to synapses and the α4-6 to non-synaptic regions on neurons. Regarding the expression of the α subunits in brain, the α1, α2, and α5 were highly expressed in all regions of the brain (Table 1). There was one exception and that was the preferential expression of α6 in the cerebellum. Curiously, the α1, α2, α3 and α5, but not the α6, contain GCSs that would be modulatable by GAPDH.

Table 1: Expression levels of the α subunit subtypes

TISSUE	1	2	3	4	5	6
Whole Brain	132.60	98.30	3.85	3.00	203.85	4.95
Amygdala	82.75	321.45	4.75	3.65	691.20	6.15
Prefrontal Cortex	113.70	163.65	5.65	4.35	80.40	7.50
Spinal Cord	5.40	7.05	4.85	3.65	6.60	6.45
Hypothalamus	42.10	10.95	5.05	3.80	12.25	6.40
Thalamus	5.50	6.85	4.65	3.45	76.85	6.20
Caudate Nucleus	4.55	33.75	4.10	4.60	106.85	5.45
Parietal Lobe	137.20	94.60	4.95	3.70	10.15	7.05
Medulla Oblongata	81.90	65.80	4.20	3.15	10.65	5.50
Cingulate Cortex	77.20	43.80	4.70	3.50	22.95	6.25
Occipital Lobe	161.40	203.55	4.10	3.50	169.15	5.40
Temporal Lobe	29.95	49.85	4.20	3.15	24.10	5.70
Subthalamic Nucleus	44.25	37.65	4.20	3.05	43.45	5.70
Pons	25.40	13.65	4.35	3.30	26.10	5.95
Globus Pallidus	61.40	72.20	3.50	2.50	24.45	4.60
Cerebellum	13.45	7.10	3.70	2.80	5.15	49.00
Adipocyte (Control)	5.05	6.35	4.50	3.40	6.10	5.85

2.1 Sequence comparison of α subunits

In doing sequence comparison using BLAST, we met an obstacle in comparing sequences of α1 with those of α4. While the sequences exhibited conserved residues at the TM3 and TM4 regions providing us with BLAST-derived numbers, the sequences associated with the ILD initially came up with no homologous regions. We then pared the sequences down to only those stretches of the corresponding ILDs for α1 and α4, and BLAST comparison performed. The regions were further pared down, omitting the just-identified similar stretches and then the remaining sequences were re-tested. This process was necessary due to the large disparity in ILD size between α1 and α4. The homologous regions were aligned and a pairwise alignment score was determined (Table 2).

Table 2: Pairwise alignment scoring

α1: *versus*	**Score**	**Identities**	**Similarities**
α2	998	94	12
α3	848	79	15
α5	833	78	12
α4	470	51	9
α6	621	58	12

Pairwise alignment scores were computed by the following rules. For each identity and similarity, a value of 10 and 5 were given; the gap penalty was -1 for each gap. Using this scoring system allowed us to evaluate all sequence pairs.

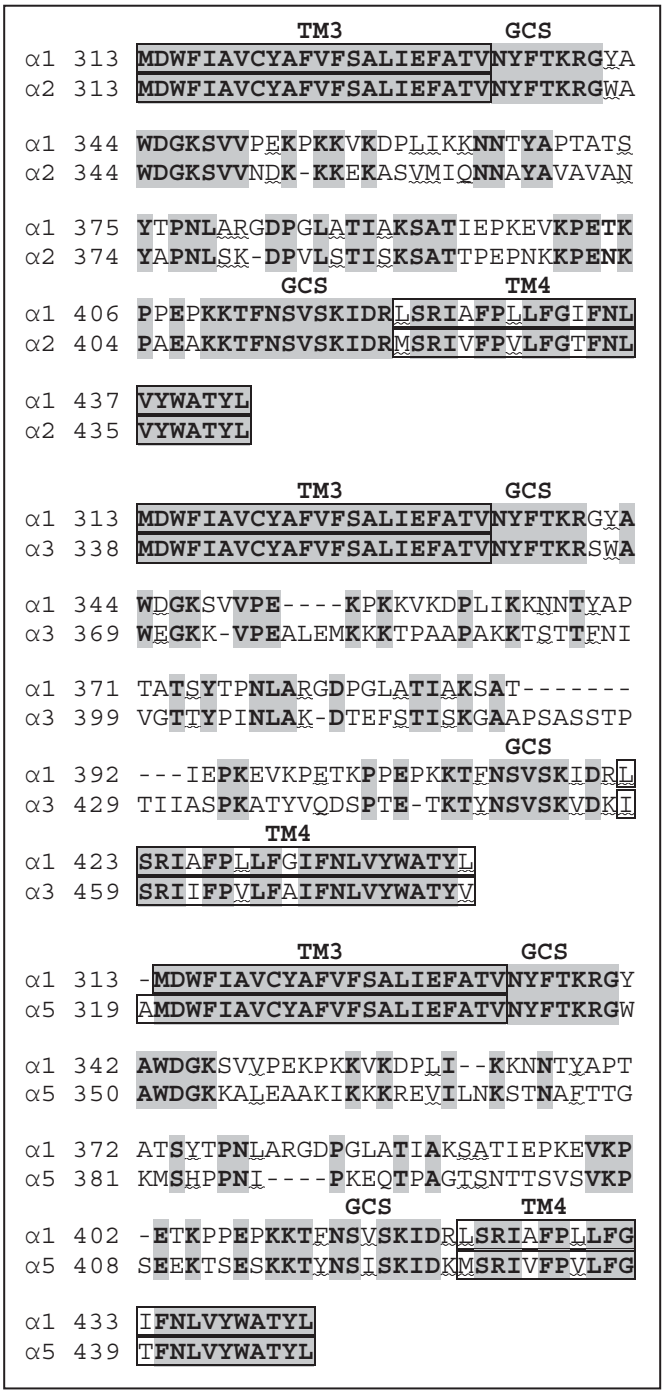


Figure 2: Sequence comparisons of conserved ILDs
The ILD sequence of the α1 subunit was compared with that of α2, α3 and α5. These comparisons gave the highest alignment scores. All of these regions contain a conserved GCS adjacent to the transmembrane domains (*shown boxed*). Identities (*grey highlight and bolded letters*) and similarities (*squiggled underline*).

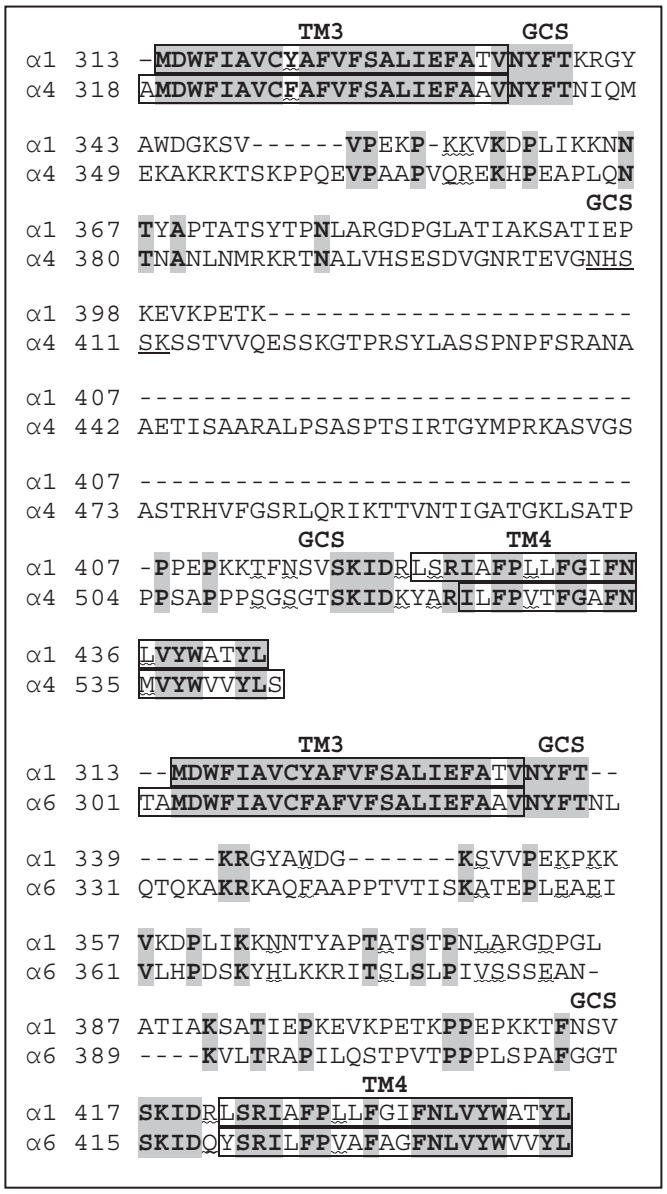


Figure 3: Comparison of less-conserved ILDs
The ILD sequence of the α1 subunit was pairwise compared with that of α4 and α6. These comparisons gave the lowest alignment scores (Table 2). The α4 and α6 subunits do not contain intact TM3-adjacent or TM4-adjacent GCSs. The α4 subunit interestingly exhibits an internal GCS (*underlined*).

2.2 Subdomains of the ILD

In a study on the role of the ILD [10], it was proposed that the sequences closest to the transmembrane regions are crucial to receptor function. In fact, the authors gave these regions an initial designation of M3A and M4A, referring to the adjacent residues relative to the transmembrane domains. For the α1 subunit, the authors suggested that the 20 and 40 residues from the TM3 and TM4 domains contain residues

important for channel function. The α1 mutations K383E and K384E resulted in enhanced desensitization of the receptor [10].

(See accession number NP_000797; it appears that the authors [10] erred on the numbering of these residues, which should be listed as K410 and K411. Consult our Figures 2 and 4 in this paper for the correct numbering of these lysine residues. We assume the accuracy of the study [10] regardless of their numbering typo.)

These authors also report that mutation of the K378 (*note:* the actual location is K405) decreases chloride conductance, but that the introduction of a lysine residue at V373 (*note:* V400) did not change chloride conductance. This observation suggests that the lysines and their location in the ILD are vital to channel function.

2.3 Sequence patterns in the ILD

We examined the subdomains in more detail and noticed several interesting features (Figure 4). The pattern of lysine residues exhibits rhythmicity. In the TM3A subdomain, we find a pattern mirrored in the TM4A subdomain (Figure 5, α1). This pattern exhibits a lysine singlet/doublet every 5 to 7 residues interspersed with larger lysine groupings. TM3A and TM4A are separated by a sequence devoid of lysines in a central region of the loop that shows the highest degree of disorder.

```
<-------------TM3A------------>
NYFTKRGYAWDGKSVVPEKPKKVKDPLIKK

         ****************
NNTYAPTATSYTPNLARGDPGLATIA

<-------------TM4A------------>
KSATIEPKEVKPETKPPEPKKTFNSVSKIDR
```

Figure 4: The α1 ILD sequence

The ILD sequence (335-421) from the α1 subunit is shown with the basic residue lysine (K) *highlighted in grey*. The centrally located disordered region is indicated by *asterisks* above those residues determined to have PONDR scores greater than 0.70. The sequence is presented showing two important subdomains, TM3A and TM4A, representing the residues adjacent to the TM3 and TM4, respectively.

The data in Figure 5 include the transmembrane domains (TM3 and TM4) flanking the ILD; the low PONDR scores indicate their minimal disorder. The graph showing the disorder profile for the α1 subunit indicates a palindromic pattern of lysine residues with greatest amount of disorder at the region without any lysines. Interestingly, this elegant palindromic feature appears partially disrupted in α2, α3, and α5 and more dramatically disrupted in α4 and α6.

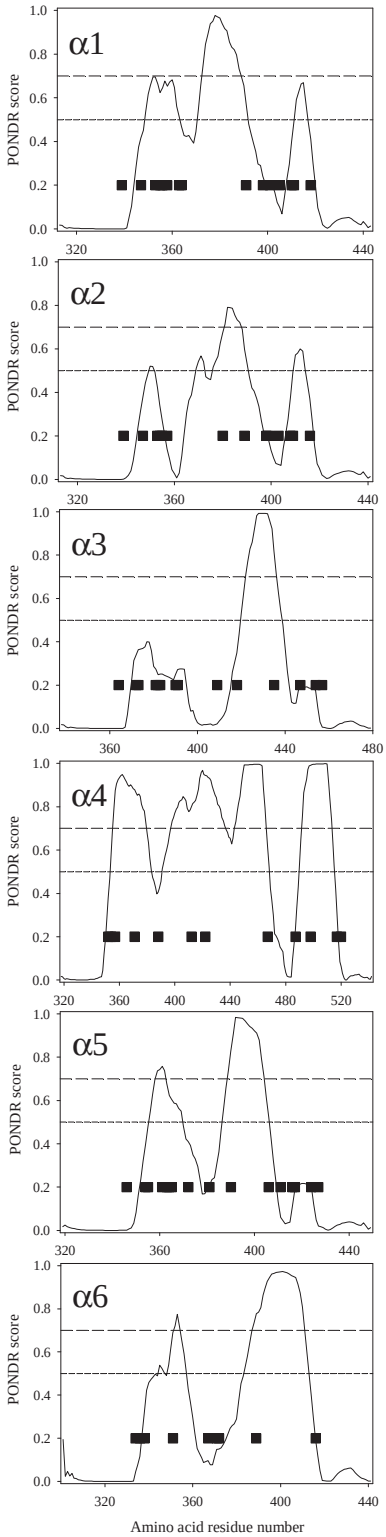


Figure 5: Comparison of order/disorder profiles

The graphs that show the predicted regions of disorder for each subunit, including TM3, TM4 and ILD sequences, were made using output data obtained from PONDR (VL-XT). The *short-* and *long-dashed* lines indicate 0.5 and 0.7 threshold levels for disorder. The *black boxes* represent the location of lysines.

2.4 Protein architecture of the ILD

When we applied the jpred predictor for secondary structure of α1 ILD, the output indicated that only residues directly adjacent to the transmembrane domains exhibited secondary structure (Figure 6).

```
335 NYFTK---AWD 345


417 SKIDR 421
```

Figure 6: Secondary structure prediction
The residues shown are predicted to have helical architecture. The *grey highlighted* residues represent the complete and partial GCS regions.

There are several lines of evidence that suggest that the peptide chains emanating from the membrane at the TM3 and TM4 domains are structurally organized in part by the physical forces provided by the phospholipid headgroups. We propose that the TM3A and TM4A chains are helical and that their periodicity is approximately six to seven residues per turn which is reflected in the regular positioning of the lysine residues. Using the Chou-Fasman predictive algorithm [12], we computed the propensities of the TM3A and TM4A regions for forming secondary structures (Figure 7).

```
   Helical Propensity
TM3A
352 EKPKKVKDPLIKK 364
TM4A
398 KEVKPETK 405
408 EPKKTF 413
415 SVSKID 420

   Strand Propensity
TM4A
412 TFNSVS 417
```

Figure 7: Chou-Fasman results
The residues shown are predicted to have helical or strand architecture. The *grey highlighted* residues represent the partial GCS regions.

The three-dimensional structure of the ILD is, as of now, unmapped [14]. The recent mapping of the homopentameric (all β3 subunits) $GABA_AR$ was accomplished only by removal of the ILD portion. This region appears to exhibit the capability of forming stretches of helical structure that may be stabilized by interaction with membrane components and/or other proteins, such as GAPDH. Additionally, one may speculate that the TM3A and TM4A show features that are somewhat reminiscent of the collagen helix. TM3A and TM4A contain high amounts of proline and lysine residues. The α1 ILD contains eleven proline and fifteen lysine residues out of a total of 87. The periodicity of the helix would allow for the regular positioning of the lysine residue towards the lipid bilayer permitting penetration of the positively charged side chains into the layer containing the phosphate moieties. Others have found the lysine-containing proteins bind to membrane lipid phosphates [15]. Interestingly, anesthetics disrupt this interaction and may contribute to the molecular mechanism of anesthesia [16], which is currently thought to involve the $GABA_AR$. The central region of this loop (i.e. α1) appears very disordered and completely devoid of lysine residues, and as such, may be unencumbered enough to occupy multiple configurations that would allow this region to disengage from the membrane leaflet.

Based on our computational evidence, we surmise that the lysine residues form a pattern that is involved in phosphatidylserine (PS) sequestration. PS is a major annular phospholipid at the inner leaflet, surrounding the pentameric channel. There is a strong probability of an interaction between the ILD lysine residues with the phosphate boundary layer of the inner leaflet of the lipid bilayer, particularly considering that researchers were unable to crystalize this protein with the ILD. We also propose that the ILDs from neighboring channels interact and that these cooperative interactions may be facilitated by GAPDH, which contains a PS-binding site and an ILD-binding site.

2.5 Role of the α1 ILD in $GABA_AR$

Approximately 50% of all $GABA_ARs$ in the adult brain contain the α1 subunit [17]. In cerebellar granule cells, Peran and colleagues [18] showed that the α1 ILD controls lateral mobility, contributing to cell surface immobilization. The authors note that immunocytochemical studies have ruled out the involvement of this region in trafficking the receptor to the cell surface. Interestingly, the α2 subunit has a de-clustering effect in engineered synapses in HEK293 cells, compared to that of the α1 subunit [19], noting homology differences in the ILD. It was previously determined that post-synaptic localization is directed in large part by the γ subunit ILDs [20]. Clustering mechanisms that involve other binding proteins include: (1) dystrophin [21], which is thought to anchor α1 (and neuroligin-2) to cortical dendrites, (2) radixin [22], which selectively anchors the α5-containing receptors to the actin cytoskeleton, and gephyrin [23], which appears to bind to α1, α2, and α3 with a lowered affinity for the α2 subunit. The gephyrin binding site was shown to be in the α3 ILD (specifically, 396 FNIVGTTYPI 405) [24] and in the α1 ILD (specifically, 361 LIKKNNTYAPTATSYT 376) with an emphasis on the phosphorylation target Thr376 [numbered 360-375 in ref. 25], each at a region approaching a disorder minima and a threonine-rich stretch.

Since the GAPDH enzyme binds to and phosphorylates the α1 ILD at the GCS (i.e. 335 NYF<u>T</u>K 339 and 414 NSV<u>S</u>K 418, at the *underlined* residues) [2], GAPDH must form cooperative interactions with the surface membrane. Curiously, GAPDH contains a phosphatidylserine binding site [26]. The residues on GAPDH that bind phosphatidylserine

are 70-94 and the near-neighbor residues are positioned at a distance of approximately 20 Å from one another in the tetrameric form of GAPDH.

The TM3A and the TM4A regions, according to the results from predictprotein.org, show two buried residues each, and they are located at or in close proximity to the GCSs on both ends of the ILD. This suggests that the GCS may exhibit some properties of tertiary structure, enabling it to bind GAPDH. The predictive features accessible through proteinpredict.org allowed us to propose that the central region (i.e. residues 365 to 390) of the α1 ILD, which is devoid of lysine residues, likely exhibits tertiary structure. The results indicate that residues T367, S374, and N378 are buried while the rest of the residues in this stretch are exposed (Figure 8).

Figure 8: Structure of the central ILD region
The predicted structure of the central region of the α1 ILD derived from information obtained from predictprotein.org. This stretch of amino acids lies between the TM3A and TM4A subdomains. The *grey highlighted* residues are predicted to be buried. The stretch of residues that is *boxed* is predicted to show strand-like secondary structure. The residues *bolded* are predicted to be protein bound. The *asterisks above* the sequence represent the binding site for gephyrin [25]. The *asterisks below* the sequence are predicted to be disordered (see Figure 4 and 5).

3 Conclusions

While the structural diversity of the $GABA_A$Rs in terms of subunit composition remains to be fully elucidated, it is general thought that the fast synaptic inhibition is due to receptor subtypes that predominately include α1, β2, γ2 subunits [17, 27]. The $GABA_A$R that mediate tonic inhibition are thought to be composed primarily of α4, α5 and α6, with β1-3, that may contain a δ subunit [27, 28]. Our study focused on the α1 subunit, particularly the ILD sequence. The six α subunit ILDs exhibit significant heterogeneity of size and sequence. We propose that the pattern of lysine residues is involved in phosphatidylserine accumulation as the major annular phospholipid at the inner leaflet, surrounding the pentameric channel. These lysine residues likely interact with the phosphate boundary layer of the inner leaflet of the lipid bilayer. We propose that ILDs interact with neighboring ILDs and that these cooperative interactions may be facilitated by GAPDH. Furthermore, GAPDH and gephyrin have a similar binding partner, dynein [29, 30], which may participate in complex formation, as described previously [31]. Our working model of the structure of the ILD suggests the primary organizing feature is the pattern of lysine residues that interact with the phosphate layer of mainly phosphatidylserine phospholipids, which are further ordered by the interaction with GAPDH.

The role of GAPDH in moderating the physical orientation of $GABA_A$R clusters suggests that modification of this protein would play a significant role in anesthesia, epilepsy, and aging-related brain disorders, such as Alzheimer's, offering a novel point of medical intervention.

4 References

[1] Seidler NW. "Multiple binding partners"; *Advances in Experimental Medicine and Biology*, 2013; 985:249-267.

[2] Laschet JJ, Minier F, Kurcewicz I *et al*. "Glyceraldehyde-3-phosphate dehydrogenase is a $GABA_A$ receptor kinase linking glycolysis to neuronal inhibition"; *Journal of Neuroscience*, 2004; 24(35):7614-7622.

[3] Mehta AK, Ticku MK. "An update on GABAA receptors"; *Brain Res Brain Res Rev*, 1999; 29(2-3):196-217.

[4] Iwata N, Virkkunen M, Goldman D. "Identification of a naturally occurring Pro385-Ser385 substitution in the GABA(A) receptor alpha6 subunit gene in alcoholics and healthy volunteers." *Mol Psychiatry*. May 2000; 5(3):316-9.

[5] Iwata N, Cowley DS, Radel M, Roy-Byrne PP, Goldman "Relationship between a GABAA alpha 6 Pro385Ser substitution and benzodiazepine sensitivity"; *Am J Psychiatry*. Sep 1999; 156(9):1447-9.

[6] Hoffman WE, Balyasnikova IV, Mahay H, Danilov SM, Baughman VL. "GABA alpha6 receptors mediate midazolam-induced anxiolysis." *J Clin Anesth*. May 2002; 14 (3):206-9.

[7] Kash TL, Jenkins A, Kelley JC, Trudell JR, Harrison NL. "Coupling of agonist binding to channel gating in the GABA(A) receptor." *Nature*. Jan 2003; 421(6920):272-5.

[8] Fisher JL. "The alpha 1 and alpha 6 subunit subtypes of the mammalian GABA(A) receptor confer distinct channel gating kinetics." *J Physiol*. Dec 2004; (2):433-48.

[9] Jenkins A1, Greenblatt EP, Faulkner HJ, Bertaccini E, Light A, Lin A, Andreasen A, Viner A, Trudell JR, Harrison NL. "Evidence for a common binding cavity for three general anesthetics within the GABAA receptor." *J Neurosci*. Mar 2001; 21(6):RC136.

[10] O'Toole KK, Jenkins A. "Discrete M3-M4 Intracellular Loop Subdomains Control Specific Aspects of γ-Aminobutryic Acid Type A Receptor Function"; *J Biol Chem*, Nov 2011; 286(44):37990-37999.

[11] Hu XQ, Sun H, Peoples RW, Hong R, Zhang L. "An interaction involving an arginine residue in the cytoplasmic domain of the 5-HT3A receptor contributes to receptor desensitization mechanism." *J Biol Chem*. Aug 2006; 281(31):21781-8

[12] Chou PY, Fasman GD. "Empirical predictions of protein conformation." *Annu Rev Biochem*. 1978; 47:251-76

[13] Luscher B, Fuchs T, Kilpatrick CL. "GABAA receptor trafficking-mediated plasticity of inhibitory synapses." *Neuron*. May 2011; 70(3):385-409.

[14] Miller PS, Aricescu AR. "Crystal structure of a human GABAA receptor." *Nature*. Aug 2014; 512(7514):270-5.

[15] Kim J, Mosior M, Chung LA, Wu H, McLaughlin S. "Binding of peptides with basic residues to membranes containing acidic phospholipids." *Biophys J*. Jul 1991; 60(1):135-48.

[16] Bangham AD, Mason W. "The effect of some general anaesthetics on the surface potential of lipid monolayers." *Br J Pharmacol*. Jun 1979; 66(2):259-65.

[17] McKernan RM, Whiting PJ. "Which GABAA-receptor subtypes really occur in the brain?" *Trends Neurosci*. Apr 1996; 19(4):139-43

[18] Perán M., Hooper H., Boulaiz H., Marchal J.A., Aránega A. and Salas R.. "The M3/M4 cytoplasmic loop of the alpha1 subunit restricts GABAARs lateral mobility: a study using fluorescence recovery after photobleaching." *Cell Moti. Cytoskeleton*, Dec 2006 63(12), 747-757.

[19] Dixon C, Sah P, Lynch JW, Keramidas A. "GABAA receptor α and γ subunits shape synaptic currents via different mechanisms." *J Biol Chem*. Feb 2014; 289(9):5399-411.

[20] Alldred MJ, Mulder-Rosi J, Lingenfelter SE, Chen G, Lüscher B. "Distinct gamma2 subunit domains mediate clustering and synaptic function of postsynaptic GABAA receptors and gephyrin." *J Neurosci*. Jan 2005; 25(3):594-603

[21] Panzanelli P, Gunn BG, Schlatter MC, Benke D, Tyagarajan SK, Scheiffele P, Belelli D, Lambert JJ, Rudolph U, Fritschy JM. "Distinct mechanisms regulate GABAA receptor and gephyrin clustering at perisomatic and axo-axonic synapses on CA1 pyramidal cells." *J Physiol*. Oct 2011; 589(20):4959-80.

[22] Loebrich S, Bähring R, Katsuno T, Tsukita S, Kneussel M. "Activated radixin is essential for GABAA receptor alpha5 subunit anchoring at the actin cytoskeleton." *EMBO J*. Mar 2006; 25(5):987-99

[23] Maric HM, Mukherjee J, Tretter V, Moss SJ, Schindelin H. "Gephyrin-mediated γ-aminobutyric acid type A and glycine receptor clustering relies on a common binding site." *J Biol Chem*. Dec 2011; 286(49):42105-14.

[24] Tretter V1, Kerschner B, Milenkovic I, Ramsden SL, Ramerstorfer J, Saiepour L, Maric HM, Moss SJ, Schindelin H, Harvey RJ, Sieghart W, Harvey K. "Molecular basis of the γ-aminobutyric acid A receptor α3 subunit interaction with the clustering protein gephyrin." *J Biol Chem*. Oct 2011; 286(43):37702-11.

[25] Mukherjee J1, Kretschmannova K, Gouzer G, Maric HM, Ramsden S, Tretter V, Harvey K, Davies PA, Triller A, Schindelin H, Moss SJ. "The residence time of GABA(A)Rs at inhibitory synapses is determined by direct binding of the receptor α1 subunit to gephyrin." *J Neurosci*. Oct 2011; 31(41):14677-87.

[26] Kaneda M, Takeuchi K, Inoue K, Umeda M. "Localization of the phosphatidylserine-binding site of glyceraldehyde-3-phosphate dehydrogenase responsible for membrane fusion." *J Biochem*. Dec 1997; 122(6):1233-40.

[27] Jacob TC, Moss SJ, Jurd R. "GABA(A) receptor trafficking and its role in the dynamic modulation of neuronal inhibition." *Nat Rev Neurosci*. May 2008; 9(5):331-43

[28] Farrant M, Nusser Z. "Variations on an inhibitory theme: phasic and tonic activation of GABA(A) receptors." *Nat Rev Neurosci*. Mar 2005; 6(3):215-29.

[29] Tisdale E. J., Azizi F., and Artalejo C. R., "Rab2 utilizes glyceraldehyde-3-phosphate dehydrogenase and protein kinase Cι to associate with microtubules and to recruit dynein." *Journal of Biological Chemistry*, Feb 2009; 284(9):5876–5884.

[30] Garcıa-Mayoral M. F., Rodrıguez-Crespo I., and Bruix M., "Structural models of DYNLL1 with interacting partners: African swine fever virus protein p54 and postsynaptic scaffolding protein gephyrin," *FEBS Letters*, Jan 2011; 585(1):53–57.

[31] Montalbano A.J., Theisen C.S., Fibuch E.E. and Seidler N.W. , "Isoflurane Enhances the Moonlighting Activity of GAPDH: Implications for $GABA_A$ Receptor Trafficking"; *International Scholarly Research Anesthesiology*, 2012, 970795.

SESSION

PROTEIN CLASSIFICATION and STRUCTURE PREDICTION, FOLDING, and COMPUTATIONAL STRUCTURAL BIOLOGY + DRUG DESIGN

Chair(s)

TBA

An Investigation of Minimum Data Requirement for Successful Structure Determination of Pf2048.1 with REDCRAFT

Casey A. Cole[1], Daniela Ishimaru[2], Mirko Hennig[3], and Homayoun Valafar[1*]

[1]Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208, USA

[2]Department of Biochemistry and Molecular Biology, Medical University of South Carolina, Charleston, SC 29425 USA

[3]Nutrition Research Institute, University of North Carolina at Chapel Hill, Kannapolis, NC 27514, USA

[*] Corresponding Author Email: homayoun@cec.sc.edu Phone: 1 803 777 2404 Fax: 1 803 777 3767

Mailing Address: Swearingen Engineering Center, Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208, USA

Abstract – Traditional approaches to elucidation of protein structures by NMR spectroscopy rely on distance restraints also know as nuclear Overhauser effects (NOEs). The use of NOEs as the primary source of structure determination by NMR spectroscopy is time consuming and expensive. Residual Dipolar Couplings (RDCs) have become an alternate approach for structure calculation by NMR spectroscopy. In this work we report our results for structure calculation of the novel protein PF2048.1 from RDC data and establish the minimum data requirement for successful structure calculation using the software package REDCRAFT. Our investigations start with utilizing four sets of synthetic RDC data in two alignment media and proceed by reducing the RDC data to the final limit of {CN, NH} and {NH} from two alignment media respectively. Our results indicate that structure elucidation of this protein is possible with as little as {CN, NH} and {NH} to within 0.533Å of the target structure.

Keywords: Protein Folding, Residual Dipolar Coupling (RDC), Residual Dipolar Coupling based Residue Assembly and Filter Tool (REDCRAFT), Secondary Structure.

1 Introduction

Proteins are a class of organic macromolecules that perform many important biochemical functions in biological cells. Protein functions run the entire gamut from structural support and transport of biomaterial, to performing important enzymatic activities within a living organism. Unlike the genetic material (DNA/RNA) within Eukaryotic cells, cytosolic proteins are not protected with an additional bilayer membrane of the nucleus. Therefore, design and delivery of protein-based intervention of diseases is more pragmatic in the near future than genetic treatment of diseases. Furthermore, principles of modern biology stress the importance of protein structure and its function. Therefore, knowledge of protein structures becomes paramount in understanding the mechanism of their function (or dysfunction) and subsequently, intelligent and appropriate drug design.

An understanding of protein structure at atomic resolution serves as the first and critical step in understanding the molecular basis of nearly all diseases. While structure determination of proteins is becoming more routine, the cost of structure determination remains the prohibitive factor. Thanks to improvements by the Structural Genomics Initiative[1], [2] and Protein Structure Initiative[3], the cost of experimental structure determination of proteins has been reduced from approximately $1,000,000 per protein to $100,000. Although this is a significant reduction in cost, it is still an impediment in achieving personalized medicine where nearly 100,000 protein structures will need to be determined for each person. This approximate cost of 10^{10} per person clearly represents a significant economical barrier.

In recent years, the use of Residual Dipolar Coupling (RDC) data acquired from Nuclear Magnetic Resonance (NMR) spectroscopy has become a potential avenue for a significant reduction in the cost of structure determination of proteins. Recent work[4]–[7] has demonstrated the challenges in structure calculation of proteins from RDC data alone, and some potential solutions have been introduced[5], [6], [8]. One such approach named REDCRAFT[4], [9], [10] has been demonstrated to be successful in structure calculation of proteins from a reduced set of RDC data. The main objective in this research is to perform a feasibility study for structure calculation of a novel protein from RDC data. Our feasibility study will establish the minimum required data for unambiguous structure calculation that is optimized for a given protein. A better understanding of minimum data requirement will help to alleviate the cost of structure determination by avoiding acquisition of unneeded data. To accomplish this objective we use a suggested structure of PF2048.1 as an approximate template for its native

structure. Albeit it is clear that the suggested structure is not the native structure, we have mounting evidence that the native structure is less than 4Å away. We argue that our findings from a suggested structure is relevant to the actual structure due to their close structural resemblance.

2 Background and Method

2.1 Residual Dipolar Couplings

RDCs can be acquired via NMR spectroscopy. The theoretical basis of RDC interaction had been established and experimentally observed in 1963 [11]. However, it has only become a more prevalent source of data for structure determination of biological macromolecules in recent years due to availability of alignment media. Upon the reintroduction of order to an isotropically tumbling molecule, RDCs can be easily acquired. The RDC interaction between two atoms in space can be formulated as shown in Eq. (1).

$$D_{ij} = D_{max}\left\langle \frac{3\cos^2(\theta_{ij}(t)) - 1}{2} \right\rangle \quad (1)$$

$$D_{max} = \frac{-\mu_0 \gamma_i \gamma_j h}{(2\pi r)^3} \quad (2)$$

In this equation, D_{ij} denotes the residual dipolar coupling in units of Hz between nuclei i and j. The θ_{ij} represents the time-dependent angle of the internuclear vector between nuclei i and j with respect to the external magnetic field, and the angle brackets signify time averaging. In Eq. (2), D_{max} represents a scalar multiplier dependent on the two interacting nuclei. In this equation, γ_i and γ_j are nuclear gyromagnetic ratios, r is the internuclear distance (assumed fixed for directly bonded atoms), h is the modified Planck's constant and μ_0 represents the permeability of free space.

2.2 REDCRAFT Structural Fitness Calculation

While generating a protein structure from a given set of residual dipolar couplings is nontrivial, it is straightforward to determine how well a given structure fits a set of RDCs. Through algebraic manipulation of Eq. (1) RDC interaction can be represented as shown in Eq. (3),

$$D_{ij} = v_{ij} * S * v_{ij}^T \quad (3)$$

where S represents the Saupe order tensor matrix [11] and v_{ij} denotes the normalized interacting vector between the two interacting nuclei i and j. REDCRAFT takes advantage of this principle by quantifying the fitness of a protein to a given set of RDCs (in units of Hz) and calculating a root-mean-squared deviation as shown in Eq. (4). In this equation D_{ij} and D'_{ij} denote the computed and experimentally acquired RDCs respectively, N, represents the total number of RDCs for the entire protein, and M represents the total number of alignment media in which RDC data have been acquired. In this case a smaller fitness value indicates a better structure.

$$Fitness = \sqrt{\frac{\sum_{j=1}^{M}\sum_{i=1}^{N}(D_{ij} - D'_{ij})^2}{M * N}} \quad (4)$$

The REDCRAFT algorithm and its success in protein structure elucidation has been previously described and documented in detail [4], [9], [10], [12], [13]. Here we present a brief overview. REDCRAFT calculates structures from RDCs using two separate stages. In the first stage (*Stage-I*), a list of all possible discretized torsion angles is created for each pair of adjoining peptide planes. This list is then filtered based on allowable regions within the Ramachandran space [14]. The list of torsion angles that remain are then ranked based on fitness to the RDC data. These lists of potential angle configurations are used to reduce the search space for the second stage.

Stage-II begins by constructing the first two peptide planes of the protein. Every possible combination of angles from *Stage-I* between peptide planes i and $i+1$ are evaluated for fitness with respect to the collected data, and the best n candidate structures are selected, where n denotes the search depth. The list of dihedral angles corresponding to the top n structures are then combined with every possible set of dihedral angles connecting the next peptide plane to the current fragment. Each of these candidate structures is evaluated for fitness and the best n are again selected and carried forward for additional rounds of elongation. All combination of dihedral angles worse than the best n are eliminated, thus removing an exponential number of candidate structures from the search space. This elongation process is repeated iteratively, incrementally adding peptide planes until the entire protein is constructed.

The number of RDCs required to correctly fold a novel protein with a bundle of four nearly parallel helices with REDCRAFT has not been previously examined in a systematic manner. Here we investigate the effect of reducing the available RDCs on the quality of the resulting computational structure. Collecting fewer RDCs per peptide plane can substantially reduce data collection times. In particular, ^{15}N-^{1}H RDCs are easily collected because they avoid expensive ^{13}C labeling. Furthermore, ^{15}N-^{1}H RDC values are typically large in magnitude, reducing the effect of measurement error. C_α- H_α RDCs are large in magnitude but require ^{13}C labeling, complicating sample preparation. RDCs for additional vectors can be collected, but with a decreasing utility and at a greater expense.

2.3 PF2048.1 Protein

The novel protein PF2048.1 is a 9.16 kDa, 71 residue monomeric protein with less than 17% sequence identity to any structurally characterized protein in PDB (as of April, 2015) serves as the primary target of our investigations. PF2048.1 was expressed in *E. coli* as an N-terminal His_6-GB1 fusion that can be efficiently cleaved by TEV protease introducing a single (non-native) Gly residue at position –1. Nearly complete assignments for backbone and sidechain protons, carbons and nitrogens were obtained using standard methods. The resulting 1045 NOE restraints together with TALOS backbone torsion restraints were employed to determine an experimental target structure. Using this reference structure and the structural alignment software 3D-Blast [15] we were able to investigate the structural uniqueness of PF2048.1. Of the resulting proteins 1AEP, a 161 residue apolipoprotein, was identified as the top entry with the highest 3D-Blast score (score of 54.4). We then utilized msTALI [16] to align 1AEP and PF2048.1 based on structural similarity. The final alignment identified 26 residues to be structurally conserved to within 2.9Å between the two proteins, corresponding to about 36% (26 conserved / 71 total residues = 0.36) structural similarity. Figure 1 shows the resulting alignment between the two structures. The two's overall structural deviation was calculated to be 5.265Å.

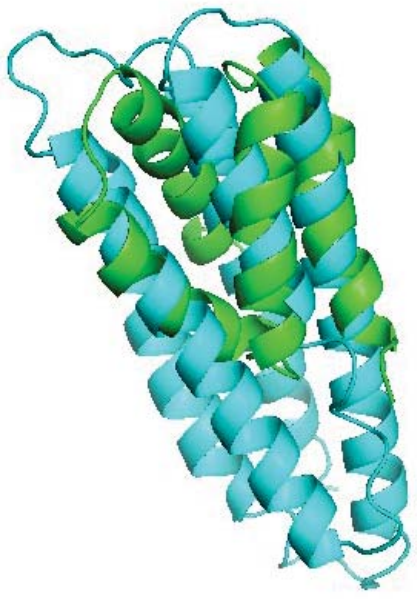


Figure 1. NOE structure of PF2048.1 (green) aligned to 1AEP (blue) using PyMOL. According to PyMOL the two exhibited structural dissimilarity of 5.265Å.

Due to its novelty in both sequence and structure PF2048.1 is an ideal candidate to study the effectiveness of computing protein structure from solely residual dipolar couplings. In addition, the unique arrangement of the helical secondary structural elements of this protein will provide a realistic exploration of the challenges that REDCRAFT will be faced during structure calculation purely from RDCs.

2.4 Simulated RDC Data

Using REDCAT [17], [18] and the reference structure residual dipolar couplings were simulated in two alignment media using the order tensors in Table 1. Error of ±1Hz was uniformly added across all N-H vectors to simulate mild experimental noise in the data sets. Similarly, RDC data from other vectors were distorted by uniformly distributed noise in a range proportional to the expected range of RDCs. These level of random noise for each vector type is shown in Table 2. In addition, Table 2 summarizes the minimum and maximum values corresponding to these order tensors for each RDC vector.

Table 1. Order tensors used for synthetic RDC calculation.

	Sxx	Syy	Szz	Alpha	Beta	Gamma
M1	$3x10^{-4}$	5x10-4	$-8x10^{-4}$	0	0	0
M2	$-4x10^{-4}$	$-6x10^{-4}$	$10x10^{-4}$	40	50	-60

Table 2. Columns 2 and 3 display minimum and maximum RDC values for each vector set using the order tensors in Table 1 in two alignment media (M1 and M2). The last column summarizes the range of uniformly distributed noise that was added to each dataset.

	RDC	Minimum	Maximum	Added noise
M1	N-C	-2.029	1.287	±0.1Hz
	N-H	-18.904	11.815	±1Hz
	C-H	-3.557	5.692	±0.3Hz
	C_α-H_α	-23.32	37.312	±1.97Hz
M2	N-C	-1.544	2.574	±0.1Hz
	N-H	-14.178	23.63	±1Hz
	C-H	-7.115	4.269	±0.3Hz
	C_α-H_α	-46.64	27.984	±1.97Hz

2.5 Evaluation

Our evaluation will proceed by incremental reduction in the data quantity; maintaining the RDC data that are easiest to acquire from NMR spectroscopy. To that end, we will proceed by first eliminating C_α- H_α RDC data from both alignment media since its acquisition increases the cost of protein production significantly. The second phase of our investigation will focus on reducing the RDC data sets from 3 RDCs per alignment medium, to 3 from medium 1 and 1 from medium 2, followed by 2 from medium 1 and 1 from medium 2.

The software REDCRAFT will be utilized for our structure calculation without refinement in any other auxiliary program such as Xplor-NIH[19] or CNS[20]. We anticipated that consistent with principles of Information Theory, more extensive search parameters of REDCRAFT will need to be enabled as a function of reduced datasets to compensate for the absence of information.

The software package PyMOL[21] was utilized in order to calculate the bb-rmsd (backbone root mean squared deviation) between the REDCRAFT structure and the target structure (the NOE structure from which the RDC data were generated). The measure of bb-rmsd is prevalently used to establish the structure similarity between two proteins and values under 3.5Å can signify presence of structural resemblance, while values under 2Å can be interpreted as strong structural resemblance. Our objective is to calculate structures of PF2048.1 using REDCRAFT that exhibit structural similarity to the target protein under 2Å.

The other measure we will use to evaluate structures is the RDC fitness score calculated by REDCRAFT (discussed in detail in section 2.2). This fitness score provides information about how well the RDCs fit the final structure. A score is considered to be of high quality if its score falls at or below the error level of the data (in our case <1Hz). The lower the score the better the structure.

3 Results and Discussion

To evaluate the ability of REDCRAFT to predict the correct structure of PF2048.1, five test cases were established. In each of the cases the amount of data was varied to simulate five different possible data sets. The data sets are summarized in Table 3.

Table 3. Summary of the RDCs used in each experiment.

Set	Medium #	RDCs Utilized
1 (4,4)	1	{C-N, N-H, C-H, C_α-H_α}
	2	{C-N, N-H, C-H, C_α-H_α}
2 (4,1)	1	{C-N, N-H, C-H, C_α-H_α}
	2	{N-H}
3 (3,3)	1	{C-N, N-H, C-H}
	2	{C-N, N-H, C-H}
4 (3,1)	1	{C-N, N-H, C-H}
	2	{N-H}
5 (2,1)	1	{C-N, N-H]
	2	{N-H}

In the sections that follow we will report our findings in each of the cases in Table 3 to evaluate the feasibility of successful protein structure elucidation with the given data set.

3.1 Structure calculation from 4, 4 RDCs

In this experiment the following RDCs corresponding to the vector set {CN, NH, CH, $C_\alpha H_\alpha$} were utilized in two alignment media. The configuration of REDCRAFT is summarized in Table 4 below:

Table 4. Parameters of REDCRAFT for experiment 1 where in the Decimation Parameters C.S. denotes Cluster Sensitivity and S.T. denotes Score Threshold.

Search Depth	Decimation Parameters		Minimization	Lennard Jones Cutoff
200	C.S.	S.T.	Performed every residue	50.0
	4	1.0		

The resulting structure, seen in Figure 2 was measured to have a REDCRAFT fitness score of 0.776 and showed 1.035Å of structural deviation from our target structure.

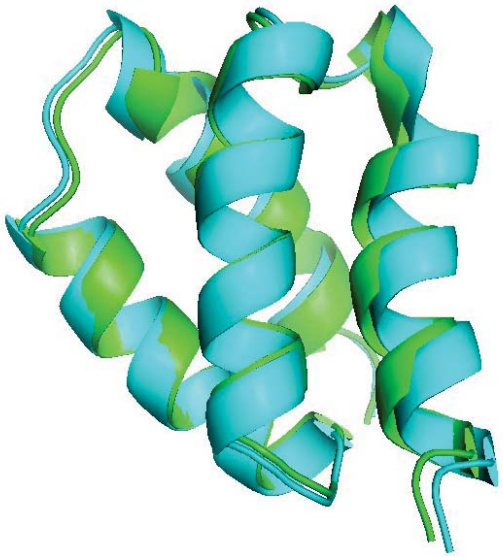


Figure 2. Resulting structure (in green) superimposed to the target target structure (in blue). The two exhibited structural difference of 1.035Å.

3.2 Structure calculation from 4,1 RDCs

In this experiment two different sets of RDC data were used in both alignment media. The first set contained four vectors {CN, NH, CH, $C_\alpha H_\alpha$} and the second just one vector set {NH}. The corresponding REDCRAFT parameters for this exercise are summarized in Table 5. Consistent with our expectation, due to the reduction in data

quantity, a more thorough search by REDCRAFT was required in order to achieve a comparable result to that of the (4,4) exercise. The more thorough search was achieved through the adjustment of the C.S. an S.T. terms. The adjustment of these two terms allow for a more refined clustering of the search space as a function of reduced dataset *N* in Eq. (4).

Table 5. Parameters of REDCRAFT for experiment 2.

Search Depth	Decimation Parameters		Minimization	Lennard Jones Cutoff
200	C.S.	S.T.	Performed every 3rd residue	50.0
	3	0.8		

The resulting structure (seen in Figure 3) exhibited a RDC fitness score of 0.741 and a bb-rmsd of 1.594Å with respect to the target structure.

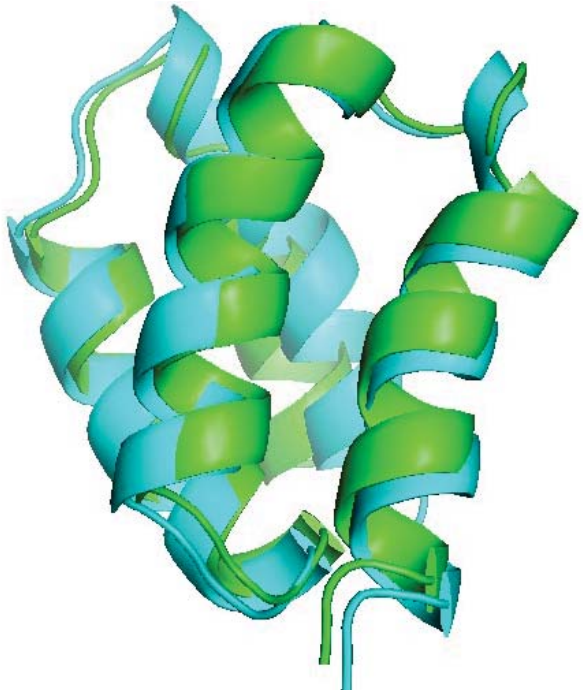


Figure 3. Resulting structure (in green) superimposed to the target structure (in blue). The two showed structural deviation of 1.594Å.

3.3 Structure calculation from 3,3 RDCs

In this experiment two sets of three RDCs {CN, NH, CH} were utilized. Several REDCRAFT configurations (similar to those in experiment 1 and 2) were attempted on this dataset but it became clear that there was something inherently anomalous about constructing a protein with these two particular sets of RDCs. As a result of these difficulties we were forced to incorporate additional secondary structural information and perform a more directed folding process. In our case the phi and psi angles were restricted to oscillate in the range of [-60:-50] and [-50:-40] respectively for the helical residues 3-16, 22-35, 39-52 and 57-70. The addition of secondary structural constraints can easily be facilitated through the use of secondary structure prediction tools such as Jpred, Jpred3 and I-TASSER[22]–[24], or through early interpretation of the data available from NMR spectroscopy without imposing any additional data acquisition costs.

The resulting structure (seen in Figure 4) had a RDC fitness score of 0.382 and a bb-rmsd of 1.002Å with respect to the target structure.

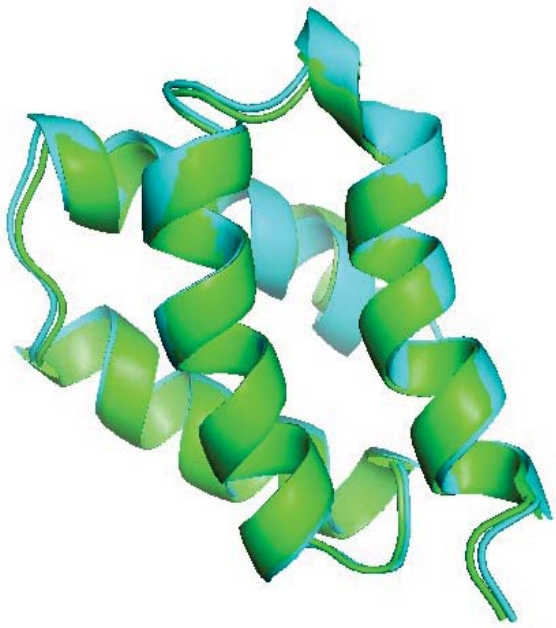


Figure 4. Resulting structure (in green) aligned to the target structure (in blue). The two showed structural deviation of just 1.001Å.

3.4 Structure calculation from 3,1 RDCs

In this experiment two different sets of RDCs were used; the first set containing three vectors {CN, NH, CH} and the second containing just one vector {NH}. The REDCRAFT configuration is summarized in the Table 6 below:

Table 6. Parameters of REDCRAFT for experiment 4.

Search Depth	Decimation Parameters		Minimization	Lennard Jones Cutoff
200	C.S.	S.T.	Performed every 3rd residue	50.0
	3	1.0		

Surprisingly, this combination of data (although a subset of the 3,3 exercise) was less refractory and did not require the incorporation of dihedral restraints or

modification of search parameters in order to perform a more extensive search of the solution space. The resulting structure, as seen in Figure 5, exhibits a RDC fitness score of 0.741 and bb-rmsd from the target structure of 1.594Å, mirroring the results in 3.2.

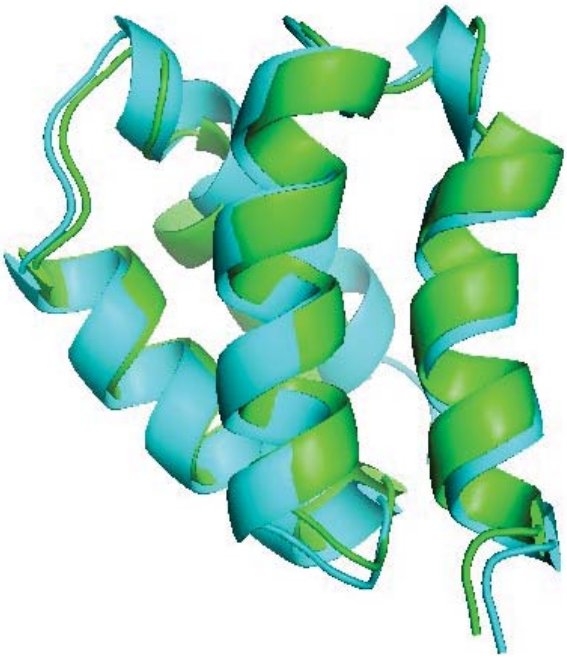


Figure 5. Resulting structure (in green) superimposed to the target structure (in blue). As in experiment 2, the two exhibited a backbone RMSD of 1.594Å.

3.5 Structure calculation from 2,1 RDCs

The final experiment in establishing the boundaries of data requirement is based on {CN, NH} and {NH}. Due to further reduction of the datasets we were again forced to incorporate secondary structure constraints along with the following REDCRAFT parameters summarized in Table 7. The ranges for the secondary structure constraints remained the same as that of the experiment described in 3.3.

Table 7. REDCRAFT parameters for experiment 5 utilizing 2,1 RDC sets resulting in a structure 3.03Å from the target structure.

Search Depth	Decimation Parameters		Minimization	Lennard Jones Cutoff
200	C.S.	S.T.	Performed every residue	50.0
	3	0.5		

The resulting structure in this experiment showed structural deviation from the target structure of 3.03Å—a bb-rmsd that indicates need for further refinement. Careful investigation of the changes in RDC fitness scores revealed that midway through the last helix (around residue 64) there was a significant spike in fitness to RDC data (as seen in Figure 6). This prompted a fragmented study of this protein where the structure is determined in two contiguous segments. Since the spike occurred in the middle of a helix, we chose to terminate the first segment at residue 57 (the beginning of the affected helix) in an attempt to conserve secondary structure elements as much as possible. This approach yielded two fragments [1:56] and [57:72] having bb-rmsd's to the target structure of 0.465Å and 0.724Å respectively. Using RDCs to predict the orientation of the two fragments (as previously shown in theory [25]) we properly oriented and connected the two fragments. The resulting structure (seen in Figure 7) exhibited a RDC fitness score of 0.173 and bb-rmsd of 0.533Å to the target structure.

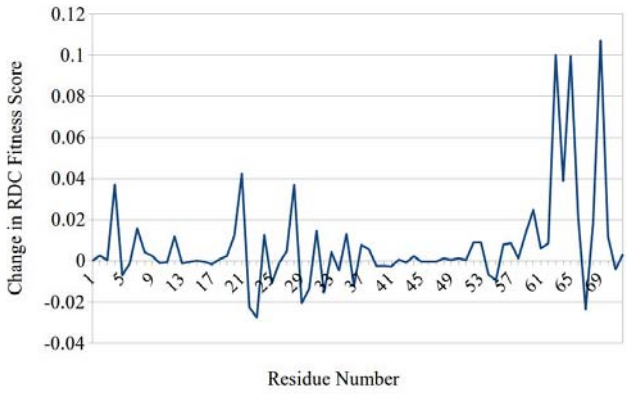


Figure 6. Graph showing the change in RDC fitness (y-axis) throughout the 72 residues (x-axis). A spike can be seen to occur at residue 64.

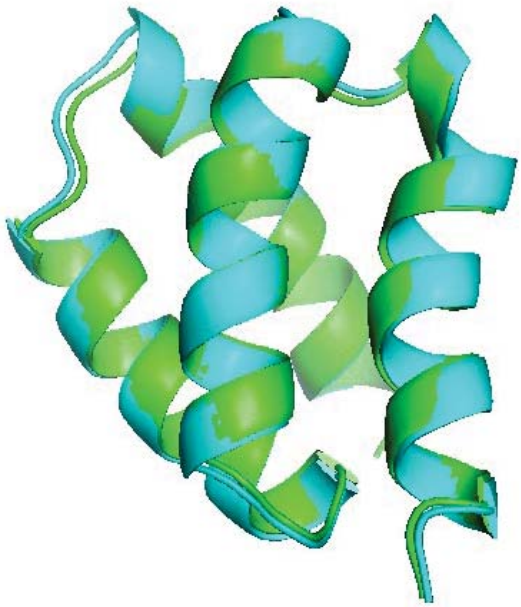


Figure 7. Resulting structure (green) from experiment 5 superimposed to the target structure (blue). The structural deviation between the two was calculated to be 0.533Å.

4 Conclusion

Exploration of the minimum data requirement is useful in order to establish the expected financial cost of a protein's structure determination. An exploration mechanism such as the one presented here will allow for appropriate allocation of funds as a function of a protein's medical or biological importance. This is a critical contribution to the repertoire of structure determination approaches especially in the context of personalized medicine where funds can be appropriate allocated toward culprit proteins in human diseases.

Our investigation through exploration of the five exercises listed in the previous section, has revealed with high degree of certainty that structure determination of PF2048.1 can be accomplished with as little as {CN, NH} and {NH} from two alignment media respectively. In addition, we believe that more thorough exploration of REDCRAFT's search options, combined with addition of readily available restraints (such as dihedral restraints) can reduce the needed dataset further. This expectation is rooted on the observance of the results from the {3,3} and {2,1} exercises where dihedral restraints were included as part of REDCRAFT's analysis. Inclusion of dihedral restraint not only helped to recover the structure of PF2048.1, but it produced the most accurate structure (to within 1.001Å of the target protein in the case of {3,3}).

Of notable interest is the anomalous nature of structure determination from the set {3,3} compared to that of {3,1}. The refractory nature of this dataset is peculiar and in contradiction with the principles of information theory. In principle, inclusion of data should not harm the outcome unless the included data introduces a level of noise that is nonuniform and more corrupt in nature than the remainder of the data. There is however the possibility of existing inherent degeneracies from the aforementioned set of vectors that when combined with the heuristics of REDCRAFT, produce the observed anomalies. Our future work will investigate these two conjectures.

Our future investigation is to determine the solution state structure of the protein PF2048.1 from experimental data. Our approach will leverage the conclusions of this work in order to acquire the least amount of data compared to the traditional approach of acquiring the most complete dataset. We are confident that our new approach will reduce the quantity of acquired data by nearly 90% and therefore result in significant reduction in financial and temporal cost of protein structure determination by NMR spectroscopy. Although base on the results reported here, structure determination should be plausible with {CN, NH} & {NH} datasets, our experimental investigation of this protein will proceed based on acquisition of the {CN, NH, CH} & {NH} as preparation for missing and noisy data.

5 Acknowledgements

This work was supported by NIH Grant Numbers 1R01GM081793 and P20 RR-016461 to Dr. Homayoun Valafar.

6 Bibliography

[1] R. F. Service, “Structural genomics - Tapping DNA for structures produces a trickle,” vol. 298, no. 5595, pp. 948–950, 2002.

[2] R. Service, “Structural biology - Structural genomics, round 2,” *Science (80-.).*, vol. 307, no. 5715, pp. 1554–1558, 2005.

[3] H. M. Berman, J. D. Westbrook, M. J. Gabanyi, W. Tao, R. Shah, A. Kouranov, T. Schwede, K. Arnold, F. Kiefer, L. Bordoli, J. Kopp, M. Podvinec, P. D. Adams, L. G. Carter, W. Minor, R. Nair, and J. La Baer, “The protein structure initiative structural genomics knowledgebase.,” *Nucleic Acids Res.*, vol. 37, no. Database issue, pp. D365–8, Jan. 2009.

[4] M. Simin, S. Irausquin, C. A. Cole, and H. Valafar, “Improvements to REDCRAFT: a software tool for simultaneous characterization of protein backbone structure and dynamics from residual dipolar couplings,” *J. Biomol. NMR*, vol. 60, pp. 241–264, Nov. 2014.

[5] F. Delaglio, G. Kontaxis, and A. Bax, “Protein Structure Determination Using Molecular Fragment Replacement and NMR Dipolar Couplings,” *J. Am. Chem. Soc.*, vol. 122, no. 9, pp. 2142–2143, Mar. 2000.

[6] J.-C. Hus, D. Marion, and M. Blackledge, “Determination of protein backbone structure using only residual dipolar couplings.,” *J. Am. Chem. Soc.*, vol. 123, no. 7, pp. 1541–2, Feb. 2001.

[7] M. Andrec, P. C. Du, and R. M. Levy, “Protein backbone structure determination using only residual dipolar couplings from one ordering medium.,” *J. Biomol. NMR*, vol. 21, no. 4, pp. 335–347, Dec. 2001.

[8] J. Zeng, J. Boyles, C. Tripathy, L. Wang, A. Yan, P. Zhou, and B. R. Donald, “High-resolution protein structure determination starting with a global fold calculated from exact solutions to the RDC equations.,” *J. Biomol. NMR*, vol. 45, no. 3, pp. 265–81, Nov. 2009.

[9] M. Bryson, F. Tian, J. H. Prestegard, and H. Valafar, “REDCRAFT: a tool for simultaneous characterization of protein backbone structure and motion from RDC data.,” *J. Magn. Reson.*, vol. 191, no. 2, pp. 322–34, Apr. 2008.

[10] H. Valafar, M. Simin, and S. Irausquin, “A Review of REDCRAFT: Simultaneous Investigation of Structure and Dynamics of Proteins from RDC Restraints,” *Annu. Reports NMR Spectrosc.*, vol. 76, pp. 23–66, 2012.

[11] A. Saupe and G. Englert, "High-Resolution Nuclear Magnetic Resonance Spectra of Orientated Molecules," *Phys. Rev. Lett.*, vol. 11, no. 10, pp. 462–464, Nov. 1963.

[12] P. Shealy, M. Simin, S. H. Park, S. J. Opella, and H. Valafar, "Simultaneous structure and dynamics of a membrane protein using REDCRAFT: membrane-bound form of Pf1 coat protein.," *J. Magn. Reson.*, vol. 207, no. 1, pp. 8–16, Nov. 2010.

[13] E. Timko, P. Shealy, M. Bryson, and H. Valafar, "Minimum Data Requirements and Supplemental Angle Constraints for Protein Structure Prediction with REDCRAFT," in *BIOCOMP*, 2008, pp. 738–744.

[14] G. N. Ramachandran, C. Ramakrishnan, and V. Sasisekharan, "Stereochemistry of polypeptide chain configurations," *J. Mol. Biol.*, vol. 7, no. 1, pp. 95–99, Jul. 1963.

[15] L. Mavridis and D. W. Ritchie, "3D-Blast: 3D Protein Structure Alignment, Comparison, and Classification Using Spherical Polar Fourier Correlations.," *Pac. Symp. Biocomput.*, pp. 281–292, 2010.

[16] P. Shealy and H. Valafar, "Multiple structure alignment with msTALI.," *BMC Bioinformatics*, vol. 13, no. 1, p. 105, May 2012.

[17] H. Valafar and J. H. Prestegard, "REDCAT: a residual dipolar coupling analysis tool.," *J. Magn. Reson.*, vol. 167, no. 2, pp. 228–41, Apr. 2004.

[18] C. Schmidt, S. J. Irausquin, and H. Valafar, "Advances in the REDCAT software package.," *BMC Bioinformatics*, vol. 14, no. 1, p. 302, Oct. 2013.

[19] C. D. Schwieters, J. J. Kuszewski, N. Tjandra, and G. M. Clore, "The Xplor-NIH NMR molecular structure determination package.," *J. Magn. Reson.*, vol. 160, no. 1, pp. 65–73, Jan. 2003.

[20] A. T. Brunger, P. D. Adams, G. M. Clore, W. L. DeLano, P. Gros, R. W. Grosse-Kunstleve, J. S. Jiang, J. Kuszewski, M. Nilges, N. S. Pannu, R. J. Read, L. M. Rice, T. Simonson, G. L. Warren, and A. T. Brünger, "Crystallography & NMR system: A new software suite for macromolecular structure determination.," *Acta Crystallogr. D. Biol. Crystallogr.*, vol. 54, no. Pt 5, pp. 905–21, Sep. 1998.

[21] W. L. DeLano, "The PyMOL Molecular Graphics System," *DeLano Sci. LLC, Palo Alto, CA, USA. http//www.pymol.org*, 2008.

[22] J. a. Cuff, M. E. Clamp, A. S. Siddiqui, M. Finlay, and G. J. Barton, "JPred: A consensus secondary structure prediction server," *Bioinformatics*, vol. 14, no. 10, pp. 892–893, 1998.

[23] C. Cole, J. D. Barber, and G. J. Barton, "The Jpred 3 secondary structure prediction server.," *Nucleic Acids Res.*, vol. 36, no. Web Server issue, pp. 197–201, 2008.

[24] Y. Zhang, "I-TASSER server for protein 3D structure prediction," *BMC Bioinformatics*, vol. 9, p. 40, 2008.

[25] H. M. Al-Hashimi, H. Valafar, M. Terrell, E. R. Zartler, M. K. Eidsness, and J. H. Prestegard, "Variation of molecular alignment as a means of resolving orientational ambiguities in protein structures from dipolar couplings.," *J. Magn. Reson.*, vol. 143, no. 2, pp. 402–6, Apr. 2000.

An improved Stochastic Proximity Embedding to Protein Structure Determination

Ivan S. Sendin[1], **Indiara B. Vale**[2] **and Marcos A. Batista**[2]
sendin@ufu.br, indiarabarbosavale@gmail.com, marcos.batista@pq.cnpq.br
[1]FACOM, Federal University of Uberlândia, Uberlândia, MG, Brasil
[2]IBiotec, Federal University of Goiás, Catalão, GO, Brasil
Av. Dr. Lamartine Pinto de Avelar, 1120, CEP 75704-020

Abstract— *The Distance Geometry Problem (DGP) is defined as the problem of finding the spatial representation of a set of points, given the distances between them. An existing method in the literature that propose to solve this problem is the Stochastic Proximity Embedding (SPE), a simple, robust and self-organizing method that starts with an initial random configuration, selects pairs of points at random and adjust their distances based on a learning rate. The purpose is generate results that are close to the given set of relations between the objects. Therefore, this method is not fully effective to deal with uncertain distances, as expected in real world situation.*

The Molecular Distance Geometry Problem (MDGP) arises from the protein structure determination problem, and consists in finding Cartesian coordinates of the atoms in a molecule, based on the set of some interval distances obtained by Nuclear Magnetic Resonance (NMR). In this work, we propose a method to address the MDGP based on SPE. To determine one protein structure, a small subset of atoms is selected and optimized by an interval distance version of SPE. Iteratively, this subset is increased until the full protein is determined. We applied this method on artificially created NMR data and obtained significant results.

Keywords: Protein Structure, NMR, Distance Geometry

1. Introduction

The Distance Geometry Problem (DGP) arises on the need to determine the coordinates for a set of objects geographically distributed using an incomplete and imprecise set of distances. Given an integer $K > 0$, the problem is embedding a simple undirected graph $G = (V, E, r)$, whose edges are weighted by a nonnegative function $r : E \to \mathbb{R}_+$. Thus, it is necessary to find a realization $x : V \to \mathbb{R}^K$ such as the Euclidean distance d, among the pair of points $\{i, j\}$, be equal to the edges weight:

$$\forall \{i, j\} \in E, \|x(i) - x(j)\| = r(\{i, j\}) \tag{1}$$

Through this article we will adopt $\{x_i, x_i, r_{ij}\}$ instead of $\{x(i), x(j), r(\{i, j\})\}$.

The graph embedding problem is *NP-Complete* in linear case and *NP-Hard* for $K > 1$ [19], although this problem can be addressed in linear time for a sufficient dense graph [5].

The problem is is present in several notable areas, like sensor network localization [9], architecture [11], [8] and protein structure determination [10], [7], [14], this work focus on protein structure determination.

The word *protein* derives from the Greek *protos*, meaning *"primary"*, *"most important"* or *"standing in front"* [18]. These macromolecules are composed of a long polypeptide chain, molding a complex and stable structure. The protein conformation is essential to determine its functionality and they are used in almost all essentials biological processes, e.g., transporting oxygen from the lungs to other organs and tissues in all vertebrates.Therefore, knowledge about the protein structure is essential to analyze and manipulate it, considering all the possible interactions between molecules.

Currently, there are two major methods to determine protein structures: X-ray crystallography and Nuclear Magnetic Resonance (NMR). The x-ray crystallography works with cristalized proteins; using x-ray beam to determine the density of electrons and after that, the protein structure. The protein crystallization process imposes some restrictions not present in NMR method.

Using NMR method, the protein is submitted to an external magnetic field which induces the alignment of atoms spin in the observed nuclei. The interference on those spins can be measured hence their distance.

The NMR measurements are not precise and the resulting distance is a interval value, typically one angstrom wide, of some atoms. Along with NMR imprecise distances, the distances of atoms separated by one or two covalent bonds can be determined precisely, once the chemical composition of the protein is known and the covalent bonds and angles are stable.

Fortunately, proteins chemical composition are known, consequently providing known distances. For example, if two atoms are chemically bonded or bonded to a common atom, it is possible to determine their relative distance. That distance is not precise, but can be considered fixed and the small variations can be used in NMR experiments.

Thus, this method provides an interval weighted graph, which represents the backbone with a sparse set of its uncertain distances. The final possible protein conformation is determined by solving a molecular distance geometry problem.

The Molecular Distance Geometry Problem (MDGP) is a variation of DGP with some interval distances

$$l_{ij} \leq r_{ij} \leq u_{ij} \tag{2}$$

where l_{ij} and u_{ij} are the lower and upper bounds for the distance from atom i do j. Therefore one realization $x : V \rightarrow \mathbb{R}^3$ is

$$\forall \{i,j\} \in E, l_{ij} \leq \|x(i) - x(j)\| \leq u_{ij} \tag{3}$$

where each atom is associated with one vertex and the known distances - accurate or interval - to one edge connecting the respective vertices. Like DGP,MDGP can be solved in linear time given enough precise distances, but using only NMR imprecise distances and covalent bonds precise distances and considering the experimental errors [6], as suggested in [4] the solution for MDGP can be formulated as a global optimization problem:

$$\begin{aligned} min_{x \in \mathbb{R}^{Kn}} \sum_{\{i,j\} \in E} & min^2(\frac{\|x_i - x_j\|^2 - l_{ij}}{l_{ij}}, 0) + \\ & max^2(\frac{\|x_i - x_j\|^2 - u_{ij}}{u_{ij}}, 0) \end{aligned} \tag{4}$$

This problem can also be called of Interval Molecular Distance Geometry Problem (iMDGP) [16]. The following are some methods to solve the MDGP.

The Geometry Build-Up algorithm (GBU) [5] uses four non-planar atoms, called geometric base, and calculate their coordinates. Then, it proceeds iteratively solving linear systems and determining the position of the remaining atoms in the molecule, given the distances between the base atoms and the atom to be determined. In [20] was proposed a new version of GBU, that can minimize the effects of errors caused by floating point operations.

The DGSOL method [17] aims to show that continuation algorithms, based on Gaussian smoothing, can be used to develop an efficient code for the solution of DGP. The algorithm searches for a global minimizer of the function based on the Gauss-Hermite transform. The problem of DGSOL is the dependence of structures on the distance data.

Branch-and-Prune (B&P) is another iterative method to address the MDGP [15]. It uses 3 already positioned atoms and three known distances to determine the position of the next atom until the entire protein is determined. As three atoms are not sufficient to uniquely determine to position of one point in R^3, each calculated atom can assume two distinct positions. At each step a new search branch is created, so the size of the solution can increase exponentially. The pruning is performed after each step, another available distances are used to cut the branches:

1) both positions are possible: the booth branches are explored;
2) only one branch is possible: the possible branch is stored and the other is pruned;
3) neither position is possible: the both branches are pruned and the search is backtracked.

The idea of B&P is use only exact distances - from molecular geometry - to position the atoms and use the intervalar distances - from NMR - in the pruning process.

An overview of the presented methods can be found in [16]. In this article we will use the Stochastic Proximity Embedding (SPE) as base for the development of new methods.

1.1 Stochastic Proximity Embedding

Stochastic Proximity Embedding (SPE), introduced in [2], [1], creates one embedding trough a continuous optimization. Given n objects and a set of expected distances r, the method starts with an initial random configuration and iteratively refines it by repeatedly selecting two points $\{u, v\}$ at random. The distance d_{uv} is calculated and the their coordinates are updated as follow:

$$x_i \leftarrow x_i + \lambda \frac{r_{ij} - d_{ij}}{4 d_{ij}} (x_i - x_j) \tag{5}$$

$$x_j \leftarrow x_j + \lambda \frac{r_{ij} - d_{ij}}{4 d_{ij}} (x_j - x_i) \tag{6}$$

The λ is the *learning rate*, used to avoid oscillations, it starts with 1 and decreases until the value becomes close enough to 0. This process is described in Algorithm 1.

Data: protein graph, λ, λ_Δ, C
Result: protein structure
while $\lambda \geq 0$ **do**
  for *($i = 0$; $i \leq C$; $i = i + 1$)* **do**
    random selection of u and v;
    update u and v;
  end
  $\lambda \leftarrow \lambda - \lambda_\Delta$;
end

Algorithm 1: SPE method

This article differs from others existing in the literature because we present solutions to the MDGP with uncertain data. The rest of the paper is organized as follows. Section 2 describes the proposed methods, Section 3 presents the computational experiments performed and their results. In Section 4 the conclusions are presented.

2. Proposed Methods

Our method consists in a structural determination based on interval inter-atomic distances. That process is very important when we consider that in real world, the data obtained in NMR are ruled by minimum and maximum constraints: the distance relations, before exacts, now have minimum (r_{uv}^{min}) and maximum (r_{uv}^{max}) values.

2.1 Stochastic Proximity Embedding Interval

The Stochastic Proximity Embedding Interval (SPE_i) is similar to the original SPE [2], [1]; but the update function is modified to work with intervalar constraints (r_{ij}^{min}, r_{ij}^{max}):

If $d_{ij} < r_{ij}^{min}$:

$$x_i \leftarrow x_i + \lambda \frac{r_{ij}^{min} - d_{ij}}{4d_{ij}}(x_i - x_j) \tag{7}$$

$$x_j \leftarrow x_j + \lambda \frac{r_{ij}^{min} - d_{ij}}{4d_{ij}}(x_j - x_i) \tag{8}$$

If $d_{ij} > r_{ij}^{max}$:

$$x_i \leftarrow x_i + \lambda \frac{r_{ij}^{max} - d_{ij}}{4d_{ij}}(x_i - x_j) \tag{9}$$

$$x_j \leftarrow x_j + \lambda \frac{r_{ij}^{max} - d_{ij}}{4d_{ij}}(x_j - x_i) \tag{10}$$

The remaining of the method is the same as presented in [2], [1].

2.2 Progressive SPE

In this section we introduce a progressive version of SPE (SPE_P). To develop this variation we considered:

- The atoms of an protein are ordered, following the backbone. We use this fact and restrict the selection of i and j to the first S atom of the protein. The value of S is increased until its achieves the size of protein;
- Using the ordering, we introduce the notion of *older* and *newer* atoms.When we update the i and j atoms, only the newer atom is changed.

The *update* function, reformulated:

If $d_{ij} < r_{ij}^{min}$:

$$x(j) \leftarrow x(j) + \lambda \frac{r_{ij}^{min} - d_{ij}}{2d_{ij}}(x_j - x_i) \tag{11}$$

If $d_{ij} > r_{ij}^{max}$:

$$x(j) \leftarrow x(j) + \lambda \frac{r_{ij}^{max} - d_{ij}}{2d_{ij}}(x_j - x_i) \tag{12}$$

The original algorithm was also modified and presented in Algorithm 2.

Data: protein graph, λ, λ_Δ, C, p;
Result: protein structure

```
while S ≤ T do
    while λ ≥ 0 do
        for (i ← 0; i ≤ C; i++) do
            random selection of i;
            random selection of j ≠ i;
            if i > j then
                (i, i) ← (j, i)
            end
            update i and j;
        end
        λ ← λ − λ_Δ;
    end
    S ← S + p;
end
```

Algorithm 2: Progressive Method

2.3 Sliding Window

The last method (SPE_{SW}) is based on sliding windows: the range for sampling is limited in its minimum and maximum, a parameter p is defined to determine the variation of the sliding window (see Algorithm 3), the first atom chosen is sampled from $S/2$ to S range - using the backbone ordering; the second one is any neighbor of the first. The *update* function is the same used in SPE_P.

Data: protein graph, λ, λ_Δ, C, p;
Result: protein structure

```
S ← p;
while S ≤ T do
    while λ ≥ 0 do
        for (k ← 0; k ≤ C; k++) do
            i ← random point of [S/2, S];
            j ← random neighbor of i;
            update i and j;
        end
        λ ← λ − λ_Δ;
    end
    S ← S + p;
end
```

Algorithm 3: Sliding Window method

This method can efficiently solve problems with sparse and inexact data, providing satisfactory results (Section 3).

3. Computational Experiments

To perform the experiments, we used artificial backbones [12] and simulated NMR experiments using this rules: with atoms separated up to 5 Å, we use a interval distance between 2 to 3 Å, 3 to 4 Å and 4 to 5 Å, according to the observed real distance. Also, exact distances for atoms separated up to 2 covalent bonds [].

Table 1: Experimental results obtained from SPE_i, based on different initial configuration.

T[1]	C[2]	RMSD^3_f	Restri.^4_f(%)	LDE^5_f	Time(s)[6]
25	6000	1.860	0.519	0.012	300
50	6000	2.772	0.676	0.002	120
100	10000	5.552	0.582	0.016	180
250	10000	16.835	0.471	0.030	240

[1]: protein size [2]: number of cycles
[3]: final value of RMSD [4]: final value of Restrictions $\times 100$
[5]: final value of ScoreLDE [6]: total time of the process

To evaluate the results, we applied the following metrics:

- **Root-mean-square Deviation (RMSD)** The RMSD is used to measure the similarity of two proteins, hence is commonly used to quantify the quality of a generated protein given a correct one[3]. The RMSD of two structures is computed as:

$$\frac{1}{n}\sqrt{\sum_{i=1}^{n}||x_i - y_i||} \tag{13}$$

  with n the size of the protein, $\{x_i\}$ is the original protein - the artificial backbone - and $\{y_i\}$ is the generated protein rotated and translated to minimize the resulting RMSD;
- **Satisfied constraints percentage** Given one protein is straightforward to verify if each distance constraint is satisfied. For interval distances the distance is calculated and verified if its lays on the interval; for exact ones we considered a tolerance of 0.1 Å;
- **Largest Distance Error (LDE)** LDE is one of the most used penalty function for the MDGP [13] and is defined as

$$\sum_{\{i,j\}\in E} min(\frac{\|x_i - x_j\| - l_{ij}}{l_{ij}}, 0) + max(\frac{\|x_i - x_j\| - u_{ij}}{u_{ij}}, 0). \tag{14}$$

All the algorithms were implemented in Python and the experiments performed on Intel Core i3 of 2nd generation, with 4GB RAM running Linux Ubuntu version 12.04.

The tests were performed with λ starting at 1.0 and λ_Δ at 0.001, these values are obtained empirically. The results for $SPE_I$are presented in Table 1 and one result is showed in Figure 1.

The results for SPE_P and SPE_{SW} are in Table 2 and Table 3, respectively. In Figure 2 we show one result with 25 atoms for SPE_P. In Figure 3

In Table 2 we can see better results to the application of PM in the same sample size of Tab. 1, that can be confirmed comparing Fig. 1 and Fig. 2.

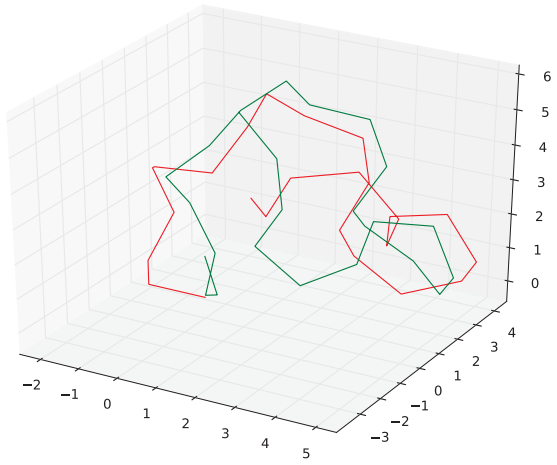


Fig. 1: In red the original protein with 25 atoms and in green the protein produced by SPE_I with $C = 6.000$

Table 2: Results from Progressive Method with 6000 cycles, now introducing p like was showed in Eq. 2.

T[1]	p[2]	RMSD^3_f	Restric^4_f(%)	ScoreLDE^5_f	Time(s)[6]
25	$2i+10$	0.492	0.648	0.008	720
50	$2i+5$	2.986	0.531	0.014	600
100	$2i+5$	3.594	0.596	0.019	2700
250	$2i+5$	6.002	0.460	0.054	10800

[1]: protein size [2]: increase rate of protein size
[3]: final value of RMSD [4]: final value of Restrictions $\times 100$
[5]: final value of ScoreLDE [6]: total time of the process

4. Conclusions

We presented three different methods, based on SPE, to solve MDGP with interval distances. The regular SPE method with intervalar distances showed non-scalable. The SPE_{SW} and SPE_P showed up more robust and able to solve successfully the problem for proteins of up to 250 atoms.

Acknowledgments

The authors would like to thank the Brazilian National Council for Scientific and Technological Development (CNPq), Research Support Foundation of Goiás State (FAPEG) and Federal University of Uberlândia (PROPP grant 04/1014/074).

References

[1] Dimitris K. Agrafiotis, Deepak Bandyopadhyay, and Eric Yang. Stochastic proximity embedding: A simple, fast and scalable algorithm for solving the distance geometry problem. In A. Mucherino, Leo Liberti, Carlile Lavor, and N. Maculan, editors, *Distance Geometry: Theory, Methods, and Applications*. Springer London, Limited, 2012.

[2] Dimitris K. Agrafiotis, H. Xu, F. Zhu, D. Bandyopadhyay, and P. Liu. Stochastic proximity embedding: Methods and applications. *Molecular Informatics*, 29:758–770, 2010.

Table 3: Final results of Sliding Window method with different numbers of cycles and $p = 2i + 10$. The method was also applied to a protein with 300 atoms.

T[1]	C[2]	RMSD$_f^3$	Restric$_f^4$(%)	ScoreLDE$_f^5$	Time(s)[6]
25	6000	1.810	0.357	0.054	600
50	6000	5.516	0.519	0.041	1800
100	6000	4.990	0.510	0.034	2400
250	10000	8.808	0.499	0.036	21000
300	10000	14.015	0.459	0.056	28200

[1]: protein size [2]: number of cycles
[3]: final value of RMSD [4]: final value of Restrictions $_{\times 100}$
[5]: final value of ScoreLDE [6]: total time of the process

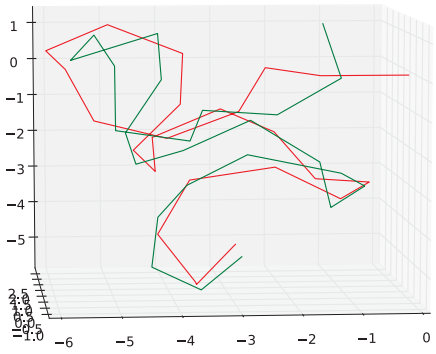


Fig. 2: In red the original protein with 25 atoms and in green the protein produced by SPE_P with $C = 6.000$.

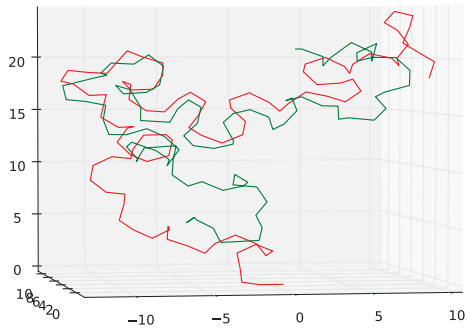


Fig. 3: In red the original protein with 100 atoms and in green the protein produced by SPE_{SW} with $C = 6.000$

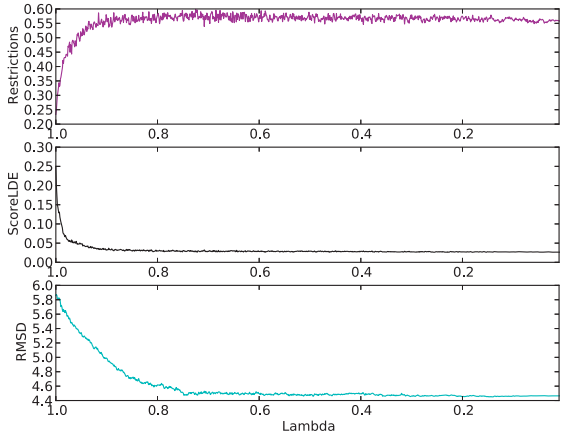


Fig. 4: Values of tests results evaluation on a protein structure with 25 atoms, using PM, $C = 6,000$ and $p = 2i + 10$.

[3] Fred E. Cohen and Michael J.E. Sternberg. On the prediction of protein structure: The significance of the root-mean-square deviation. *Journal of Molecular Biology*, 138(2):321 – 333, 1980.

[4] Gordon M Crippen and Timothy F Havel. *Distance geometry and molecular conformation*, volume 74. Research Studies Press Taunton, England, 1988.

[5] Qunfeng Dong and Zhijun Wu. A geometric build-up algorithm for solving the molecular distance geometry problem with sparse distance data. *Journal of Global Optimization*, 26(3):321–333, 2003.

[6] Jack D. Dunitz. Distance geometry and molecular conformation, by g. m. crippen and t. f. havel, research studies press, taunton, england, john wiley and sons, new york, 1988. pp. 541 + x pp. *Journal of Computational Chemistry*, 11(2):265–266, 1990.

[7] I. Z. Emiris and B. Mourrain. Computer algebra methods for studying and computing molecular conformations. *Algorithmica*, 25(2-3):372–402, 1999.

[8] D.G. Emmerich. *Structures tendues et autotendantes*. Monographies de géometrie constructive. Ecole d'architecture de Paris la Villette, 1988.

[9] T. Eren, O.K. Goldenberg, W. Whiteley, Y. R. Yang, A.S. Morse, B. D O Anderson, and P.N. Belhumeur. Rigidity, computation, and randomization in network localization. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 4, pages 2673–2684 vol.4, March 2004.

[10] Donald J. Jacobs and et al. Protein flexibility prediction using graph theory. *Proteins Structure Function and Genetics*, 44(2):150–165, 2001.

[11] Valentín Gómez Jáuregui. *Tensegrity Structures and Their Application to Architecture*. Ediciones Universidad de Cantabria, 2010.

[12] Carlile Lavor. On generating instances for the molecular distance geometry problem. In *Global Optimization: From Theory to Implementation, Nonconvex Optimization and Its Application Series*, pages 405–414. Springer, 2006.

[13] Carlile Lavor, Leo Liberti, and Nelson Maculan. Molecular distance geometry problem. In Christodoulos A. Floudas and Panos M. Pardalos, editors, *Encyclopedia of Optimization*, pages 2304–2311. Springer US, 2009.

[14] Carlile Lavor, Antonio Mucherino, Leo Liberti, and Nelson Maculan. On the computation of protein backbones by using artificial backbones of hydrogens. *Journal of Global Optimization*, 50(2):329–344, 2011.

[15] Leo Liberti, Carlile Lavor, and Nelson Maculan. A branch-and-prune algorithm for the molecular distance geometry problem. *International Transactions in Operational Research*, 15(1):1–17, 2008a.

[16] Leo Liberti, Cerlile Lavor, Nelson Maculan, and Antonio Mucherino. Euclidean distance geometry and applications. *ArXiv e-prints*, 2012.

[17] Jorge Moré and Zhijun Wu. Distance geometry optimization for protein structures. *Journal of Global Optimization*, 15(3):219–234, 1999.

[18] Robert Roskoski. Nature's robots: A history of proteins: Tanford, c., reynolds, j. *Biochemistry and Molecular Biology Education*, 30(5):343–345, 2002.

[19] J. Saxe. Embeddability of weighted graphs in k-space is strongly np-hard. *Proceedings of 17th Allerton Conference in Communications, Control and Computing*, pages 480–489, 1979.

[20] Di Wu and Zhijun Wu. An updated geometric build-up algorithm for solving the molecular distance geometry problems with sparse distance data. *Journal of Global Optimization*, 37(4):661–673, 2007.

Chinese Whispers for Protein-Protein Interaction Network Analysis to Discover Overlapping Functional Modules

Ying Liu

[1]Department of Computer Science, Mathematics and Science, College of Professional Studies, St. John's University, Queens, NY 11439

Abstract *- One of the most pressing problems of the post genomic era is identifying protein functions. Clustering Protein-Protein-Interaction networks is a systems biological approach to this problem. Traditional Graph Clustering Methods are crisp, and allow only membership of each node in at most one cluster. However, most real world networks contain overlapping clusters. Recently the need for scalable, accurate and efficient overlapping graph clustering methods has been recognized and various soft (overlapping) graph clustering methods have been proposed. This paper introduces Chinese Whisper, for protein-protein interaction network analysis to discover overlapping functional modules. The paper illustrated the importance of soft clustering methods in systems biology by giving a few concrete examples of how the biological function of the overlap nodes relates to the functions of the respective clusters.*

Keywords: Protein-Protein Interaction networks; Graph Clustering; Overlapping functional modules; Coupling Matrix; Systems biology

1 Introduction

Homology based approaches have been the traditional bioinformatics approach to the problem of protein function identification. Variations of tools like BLAST [1] and Clustal [2] and concepts like COGs (Clusters of orthologous Groups) [3] have been applied to infer the function of a protein or the encoding gene from the known a closely related gene or protein in a closely related species. Although very useful, this approach has some serious limitations. For many proteins, no characterized homologs exist. Furthermore, form does not always determine function, and the closest hit returned by heuristic oriented sequence alignment tools is not always the closest relative or the best functional counterpart. Phenomena like Horizontal Gene Transfer complicate matters additionally. Last but not least, most biological Functions are achieved by collaboration of many different proteins and a proteins function is often context sensitive, depending on presence or absence of certain interaction partners.

A Systems Biology Approach to the problem aims at identifying functional modules (groups of closely cooperating and physically interacting cellular components that achieve a common biological function) or protein complexes by identifying network communities (groups of densely connected nodes in PPI networks). This involves clustering of PPI-networks as a main step. Once communities are detected, a hypergeometrical p-value is computed for each cluster and each biological function to evaluate the biological relevance of the clusters. Research on network clustering has focused for the most part on crisp clustering. However, many real world functional modules overlap. The present paper introduces a new simple soft clustering method for which the biological enrichment of the identified clusters seem to have in average somewhat better confidence values than current soft clustering methods.

2 Previous Work

Examples for crisp clustering methods include HCS [4], RNSC [5] and SPC [6]. More recently, soft or overlapping network clustering methods have evolved. The importance of soft clustering methods was first discussed in [7], the same group of authors also developed one of the first soft clustering algorithms for soft clustering, Clique Percolation Method or CPM [8]. An implementation of CPM , called CFinder [9] is available online. The CPM approach is basically based on the "defective cliques" idea and has received some much deserved attention. Another soft clustering tool is Chinese Whisper [10] with origins in Natural Language Processing. According to its author, Chinese Whispers can be seen as a special case of the Random Walks based method Markov-Chain-Clustering (MCL) [11] with an aggressive pruning strategy.

Recently, some authors [12, 13] have proposed and implemented betweenness based [14] Clustering (NG) method, which makes NG's divisive hierarchical approach capable of identifying overlapping clusters. NG's method finds communities by edge removal. The modifications involve node removal or node splitting. The decisions about which edges to remove and which nodes to split, are based on iterated all pair shortest path calculations.

In this paper, we apply Chinese Whispers for protein-protein interaction network analysis to discover overlapping functional modules. In the rest of the paper, we first describe Chinese Whispers. The second part of this work aims to illustrate the biological relevance of soft methods by giving several examples of how the biological functions of overlap nodes relate to biological functions of respective clusters.

3 Chinese Whispers

Chinese Whispers [10] is a randomized bottom-up Clustering algorithm with a time complexity of $O(|E|)$. In terms of complexity, the algorithm is quasi unbeatable. The Algorithm is outlined as (Figure 1):

```
initialize:
 forall vi in V: class(vi)=i;
while changes:
 forall v in V, randomized order:
 class(v)=highest ranked class
        in neighbourhood of v;
```

Figure 1. Pseudocode of Chinese Whispers

The algorithm is parameter free (there is no need to specify the number of clusters, a threshold, an external stopping condition etc.). There are however several configuration options that can strongly influence its behavior (a node changes its label in an update step differently, depending on chosen options).

The most important one is the choice of how the "highest ranked class" (fifth line in the description of the algorithm, Figure 1) in neighborhood of a vertex is determined.

To explain the difference between the possible choices we use the same example as Chinese Whispers User's manual [10] and paraphrase it where necessary:

Assume that we want to determine the highest ranked class in neighborhood of node A in Figure 2. Node A is currently labeled (i.e. assigned to community) L1, node B is labeled L4, C and E are assigned to community L3 and D is assigned to community L2. Furthermore link-strengths (weights) and degrees of the nodes are as shown in the figure.

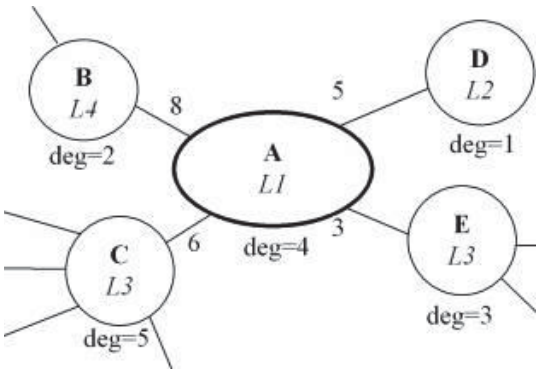


Figure 1: Calculation of Node Labels in Chinese Whisper

The strengths of classes for the situation in figure 1 are, dependent on the algorithm option:

top: strength(**L3**)=9; strength (L4)=8; strength (L2)=5

top sums over the neighbourhood's classes; there is an edge of weight 6 between A and C, and a Link of weight 3 between A and E. Both C and E have label L3, hence the total strength of L3 at A is 6+3=9. This is larger than the strengths of the two other classes (L2 and L4), so using this option would change A's Label to L3.

dist nolog: strength (**L2**)=5; strength (L4)=4; strength (L3)=2.2

dist downgrades the influence of a neighbouring node by its degree. For example, the total strength of L3 at A can be computed as: $\frac{6}{5}$ *(for node C)*$+\frac{3}{3}$ *(for node E)* =2.2. This is smaller than the influence of L2 at A $(\frac{5}{1} = 5)$. Using this option would change A's label to L2.

dist log: strength (**L4**)=7.28; strength (L3)=5.51; strength (L2)=3.46

The influence of neighbouring nodes is downgraded by their degree, but the penalty is less severe than in the previous case.

vote: strength (**L3**)=0.409; strength (L4)=0.363; strength (L2)=0.227

Setting the algorithm option to 'vote' gives essentially the same ranking as top, but expresses the strength of each label at a node as the fraction of their vote in the total vote. In other words, it divides each strength value by the sum of all strength values. Therefore strength of L3 at A, using the vote option is calculated as $\frac{9}{8+5+9}$ =0.409. When using vote as algorithm option, an additional vote threshold must be set. If the vote threshold is set to a value above 0.409, then A keeps its label L1.

As mentioned before, the algorithm option in ChineseWhispers is the most influential option. But the choices are limited and the algorithm is very fast, so in the worst case, there is the possibility to try out all options and consider only the best results. Furthermore, the decision is by far not as arbitrary as many other parameters that often surface in ML tasks. Using the knowledge from the last chapter, regarding the multi-functionality of highly-connected nodes, we can already speculate that the *dist nolog* Algorithm option will yield better results than the top or the vote option. This idea was confirmed in the analysis of the results on the yeast-PPI-Network.

Other configuration options include a random mutation rate that assigns new classes with a probability decreasing in

the number of iterations to avoid premature convergence in small graphs and to further decrease the influence of extraordinary well connected nodes (hubs). Lastly, there is a choice between continuous and stepwise update: in the continuous mode, a nodes label is changed immediately, so that it will participate in any calculation of its neighbors label with its new label. In the stepwise update mode, all class labels are updated at once, after all labels have been computed.

Biemann [10] explains how Chinese Whispers in stepwise mode can be interpreted as a tuned up version of a very popular graph clustering method, namely MCL.

The result of CW is a hard partitioning of the input graph into a number of partitions that emerges in the process – there is no need to specify the number of clusters in advance. The algorithm outputs the two highest ranked classes in the immediate neighborhood of each node. Therefore it is possible to obtain a *soft partitioning* based on the weighted distribution of (hard) classes in the neighborhood of a node in a final step.

Table 1. 38 Clusters with size >=10 identified by Chinese Whispers

Cluster Number	Cluster Size	GO Enriched ?
1	151	Yes
2	59	Yes
3	50	Yes
4	48	No
5	45	Yes
6	33	Yes
7	25	Yes
8	24	No
9	24	Yes
10	21	Yes
11	21	Yes
12	20	Yes
13	20	Yes
14	20	No
15	20	No
16	18	Yes
17	17	No
18	16	Yes
19	16	Yes
20	16	Yes
21	16	No
22	16	No
23	15	Yes
24	15	Yes
25	14	No
26	13	No
27	12	Yes
28	12	Yes
29	12	No
30	12	Yes
31	12	No
32	12	No
33	11	Yes
34	11	Yes
35	11	Yes
36	11	Yes
37	10	Yes
38	10	No

4 Experimental Results and Discussions

There are 38 clusters with more than 10 nodes. We were able to confirm a significant enrichment in Terms of Gene Ontology for 26 of these clusters. Table 1 summarizes the information about size and GO-significance of the clusters.

Table 2. Overlaps between Communities in Clustering Results

Cluster	Cluster	overlaps
1	119	2
1	153	1
19	73	1
19	83	2
19	85	1
19	119	1
19	392	2
22	43	2
22	73	2
22	83	3
22	129	1
22	137	5
22	869	1
43	60	1
43	73	1
43	83	1
43	85	2
43	364	5
60	492	1
65	83	1
65	143	1
65	869	1
73	83	1
73	85	1
73	137	1
73	226	3
83	137	1
83	196	3
83	392	2
83	870	2
85	153	2
85	236	11
119	153	2
119	364	2
129	226	2
137	170	1
143	153	1
143	170	2
150	364	1

In general, our Chinese Whispers cluster sizes are small. Also, the interfaces between clusters – where they exist- tend to be relatively sharp. The 26 clusters of size 10 and larger with significant GO-Enrichment "share" 79 nodes. These are nodes that after the final softening step have one of the clusters as primary and another one of the clusters as secondary class. Talbe 2 summarizes the overlaps between all of those clusters that were deemed biologically significant by GO-Enrichment analysis.

4.1 Enrichment Analysis

We preformed both GO-Enrichment and MIPS-functional catalogue Enrichment analysis for all clusters of size 10 and larger. Table 3 reviews the results, ordered by p-values. The table lists up to 6 different assignments for each of the clusters. The clusters listed are all clusters with a corr. p-value better than e-12.

4.2 GO Enrichment Analysis for Overlaps

Interestingly, the two clusters with the highest number of common nodes, namely clusters 85 and 236, have exactly the same GO-ID assigned to them. Here are two more examples of how the enrichment values of overlaps fit into enrichment values of the clusters.

Table 3. Best CW Communities - GO Enrichment

Cluster ID	GO_ID	p_val	cor_pval	hits	Network total	**Description**
distlognm85	377	5.6052E-45	4.8905E-43	31	96	RNA splicing, via transesterification reactions with bulged adenosine as nucleophile
distlognm85	398	4.2039E-45	4.8905E-43	31	95	Nuclear
distlognm85	375	7.567E-44	4.2908E-42	31	103	RNA splicing, via transesterification reactions
distlognm83	30163	5.814E-42	1.7498E-39	36	172	Protein
distlognm83	6508	1.4431E-41	2.1717E-39	36	176	Proteolysis
distlognm85	6395	7.5524E-41	3.2098E-39	31	125	RNA splicing
distlognm83	6511	4.8044E-41	3.6152E-39	34	146	ubiquitin-dependent protein catabolic process
distlognm83	19941	4.8044E-41	3.6152E-39	34	146	modification-dependent protein catabolic process
distlognm83	51603	1.0384E-40	6.2512E-39	34	149	proteolysis involved in cellular protein catabolic process
distlognm83	43632	2.8234E-40	1.4164E-38	34	153	modification-dependent macromolecule catabolic process
distlognm85	16071	3.2687E-39	1.1114E-37	34	201	mRNA metabolic process
distlognm85	6397	3.282E-38	9.299E-37	31	149	mRNA processing
distlognm43	42254	2.3371E-31	5.7961E-29	32	327	Ribosome
distlognm364	6365	1.1333E-30	7.5934E-29	20	169	rRNA
distlognm364	16072	2.6754E-30	8.9626E-29	20	176	rRNA metabolic process
distlognm143	7035	3.8047E-28	1.2429E-26	12	24	vacuolar acidification
distlognm143	45851	3.8047E-28	1.2429E-26	12	24	pH
distlognm143	51452	3.8047E-28	1.2429E-26	12	24	cellular pH reduction
distlognm143	51453	7.3133E-28	1.4334E-26	12	25	regulation of cellular pH
distlognm143	30641	7.3133E-28	1.4334E-26	12	25	cellular hydrogen ion homeostasis
distlognm43	22613	1.224E-28	1.5178E-26	32	396	ribonucleoprotein complex biogenesis and assembly
distlognm226	6383	8.4203E-28	2.1051E-26	12	38	Transcription
distlognm143	6885	7.2843E-27	1.1898E-25	12	29	regulation of pH
distlognm119	6402	2.6261E-27	3.6503E-25	14	59	mRNA
distlognm137	6350	1.1452E-26	1.5231E-24	25	546	Transcription
distlognm119	6401	3.0431E-26	2.115E-24	14	69	RNA catabolic process
distlognm137	32774	3.5555E-25	1.4628E-23	24	501	RNA biosynthetic process
distlognm137	6351	2.7814E-25	1.4628E-23	24	496	transcription, DNA-dependent
distlognm137	6366	4.3995E-25	1.4628E-23	22	333	transcription from RNA polymerase II promoter
distlognm364	42254	1.0632E-24	2.3744E-23	20	327	ribosome biogenesis and assembly
distlognm43	42273	3.259E-25	2.6941E-23	18	64	ribosomal large subunit biogenesis and assembly
distlognm60	31123	2.1759E-24	2.263E-22	12	39	RNA
distlognm364	22613	5.3701E-23	8.9949E-22	20	396	ribonucleoprotein complex biogenesis and assembly
distlognm60	31124	2.2489E-23	9.637E-22	11	29	mRNA 3'-end processing
distlognm60	6378	2.7799E-23	9.637E-22	10	18	mRNA polyadenylation
distlognm364	6394	8.0791E-23	1.0826E-21	20	404	RNA processing
distlognm236	398	9.9106E-23	4.4641E-21	14	95	Nuclear
distlognm236	377	1.1595E-22	4.4641E-21	14	96	RNA splicing, via transesterification reactions with bulged adenosine as nucleophile
distlognm236	375	3.3191E-22	8.5191E-21	14	103	RNA splicing, via transesterification reactions
distlognm60	43631	1.239E-21	3.2213E-20	10	24	RNA polyadenylation
distlognm236	6395	5.8246E-21	1.1212E-19	14	125	RNA splicing
distlognm236	6397	7.6012E-20	1.1706E-18	14	149	mRNA processing
distlognm153	6810	2.4763E-20	9.5833E-18	72	958	Transport
distlognm119	16071	2.3906E-19	1.1077E-17	14	201	mRNA metabolic process
distlognm153	51234	7.5479E-20	1.4605E-17	72	976	establishment of localization
distlognm60	6379	1.0696E-18	2.2249E-17	9	25	mRNA cleavage

distlognm153	51179	1.7797E-19	2.2958E-17	73	1017	Localization
distlognm137	16070	1.7109E-18	4.5509E-17	24	944	RNA metabolic process
distlognm236	16071	5.736E-18	7.3612E-17	14	201	mRNA metabolic process
distlognm22	6366	7.8764E-19	1.402E-16	21	333	Transcription
distlognm43	16072	3.5612E-18	2.2079E-16	19	176	rRNA metabolic process
distlognm22	6357	4.9211E-18	2.9199E-16	18	215	regulation of transcription from RNA polymerase II promoter
distlognm22	114	3.5293E-18	2.9199E-16	9	14	G1-specific transcription in mitotic cell cycle
distlognm119	43285	1.6413E-17	5.7037E-16	14	270	biopolymer catabolic process
distlognm119	44265	3.3728E-17	9.3763E-16	14	284	cellular macromolecule catabolic process
distlognm22	51318	3.7099E-17	1.3207E-15	10	26	G1 phase
distlognm22	80	3.7099E-17	1.3207E-15	10	26	G1 phase of mitotic cell cycle
distlognm60	6397	8.2237E-17	1.4254E-15	12	149	mRNA processing
distlognm43	6365	4.4492E-17	2.2068E-15	18	169	rRNA processing
distlognm119	9057	1.5382E-16	3.5635E-15	14	316	macromolecule catabolic process
distlognm137	6139	3.7892E-16	8.3994E-15	25	1419	nucleobase, nucleoside, nucleotide and nucleic acid metabolic process
distlognm43	42255	3.6745E-16	1.5188E-14	13	64	ribosome assembly
distlognm364	30490	2.1141E-15	1.8998E-14	9	38	maturation of SSU-rRNA
distlognm73	6366	1.4017E-16	2.439E-14	16	333	Transcription
distlognm22	6351	2.9808E-15	8.8432E-14	21	496	transcription, DNA-dependent
distlognm129	6338	8.1485E-15	1.1326E-12	11	149	Chromatin
distlognm226	32774	1.4217E-13	1.1848E-12	12	501	RNA biosynthetic process
distlognm226	6351	1.2588E-13	1.1848E-12	12	496	transcription, DNA-dependent
distlognm129	6323	3.9905E-14	1.3867E-12	12	247	DNA packaging
distlognm129	6366	2.606E-14	1.3867E-12	13	333	transcription from RNA polymerase II promoter
distlognm129	6325	3.9905E-14	1.3867E-12	12	247	establishment and/or maintenance of chromatin architecture
distlognm226	6350	4.0352E-13	2.522E-12	12	546	Transcription
distlognm73	6351	8.0707E-14	5.4854E-12	16	496	transcription, DNA-dependent
distlognm73	32774	9.4575E-14	5.4854E-12	16	501	RNA biosynthetic process
distlognm392	6454	4.8605E-14	5.9784E-12	8	46	Translational
distlognm129	6368	3.2547E-13	9.0481E-12	8	53	RNA elongation from RNA polymerase II promoter

Table 4. Two examples of how the enrichment values of overlaps fit into enrichment values of the clusters

Example 1: 137 and 22 share 5 nodes.

Distlognm137: (25 nodes)

GO-ID	p-value	corr p-value	# selected	# total	Description
6350	1.15E-26	1.52E-24	25	546	transcription
6351	2.78E-25	1.46E-23	24	496	transcription, DNA-dependent
32774	3.56E-25	1.46E-23	24	501	RNA biosynthetic process

Distlognm22:(33 nodes)

GO-ID	p-value	corr p-value	# selected	# total	Description
6366	7.88E-19	1.40E-16	21	333	transcription from RNA polymerase II promoter
114	3.53E-18	2.92E-16	9	14	G1-specific transcription in mitotic cell cycle
6357	4.92E-18	2.92E-16	18	215	regulation of transcription from RNA polymerase II promoter

Overlap of 137 and 22 (5 nodes)

GO-ID	p-value	corr p-value	# selected	# total	Description
6355	1.93E-04	4.40E-03	3	338	regulation of transcription, DNA-dependent
45449	2.41E-04	4.40E-03	3	364	regulation of transcription
122	3.10E-04	4.40E-03	2	60	negative regulation of transcription from RNA polymerase II promoter

Example 2: 43 and 364 share 5 nodes

Distlognm43(45 nodes, hereof 1 un-annotated):

GO-ID	p-value	corr p-value	# selected	# total	Description
42254	2.34E-31	5.80E-29	32	327	ribosome biogenesis and assembly
22613	1.22E-28	1.52E-26	32	396	ribonucleoprotein complex biogenesis and assembly
42273	3.26E-25	2.69E-23	18	64	ribosomal large subunit biogenesis and assembly

Distlognm364(21 nodes):

GO-ID	p-value	corr p-value	# selected	# total	Description
6365	1.13E-30	7.59E-29	20	169	rRNA processing
16072	2.68E-30	8.96E-29	20	176	rRNA metabolic process
42254	1.06E-24	2.37E-23	20	327	ribosome biogenesis and assembly

Overlap of 43 and 364(5 nodes):

GO-ID	p-value	corr p-value	# selected	# total	Description
42254	5.37E-07	1.77E-05	5	327	ribosome biogenesis and assembly
22613	1.41E-06	2.32E-05	5	396	ribonucleoprotein complex biogenesis and assembly
6365	3.32E-06	3.22E-05	4	169	rRNA processing

5 Conclusions

This paper introduced Chinese Whispers [10], a randomized bottom-up Clustering algorithm with a time complexity of $O(|E|)$, for protein-protein interaction network analysis to discover overlapping functional modules. In this paper, we first described Chinese Whispers. We further illustrated the biological relevance of soft methods by giving several examples of how the biological functions of overlap nodes relate to biological functions of respective clusters. The paper illustrated the importance of soft clustering methods in systems biology by giving a few concrete examples of how the biological function of the overlap nodes relates to the functions of the respective clusters.

6 References

[1] Altschul, SF, et al. "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs". Nucleic acids research 25, no. 17: 3389, 1997.

[2] Thompson, JD, DG Higgins, and TJ Gibson. "CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice". Nucleic acids research 22, no. 22: 4673-4680, 1994

[3] Tatusov, R. L., E. V. Koonin, and D. J. Lipman. "A genomic perspective on protein families". Science 278, no. 5338: 631, 1997.

[4] Hartuv, E., R. Shamir. "A clustering algorithm based on graph connectivity". Information processing letters 76, no. 4-6: 175-181, 2000.

[5] King, A. D., N. Przulj, and I. Jurisica. "Protein complex prediction via cost-based clustering". Bioinformatics 20,: 3013-3020, 2004.

[6] Spirin, V., L. A. Mirny. "Protein complexes and functional modules in molecular networks". Proceedings of the National Academy of Sciences 100, no. 21: 12123-12128, 2003.

[7] Palla, G., I. Derenyi, I. Farkas, and T. Vicsek. "Uncovering the overlapping community structure of complex networks in nature and society". Nature 435, no. 7043 (Jun 9): 814-818, 2005.

[8] Derenyi, I., et al. "Clique percolation in random networks". Physical Review Letters 94, no. 16: 160202, 2005.

[9] Adamcsek, B., G. et al. "CFinder: locating cliques and overlapping modules in biological networks". Bioinformatics 22, no. 8: 1021-1023, 2006.

[10] Biemann, C. "Chinese whispers-an efficient graph clustering algorithm and its application to natural language processing problems". In Proceedings of the HLT-NAACL-06 workshop on textgraphs-06, new york, USA, 2006.

[11] Van Dongen, S. "A cluster algorithm for graphs". Report- Information systems , no. 10: 1-40, 2000.

[12] Pinney, J. W., D. R. Westhead. "Betweenness-based decomposition methods for social and biological networks". In Interdisciplinary statistics and bioinformatics. Edited by S. Barber, P. D. Baxter, K. V. Mardia and R. E. Walls. Leeds University Press, 2000.

[13] Gregory, S. "An algorithm to find overlapping community structure in networks". Lecture Notes in Computer Science 4702: 91, 2007.

[14] Girvan, M., M. E. Newman. "Community structure in social and biological networks". PNAS 99: 7821-7826, 2002.

Profrager Web Server: Fragment Libraries Generation for Protein Structure Prediction

Karina B. Santos[1], **Raphael Trevizani**[2], **Fábio L. Custódio**[1*] **and Laurent E. Dardenne**[1]
[1]Dept. of Comp. Mechanics - National Laboratory for Scientific Computing (LNCC), Petrópolis, RJ, Brazil
[2]Oswaldo Cruz Foundation (Fiocruz), Fortaleza, CE, Brazil
Email: karinabs@lncc.br, raphael.trevizani@fiocruz.br, *flc@lncc.br, dardenne@lncc.br
*Corresponding author

Abstract— *This paper describes Profrager, a new flexible web server for generation of protein fragment libraries. These libraries have widespread use amongst modern protein structure prediction methods. Profrager offers several options for generating customized libraries, e.g., the users can choose between three options of structural databases to generate the libraries and can also define the number of fragments per position and the fragments lengths. The selection of fragments can be guided by three scoring strategies: (i) use only sequence similarity to the target sequence; (ii) use a weighted sum of the sequence similarity score and a secondary structure score; (iii) use a Pareto Efficiency strategy with the two scores. The software outputs useful statistics about the fragments in addition to files fully compatible with the GAPF and Rosetta protein structure prediction programs. Profrager is available at* `http://www.lncc.br/sinapad/Profrager/` *as a web service.*

Keywords: protein fragment libraries; protein structure prediction

1. Introduction

The prediction of protein structures is a central challenge of modern computational biology [1]. The use of fragment libraries is one of the basic strategies employed by several successful protein structure prediction (PSP) methods [2], [3]. The objective is to simplify the complexity of PSP by reducing the conformational search space [4]. Fragment libraries are assembled from a database of experimentally determined structures and are specific to each target protein sequence. These libraries can be understood as a selected collection of possible fragments which are used to construct segments of a target sequence. Information contained within the fragments is used to build the whole tridimensional structure of the target protein [5]. Commonly, libraries are constructed by similarity between the amino acids sequences of the fragments and the target protein [6]. However, other criteria may be used, e.g., the agreement between the observed fragments secondary structure and the predicted secondary structure of the target [7].

Robust fragment libraries should allow the reconstruction of the correct protein folding using only the fragments from non-homologous structures [8]. Therefore, programs for fragment libraries generation should present several options to guide the choice of fragments in order to improve the prediction capacity of PSP methods, e.g., the amino acid substitution matrix used to select the fragments and the database of experimental structures from which to extract the fragments.

A web server provides a user friendly interface to generate the libraries without the need to install any programs, and avoids the lengthy process of creating a geometry database that are used for the fragment construction. An example of a web server, which enables users to create fragment libraries, is the Robetta server [9] (`http://robetta.bakerlab.org/`). The Rosetta method uses, as one the initial steps in its PSP protocol, the generation of fragment libraries for a specific target sequence [10]. Libraries generated by Robetta are specifically formatted to be used with the Rosetta PSP software.

The objective of this work was the development of a flexible program for creating customized PSP fragment libraries, using different databases, amino acids substitution matrices and scoring criteria for fragment selection. This program is available to the scientific community in the form of an interactive web server called Profrager (`http://www.lncc.br/sinapad/Profrager/`).

2. Implementation

Profrager creates fragment libraries from a selected database of known protein structures. This database is a subset of the Protein Data Bank (PDB) [11] extracted using PISCES (Protein Sequence Culling Server) [12]. At present, the user can choose from two different PISCES databases: (i) one comprising 5387 sequence entries, with no more than 20% identity between the sequences and resolution up to 2.0 angstroms, or (ii) one with 17342 sequence entries, with no more than 50% identity between the sequences and resolution up to 2.5 angstroms. These two databases have structures elucidated by X-ray crystallography (R-factor bellow 0.3) and NMR. Additionally, a third database option is available for the users, the Rosetta's Vall database, with 16800 sequence entries.

Profrager is capable of generating libraries with fragments of any length. Furthermore, in addition to the fragment length the user can also define the number of fragments per position. A "position" refers to the residue in the target sequence where the fragment starts. Thus, a fragment of three residues from the first position contains the structural information of residues 1, 2 and 3. The fragments overlap in consecutive positions, e.g., the second position appertain to residues 2, 3 and 4, the third 3, 4 and 5 and so forth until the end of the target sequence is reached. The default options are 200 fragments per position and libraries with three and nine residues length.

2.1 Profrager in Use

From a target sequence the program scans the chosen database building a list of candidate fragments for each position. The choice of which fragments will be included in the final library is guided by a ranking score. Each candidate fragment has its sequence similarity, to the corresponding segment on the target sequence, evaluated using an amino acids substitution matrix. The user may choose to use BLOSUM62 (default), BLOSUM45, PAM30 or PAM80 matrices. Sequence similarity identifies the probability of an amino acid being replaced by another in the protein sequence. The sequence similarity score is given by the sum of values from the matrix comparing the fragment sequence with the target segment sequence.

The selection of fragments can be augmented by comparing the predicted secondary structure for the target sequence, using PSIPRED [13] (or other program by providing a secondary structure file in the horizontal format), and the secondary structure for the proteins in the database detected using STRIDE [14]. The score for this comparison is calculated using the confidence given by PSIPRED for each residue. When the predicted secondary structure for a position on the target sequence is the same as the detected in the corresponding position in a fragment, the confidence score is added to the score of that fragment. Otherwise, the confidence is subtracted from the score. The secondary structure score is added to the similarity score and the final score is used for fragment ranking. Moreover, an important customization aspect is that the secondary structure score can be multiplied by a weight defined by the user (1.0 by default).

Another fragment selection option implemented in Profrager, is the use of a multi-objective Pareto Efficiency strategy [15]. This strategy avoids the choice of a particular value to weight the two scores (amino acid and secondary structure similarities). Pareto Efficiency employs the concept of dominance where fragments which have the best scores for at least one criterion are classified as non-dominated and make up the Pareto Front. Successive fronts are used to build the fragment libraries until the desired number of fragments per position is fulfilled. In general, these are the fragments that have the best values for at least one evaluation criteria.

Users have access to other advanced options during the creation of their libraries. The minimum score a fragment need to obtain to be included in the library can be controlled. Furthermore, fragments might be extracted from: (i) any protein in the database, (ii) only non-homologous proteins to the target sequence or (iii) exclusively from homologous proteins to the target sequence. In these two last cases, homology is detected using PSI-BLAST [16] via the E-Value.

2.2 Output

The default output format is compatible with the GAPF PSP suite developed in our group (http://www.gmmsb.lncc.br) [17], [18], [19]. The fragment libraries files contain, for each residue at each position, the following information in separate columns: (1) PDB code and chain of the structure which originated the fragment, (2) type of amino acid (one letter code), (3) type of secondary structure, (4) position in the target sequence, (5) position in the sequence from the PDB, (6) backbone dihedral angles ϕ, ψ and ω, (7) main chain bond angles defined by N-Cα-C, Cα-C-N and C-N-Cα, (8) the score from sequence similarity, (9) the score from secondary structure agreement and (10) the total score. Each file is a fragment library of a particular length and all target sequence positions are marked with a header line containing the position number and the number of fragments at that position. Moreover, allowing for a wide range of applications for the libraries generated by Profrager, files in a format compatible with Rosetta are created by default. Another useful output is an automatically generated plot depicting the fragments' secondary structure distribution, per position, for each library created.

It has been shown that when using backbone angles from experimental structures with idealized (and fixed) bond geometries, e.g., in *de novo* and *ab initio* PSP, the resulting structures can present large deviations from the original structure, for longer sequences this is more severe [8]. This can be solved by freezing bonds under idealized geometries and then optimizing the backbone dihedral angles to recreate structures as close as possible to the originals. Alternatively, the libraries can include the backbone geometry bond angles. The first option has the disadvantage of requiring a preliminary processing of all structures contained in the database, in addition to changing the actual values of the original backbone angles. Profrager generated libraries have backbone torsion angles values extracted directly from experimental structures. A different choice can be found in libraries generated by Robetta, which have backbone dihedral angles calculated from structures with idealized bond geometries [20]. Nevertheless, Profrager users' have the option of using Rosetta's geometries database to generate the fragment libraries. In this case, Profrager libraries will

contain recalculated dihedral angles and fixed idealized main chain bond angles.

3. Methods

For the validation of the libraries and to demonstrate their compatibility with the Rosetta suite, a set of 48 proteins ranging from 54 to 148 residues was selected from the CASP9 experiment (Table 1). For each target, three differents fragment libraries were generated by Profrager with the Rosetta's Vall database using: (I) only sequence similarity, given by Blosum62, (II) sum of the similarity score and the secondary structure score (weight=1.0) and (III) Pareto Efficiency strategy. Each generated library has 200 fragments with three residues (3-mers) and nine residues (9-mers) for each position of the target sequence. For each type of generated fragment libraries we perform a protein structure prediction protocol using Rosetta (version 3.4). The default *ab initio*-relax protocol was used and 1000 models for each sequence were generated. The quality of generated structures was evaluated with the TM-Score program [21]. This program gives the GDT-TS criterion (Global Distance Test Total Score) and only the best model (i.e., with greater GDT-TS value) was considered during comparisons. Models with GDT-TS $\geq 50\%$ indicate good predictive ability [22]. For the sake of comparison, all tests were also performed against fragment libraries generated by the Robetta server (using secondary structure prediction and sequence similarities scoring schemes), which are the default libraries for Rosetta predictions.

4. Results and Discussion

Table 1 shows the GDT-TS values of the best models generated using fragment libraries from Robetta server and using fragment libraries from Profrager server - Library I (based on sequence similarity alone), Library II (based on secondary structure prediction and sequence similarity) and Library III (based on Pareto Efficiency).

Library I showed the worst results and for the majority of sequences the best models generated using the Robetta libraries have GDT-TS values similar to those generated using Profrager libraries II and III.

The number of sequences that had models with good quality generated, GDT-TS > 50%, was: Library I: 3, Library II: 9, Library III: 9 and Robetta Library: 11. Thus, the models created by Rosetta using libraries generated by the Robetta server have a small advantage over those created using libraries generated by Profrager. By comparing the distribution of secondary structures between different libraries (Fig. 1) it becomes apparent that there are considerable differences between those from Profrager libraries, which uses PSIPRED, and those from Robetta libraries. For example, Profrager provides mainly coil fragments between residues 45 and 55, while Robetta provides mainly helical fragments.

Table 1: GDT-TS score (%) of the best models.

CASP9 ID	PDB	Library I	Library II	Library III	Robetta*	Length
T0522	3nrd	28.85	41.54	39.42	42.31	134
T0523	3mqo	27.48	40.77	35.36	42.79	120
T0527	3mr0	24.61	29.33	28.35	32.09	142
T0530	3npp	33.43	44.77	47.97	58.43	115
T0531	2kjx	33.85	41.54	40.00	43.46	65
T0538	2l09	60.08	84.27	79.44	82.66	54
T0539	2l0b	22.25	21.70	21.7	20.60	81
T0540	3mx7	46.67	57.50	54.44	57.50	90
T0541	2l0d	28.95	40.13	32.46	36.40	106
T0544	2l3w	33.74	35.66	30.07	46.50	135
T0546	2l5q	27.29	32.39	30.46	29.93	134
T0548	3nnq	36.96	51.09	45.92	45.38	106
T0549	2kzv	40.49	50.00	46.74	55.16	84
T0551	3obh	28.13	28.52	28.52	31.25	74
T0552	2l3b	29.81	33.85	36.73	41.73	122
T0553	2ky4	28.69	40.10	31.71	40.60	141
T0555	2l06	28.23	27.42	25.16	34.19	148
T0557	2kyy	26.31	32.03	32.19	34.15	145
T0559	2l01	47.40	66.88	64.94	77.92	69
T0560	2l02	49.70	61.28	61.59	68.60	74
T0562	2kzx	32.44	37.98	34.35	39.89	123
T0564	2l0c	26.80	41.49	40.98	41.75	89
T0567	3n70	21.07	24.82	24.82	26.07	145
T0569	2kyw	37.93	38.51	42.53	38.79	79
T0572	2kxy	25.00	26.50	26.75	29.00	93
T0574	3nrf	26.49	32.18	35.64	41.83	126
T0579	2ky9	19.32	21.78	20.83	22.92	124
T0580	3nbm	35.89	47.28	45.54	47.52	105
T0581	3npd	40.77	38.96	34.91	43.69	136
T0586	3neu	28.91	36.52	36.09	33.48	125
T0590	2kzw	24.14	18.28	18.62	26.90	137
T0592	3nhv	19.70	19.89	19.13	17.99	144
T0594	3ni8	22.68	22.50	21.25	20.54	140
T0600	3nja	25.48	25.48	25.00	26.92	125
T0602	3nkz	41.40	47.04	50.27	50.27	123
T0605	3nmd	53.85	62.50	60.58	62.50	72
T0612	3o0l	33.41	36.45	36.92	40.89	129
T0614	3voq	24.35	23.71	22.20	24.57	135
T0616	3nrt	32.78	39.17	38.89	39.72	103
T0617	3nrv	27.38	34.13	36.11	37.50	148
T0619	3nrw	36.52	29.17	31.13	34.31	111
T0622	3nkl	40.21	60.83	51.25	57.71	138
T0624	3nrl	41.04	51.49	44.78	55.97	81
T0630	2kyt	26.40	31.20	30.80	37.40	132
T0634	3n53	25.00	34.48	39.44	42.46	140
T0637	2x3o	22.33	26.53	25.19	23.47	146
T0639	3nym	28.54	32.08	36.50	34.96	128
T0643	3nzl	51.43	58.21	63.93	63.21	83
Average		32.38	38.75	37.66	41.33	

Library I: built using only sequence similarity. Library II: built using sequence similarity score and secondary structure prediction agreement. Library III: built using a Pareto Efficiency Strategy. Robetta: libraries built using Robetta server.

This is because the Robetta server uses a weighted average of various secondary structure prediction methods [23]: PSIPRED, JUFO [24], SAM [25] and PROF [26]. In general, the use of a consensus strategy improves the accuracy of the prediction [27].

It is interesting to note that the use of secondary structure prediction information greatly reduces fragment diversity within the libraries to the point of some positions having

only one type of secondary structure represented (Fig. 1). This is a desirable trait as it reduces the search space for PSP algorithms, allowing that better models be generated. However, good predictions become extremely dependent of accurate secondary structure predictions.

5. Conclusions

Profrager web server for protein fragment libraries generation presents a wide range of flexible and useful options, not present in other similar services. Beyond the basic options of number of fragments per position and fragment length, the web server allows the creation of libraries from different pre-built databases (20% or 50% identity cut-off) or even Rosetta's Vall database. The different strategies for fragment selection offered by Profrager can be used to generate distinct fragment libraries for PSP programs. Moreover, the facilities provided by Profrager can be interesting to enable the development of new PSP strategies by other research groups in this field.

To further improve the quality of the libraries, an option to use several secondary structure prediction methods is being implemented in Profrager.

Acknowledgment

Funding: This work was supported by CNPq - Contract grant n°. 307062/2010-4 and FAPERJ - Contract grant n°. E26/102.443/2009. SINAPAD (`https://www.lncc.br/sinapad/`) team for hosting and web server and Eduardo Krempser for drafting the first versions of the server.

References

[1] D. Baker and A. Sali, "Protein structure prediction and structural genomics," *Science*, vol. 294, no. 5540, pp. 93–96, 2001.

[2] S. Raman, R. Vernon, J. Thompson, M. Tyka, R. Sadreyev, J. Pei, D. Kim, E. Kellogg, F. DiMaio, O. Lange, *et al.*, "Structure prediction for casp8 with all-atom refinement using rosetta," *Proteins: Structure, Function, and Bioinformatics*, vol. 77, no. S9, pp. 89–99, 2009.

[3] Y. Zhang, "I-tasser: Fully automated protein structure prediction in casp8," *Proteins: Structure, Function, and Bioinformatics*, vol. 77, no. S9, pp. 100–113, 2009.

[4] R. Kolodny, P. Koehl, L. Guibas, and M. Levitt, "Small libraries of protein fragments model native protein structures accurately." *Journal of Molecular Biology*, vol. 323, no. 2, pp. 297–307, 2002.

[5] R. Bonneau, J. Tsai, I. Ruczinski, D. Chivian, C. Rohl, C. Strauss, and D. Baker, "Rosetta in casp4: progress in ab initio protein structure prediction," *Proteins: Structure, Function, and Bioinformatics*, vol. 45, no. S5, pp. 119–126, 2001.

[6] S. Li, D. Bu, X. Gao, J. Xu, and M. Li, "Designing succinct structural alphabets," *Bioinformatics*, vol. 24, no. 13, pp. i182–i189, 2008.

[7] D. Chivian, D. Kim, L. Malmström, J. Schonbrun, C. Rohl, and D. Baker, "Prediction of casp6 structures using automated robetta protocols," *Proteins: Structure, Function, and Bioinformatics*, vol. 61, no. S7, pp. 157–166, 2005.

[8] J. Holmes and J. Tsai, "Some fundamental aspects of building protein structures from fragment libraries," *Protein science*, vol. 13, no. 6, pp. 1636–1650, 2004.

[9] D. Kim, D. Chivian, and D. Baker, "Protein structure prediction and analysis using the robetta server," *Nucleic Acids Research*, vol. 32, no. suppl 2, pp. W526–W531, 2004.

[10] P. Bradley, D. Chivian, J. Meiler, K. Misura, C. Rohl, W. Schief, W. Wedemeyer, O. Schueler-Furman, P. Murphy, J. Schonbrun, *et al.*, "Rosetta predictions in casp5: successes, failures, and prospects for complete automation," *Proteins: Structure, Function, and Bioinformatics*, vol. 53, no. S6, pp. 457–468, 2003.

[11] H. Berman, K. Henrick, H. Nakamura, and J. Markley, "The worldwide protein data bank (wwpdb): ensuring a single, uniform archive of pdb data," *Nucleic acids research*, vol. 35, no. suppl 1, pp. D301–D303, 2007.

[12] G. Wang and R. Dunbrack Jr, "Pisces: a protein sequence culling server," *Bioinformatics*, vol. 19, no. 12, pp. 1589–1591, 2003.

[13] L. McGuffin, K. Bryson, and D. Jones, "The psipred protein structure prediction server," *Bioinformatics*, vol. 16, no. 4, pp. 404–405, 2000.

[14] D. Frishman and P. Argos, "Knowledge-based protein secondary structure assignment," *Proteins: Structure, Function, and Bioinformatics*, vol. 23, no. 4, pp. 566–579, 1995.

[15] A. Charnes, W. Cooper, B. Golany, L. Seiford, and J. Stutz, "Foundations of data envelopment analysis for pareto-koopmans efficient empirical production functions," *Journal of Econometrics*, vol. 30, no. 1, pp. 91–107, 1985.

[16] S. Altschul, T. Madden, A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. Lipman, "Gapped blast and psi-blast: a new generation of protein database search programs," *Nucleic acids research*, vol. 25, no. 17, pp. 3389–3402, 1997.

[17] F. L. Custódio, H. J. Barbosa, and L. E. Dardenne, "Genetic algorithm for finding multiple low energy conformations of poly alanine sequences under an atomistic protein model," *Advances in Bioinformatics and Computational Biology*, pp. 163–166, 2007.

[18] ——, "Full-atom ab initio protein structure prediction with a genetic algorithm using a similarity-based surrogate model," in *Evolutionary Computation (CEC), 2010 IEEE Congress on*. IEEE, 2010, pp. 1–8.

[19] ——, "A multiple minima genetic algorithm for protein structure prediction," *Applied Soft Computing*, vol. 15, pp. 88–99, 2014.

[20] C. A. Rohl, C. E. Strauss, K. M. Misura, and D. Baker, "Protein structure prediction using Rosetta," *Methods Enzymol.*, vol. 383, pp. 66–93, 2004.

[21] Y. Zhang and J. Skolnick, "Scoring function for automated assessment of protein structure template quality," *Proteins: Structure, Function, and Bioinformatics*, vol. 68, no. 4, pp. 1020–1020, 2007.

[22] J. Xu and Y. Zhang, "How significant is a protein structure similarity with tm-score= 0.5?" *Bioinformatics*, vol. 26, no. 7, pp. 889–895, 2010.

[23] R. Das, B. Qian, S. Raman, R. Vernon, J. Thompson, P. Bradley, S. Khare, M. D. Tyka, D. Bhat, D. Chivian, *et al.*, "Structure prediction for casp7 targets using extensive all-atom refinement with rosetta@ home," *Proteins: Structure, Function, and Bioinformatics*, vol. 69, no. S8, pp. 118–128, 2007.

[24] J. Meiler, M. Müller, A. Zeidler, and F. Schmäschke, "Generation and evaluation of dimension-reduced amino acid parameter representations by artificial neural networks," *Molecular modeling annual*, vol. 7, no. 9, pp. 360–369, 2001.

[25] K. Karplus and B. Hu, "Evaluation of protein multiple alignments by sam-t99 using the balibase multiple alignment test set," *Bioinformatics*, vol. 17, no. 8, pp. 713–720, 2001.

[26] M. Ouali and R. D. King, "Cascaded multiple classifiers for secondary structure prediction," *Protein Science*, vol. 9, no. 06, pp. 1162–1176, 2000.

[27] Y. Wei, J. Thompson, and C. Floudas, "Concord: a consensus method for protein secondary structure prediction via mixed integer linear optimization," in *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*. The Royal Society, 2011, p. rspa20110514.

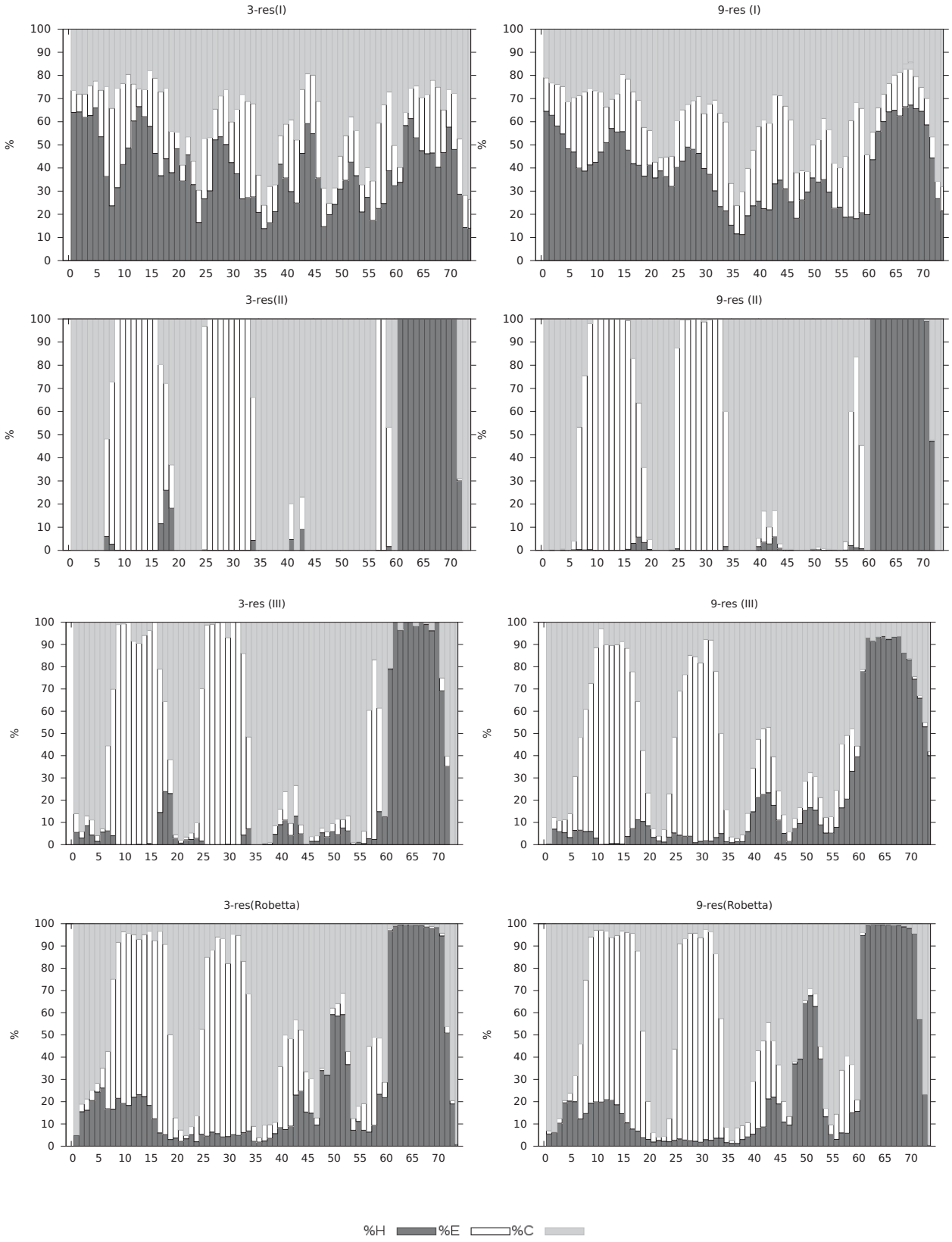


Fig. 1: Secondary structure distribution profile for generated libraries with fragments containing 3 and 9 residues, target T0551. The horizontal axis represent each residue in the sequence. Vertical axes show secondary structure percentages among the fragments at each residue (position): H = helix, C = coil and E = extended. (I) Libraries built using only sequence similarity. (II) Libraries built using sequence similarity score and secondary structure prediction agreement score. (III) Libraries built using Pareto Efficiency strategy. (Robetta) Libraries built using Robetta server. Graphics I, II and III are automatically generated by Profrager.

Ant Colony Optimization for Construction of Common Pattern of the Protein Motifs

J. Altamiranda[1], **J. Aguilar**[1,3], **and C. Delamarche**[2]
[1]Computer Department, University of Los Andes, Mérida, Venezuela, {altamira, aguilar}@ula.ve
[2]Structure et Dynamique des macromolecules, University of Rennes I, Rennes, France,
christian.delamarche@univ-rennes1.fr
[3]Prometeo Researcher, Universidad Técnica Particular de Loja, Ecuador

Abstract - *In this work is presented an approach for the construction of common patterns of the protein motifs of the amyloid protein motifs, extracted from the database AMYPdb, denoted as regular expressions using the rules PROSITE. Our task is to analyze a set of possible motifs and to detect if similarity exists between them, in order to construct a general motif. The Ant Colony Optimization Model uses an algorithm of combinatorial optimization based on Ant Colonies. It uses the amino acids of the first motif to construct the graph where the ants will walk. Then, the graph is crossed by the ants according to the path of the second motif, used by a transition function that promove to flow the path between similars amino acids. The ants when walking leave pheromone in the nodes, in a way that at the end several have a lot of or little pheromone. Finally the graph is crossed again to construct the resultant regular expression composed by the nodes with much pheromone.*

Keywords: Bioinformatics, Ant Colony Optimization, Proteins, Biology Computing, Biological Process

1 Introduction

This paper defines and develops a computational model for the construction common patterns of protein motifs. It proposes an algorithm based on ACO [1], with some modifications. This algorithm can efficiently find the union between two motifs and allows the generation of a new motif.

The two important Bioinformatics tools BLAST [2], FASTA [3] have been developed as a response to the needs of new knowledge about the sequences and protein motifs, using the information stored in these databases. For perform multiple alignment of protein sequences CLUSTAL which is software that provides comprehensive multiple alignment using progressive strategies for aligning DNA and protein sequences of multiple species and helps to find common conserved domains [4]. But there are still problems to solve at the level of information discovery, data classification, among others.

Currently, there are several methods of patterns discovery (using Regular Expressions [5], [6], Hidden Markov Model (HMM) [7], Automata, and PSSM Matrix). The regular expressions are the most commonly used by biologists, as well as the graphical method of LOGOS, since visually are simpler to understand and interpret for them [8], [9].

To discover of DNA motif historically has been used the Pratt method [10], which is based on the algorithm Knuth-Morris-Pratt [11], but there are other tools, between the most well-known it have [12], [13], [14], [15], [16], [17]: TEIRESIAS, MEME.

The discovery of common motifs between sequences that are distant in evolutionary level (non-homologous or non-related sequences) is a very complex problem. In addition, there are tools that allow comparing DNA motifs and SLM (Short Linear Motifs) defined as regular expressions, such as CompariMotif [18], FunClust [19], and Bio.motif [20]. However, these tools do not allow fusing them into a common expression.

It's possible to discover relationships between proteins construction a common pattern between multiple motifs. The relationships can be illustrated by looking at the following example: If three motifs If S1 (C-x-H-x-[LIVMFY]-C-x (2)-C-[LIVMYA]), S2 (H-A-M-C-x-(2)-C) and S3 (H-x-L-C-{R}-C). It is observed that S2 and S3 are sub-motifs of S1. May be written a common motif would be (H-x-[ML]-C-x (2)). Assuming that S1, S2, S3 are specific motif of 3 different families and that common motif does not match any sequence families 1, 2 and 3, represents a consensus motif. As this example, other biological analyzes for groups of motives could do. For this will be need to define a method generating a common pattern for them, and then test the quality.

Specifically, our task is to analyze a set of Protein motifs stored in a database, detect if there are similarities between them, and construct general patterns. The patterns found can be explained by the existence of segments that have been preserved during the natural evolution of proteins, and suggest that the obtained regions play a functional role in their mechanisms and structure. Most of the algorithms of motifs search use heuristic techniques to obtain near optimal solutions with a relatively low computational cost [9]. For

example, some works based on bio-inspired algorithms are: [21], [22], these works are applied only to DNA sequences that contain four nucleotides. In our case the method is used in protein motifs that contain twenty amino acids and are represented for regular expressions.

This paper propose to define and to develop a computational method for the construction common patterns of protein motifs denoted in PROSITE rules composed of 20 amino acid and X that represent a gap.

2 Problem

One of the most important fields in Bioinformatics is related to the identification of motifs in a protein. The problems that emerge are interesting because they face the use of mathematical and computer methods to build motifs. In addition, usually the execution times of the algorithms for a result are highly algorithmic complexity.

To understand the utility of the construction common patterns of protein motifs is necessary to understand how proteins evolve, and to know that there are similarities between them. When a protein differs from another of the same kind , from that time, two versions of the same protein is established , they begin to evolve randomly occurring changes in each version , the structure slowly begins to change independently in both, while retaining certain identical regions. Another form of evolution occurs when two proteins have a homologous structure, without having a common ancestor; it is known as convergent evolution. These regions of proteins exist because they are necessary to maintain their biological properties. These small regions suffer strong restrictions structural in the evolution, so they can be recognized by analyzing motifs. Analysis of these changes to infer the origin of certain motifs, or discover new motifs in proteins.

In this way it can build a common pattern to these motifs,

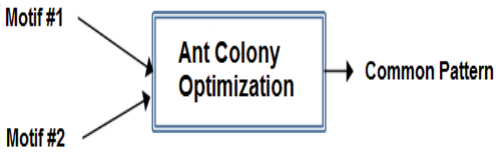


Fig 1. Model Propose

representing these small conserved regions, and get a list of motifs which are conserved among protein families.

3 Solution

It proposes a model for construction common patterns of Protein Motifs denoted as regular expression using the PROSITE rules. This model can efficiently find the union between two regular expressions, and allows the generation of a new regular expression.

It used the Ant Colony Optimization to construct a graph with the amino acids of the first motif. Then, the graph is crossed by the ants according to the path of the second motif. Finally the graph is crossed again to construct the resultant regular expression. In each execution of our algorithm, two motifs are fused (Fig. 1). In general, the macro-algorithm for the construction common pattern process is:

- Create the route graph.
- Walk of the ants on the route graph.
- Choose the best nodes
- Construct the resultant motif

3.1 Create the route graph

The problem of motif common pattern emerges from the study of the primary structure of proteins, there are two basic conditions for the design of the graph where will walk the ants:

- From an analysis in the construction of motifs, which shows that is essential for this task the position of different nucleotides along of the chains that can be viewed as one dimensional array.
- The product of the construction common patterns must generate a new motif that contains the nucleotides chains that belong to motifs fused.

For the previous reasons, our graph will be represented in the plane, and each node will have arcs at the right and left sides, in this way the ants can only move them in horizontal direction. The nodes must store the pheromone level deposited by the ants that visit them and the nucleotide that represent (Fig. 2). This information will be constituted by the type of amino acid that represents, and the family to which it belongs

TABLE I
CLASSIFICATION AND FAMILY OF THE AMINO ACIDS

Amino Acids Family	Amino Acids	Classification
Aliphatic Amino Acids	G A V I L M	1
Aromatic Amino Acids	F Y W	2
Basic Amino Acids	K R H	3
Neutral Amino Acids	S T N Q	4
Acid Amino Acids	D E	5
Sulfur Amino Acids	C	6
Imino Acid	P	7

F

(Table I), or an identifier for special nodes (Table II).

TABLE II
IDENTIFIER FOR SPECIAL NODES

Information	Special Identifier	Classification
Gap	G A V I L M	0
Empty	F Y W	-1
Start	K R H	-2
End	S T N Q	-2

For the graph construction it transform the first motif in a stack data structure (for example, see the motif S-A(1,3)-x-[KV]. (Figure 3).

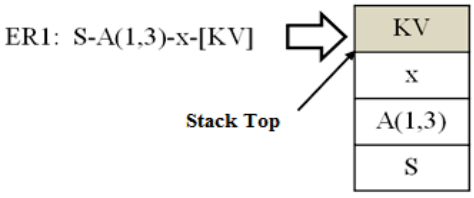


Fig. 3. Transformation of a motif 1 in a stack

Additionally, two nodes are defined, that serve as guide for the construction of the graph, to indicate the start and end of the route (Fig. 4). Then, it proceed to extract the elements that are at the top of the stack iteratively, and built the nodes in the graph (amino acids) which are in the same position in the chain. Also adds a node gap, which will serve as an auxiliary route for cases in which the ants must not continue for any of the available nodes. In this way, it avoid that an ant stops itself.

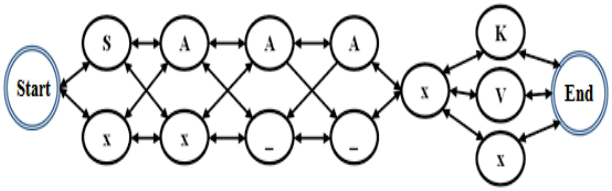


Fig. 4. Route graph of the motif 1

For the special case when there are 2 values within the parentheses, it defines a special node, called empty "_", to avoid the deadlock. It is necessary because the arcs that lead to these nodes must meet certain conditions: when an agent decides to go to an empty node, it would continue its route by nodes of this type until it does not find another node empty. For example, in the case of the Figure. 3 "A(1,3)" indicates that it is possible to have one to three Alanines, for this reason it need to include a given number of items as empty positions ((in this case 2).

Finally, when the stack is empty it stops the construction of the graph. In our approach, it builds the route graph using the first motif selected.

3.2 Walk of the ants on the route graph

The artificial ant's colony, as in natural ant colonies, evolves by the actions performed by its members. This way, the route graph is walked by the N-ants that constitute the colony. So, it is necessary to define the number of individuals of the colony, before they begin to walk on the route graph. In our case, each ant has a route map defined by the second motif to used. It defines an ant type data structure composed of 7 elements, whose characteristics are described in Table III. It contains the information necessary that the ants can to walk on the route graph.

It uses the second motif to construct the route map of the ants, transforming the motif in a stack data structure. The stack of motif 2 "L (2) - A (2) - Q" is shown in Figure. 5.

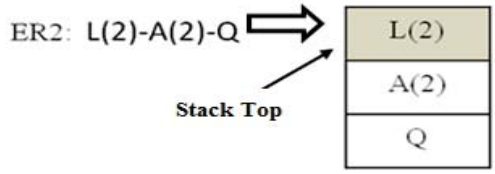


Figure 5. Transformation of a motif 2 in a stack

TABLE III
IDENTIFIER FOR SPECIAL NODES

Element	Characteristics
Start Node	Address of the node where start the ant to walk the route graph
Route Map	Stack that contains the regular expression that must follow the ant, and serves to know that nodes should be visited by the ant in the route graph
Pheromone Increase Coefficient	It is used to establish the pheromone concentration that deposits the ants in each visited node of the route graph.
Equalities Similarity Index	It determines the pheromone level deposited by the ant, when the node found in the graph is identical to the expected to the route map.
Families Similarity Index	It determines the pheromone level deposited by the ant, when the amino acid found in the route graph is not equal to the route map, but belongs to the same family of amino acid
Differences Similarity Index	It determines the pheromone level deposited by the ant, when the amino acid found in the route graph is not equal to the route map, and does not belong to the amino acid family
Gaps Similarity Index	It serves to mark the selected node, if node type is a Gap.

At the start, the ant is placed in the initial node of the route graph, and with the route map it observes the contiguous nodes at the right side (see Figure. 6).

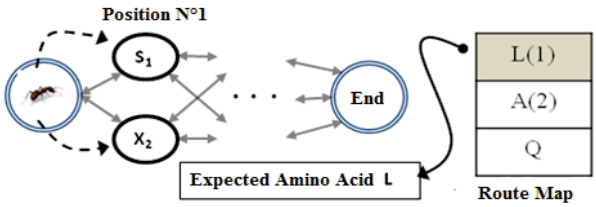


Figure 6. Ant in the initial node of the route graph

The ant executes the function of transition to each one of the nodes that can visit in the next position. This function consists of two phases; the first phase calculates the probability to visit each contiguous nodes ($P_n^k(r)$) (1) based on its pheromone level 'τ_r' and the index of similarity `$\varphi_r$' of each node ('**r**' indicates the neighboring node in the position `k', and `n' is the number of neighboring nodes at the right side for that position `k'.

$$P_n^k(r) = \begin{cases} \dfrac{\tau_r * \varphi_r}{\sum_{i=1}^{n} \tau_r * \varphi_r} & \sin > 1 \\ 1 & \sin = 1 \end{cases} \quad (1)$$

The second phase decides the node to visit using the simulation of Monte Carlo. When the ant moves to a node, it deposits pheromone that increases the pheromone concentration in the node. The quantity of pheromone depends on the similarity index with respect to the amino acid waited according to the route map (2).

$$\tau_r^k = \tau_r^k + \sigma * \varphi_r^k \quad (2)$$

The similarity index is defined as follows: if the amino acid of the route graph is equal to the amino acid of the route map of the ant, then it use the equalities similarity index; otherwise, if both belong to the same family, then the families similarity index, otherwise, if visited node contains gap, then the gaps index is used; otherwise, is used the differences similarity index. In our example, the final route of an ant is observed in the Fig.. 7.

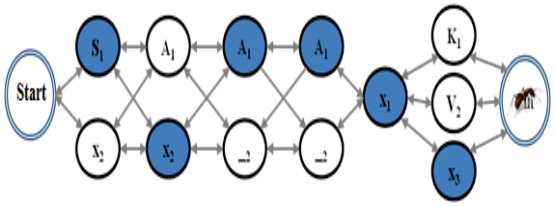


Figure 7. The final route of the ant

For a colony, the previous process is repeated for each ant. Additionally, the same process is executed recursively until the number of colony cycles desired. At the end of a cycle, there is an evaporate pheromone traces, decrementing the pheromone levels of all nodes in the graph (3), where "ρ" is the pheromone evaporation coefficient.

$$\tau_r^k = (1-\rho) * \tau_r^k \quad (3)$$

3.3 Choose the best nodes

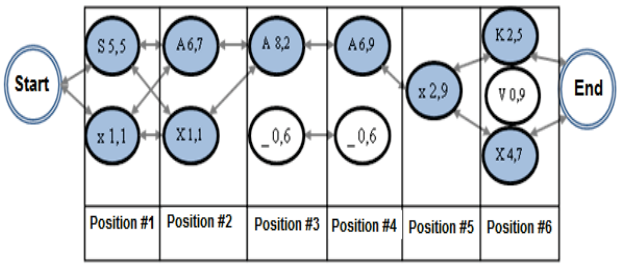


Figure 8. Route graph with pheromone levels of each node

Once the colony has completed its work, it delete the arcs that lead to those nodes with a pheromone level below the pheromone threshold that the user has defined (for our example, it fix the pheromone threshold to 1, 0), which help to preselect to the amino acids that contribute to the best solutions. Fig. 8 shows the nodes selected because they exceeded the threshold (in blue).

3.4 Construct of the resultant motifs

Finally, the selected route graph is filtered to delete irrelevant information and to define the resulting patterns. To make this task it analyzes the marked nodes of the graph, position by position, and inserts the amino acids selected in a list. To achieve this goal the following criteria are used:

- If in the position exists only one node (amino acid or gap) that has passed the pheromone threshold exit, it will be inserted in the list.
- If exist more than one node in the same position (amino acid or gap) that has passed the pheromone threshold exits. the following conditions apply:

  a) If the level of pheromone of the amino acids node is superior to the gap node it inserts the amino acid on the list.

  b) In other case it insert the gap in the list

  c) It applies the same conditions of the gap nodes for the empty nodes.

It takes the list that contains the amino acids corresponding to each position in order to construct the resultant motif, (Fig. 9)

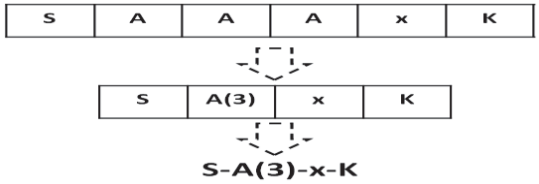


Figure 9. Resultant common pattern

4 Experiment

Amyloid term is used in biology to define a set of diseases characterized by the presence in specific organs (brain, kidney, eyes, skin, heart and pancreas) of insoluble deposits with proteins essentially. Only under pathological conditions these proteins have the capacity to change their structure and auto-assembles in fibers. [23], [24], [25]. To facilitate the comparison of amyloid proteins has been created the Amyloid Protein database (AMYPdb) [23]. The β-amyloid precursor protein (APP) y the TAU protein is connected to Alzheimer's disease by both biochemistry and genetic reasons. Therefore, the study of the APP will allow a major knowledge of the functioning of this in the development of Alzheimer's disease [26]. The motif proteins were taken from the database AMYPdb. To run the system is necessary to adjust a set of parameters. Because the number of adjustable parameters in the developed system is quite extensive, some values for the

tests were left fix (Table IV). The only parameters that it has varied are the parameters that determine the collective behavior: the cycle's number and the ant's number. This way, the solution depends fundamentally on the behavior of the colony.

TABLE IV
PARAMETER LIST

System Parameters	Value
Pheromone Increase Coefficient	0,1
Similarity indices for the amino acids that are the same	10
Similarity indices for the amino acids that belong to a family	8
Similarity indices for the amino acids that are different	1
Similarity indices for Gaps	3
Approving Similarity Index	3
Failures Maximum number	0
Pheromone Initial level on the graph nodes	1,0
Pheromone evaporation coefficient	0,05

4.1 Construct Common Pattern of [ST]-x(2)-[ST]-x-[RT] with [ST]-x-[RK]

It's possible to observe that with the motif "[ST]-x(2)-[ST]" can be obtained a chain of 4 amino acids (for example, SAKT), whereas with the motif "[ST]-x-[RK]" is obtained a chain of 3 amino acids (for example, TER). Our algorithm takes the motif "[ST]-x(2)-[ST]" as motif 1 for the construction of the route graph (the longest), and the other motif 2 (with it the ants define the route map). In addition, the pheromone threshold is equal to the pheromone initial level in the nodes (1,0).

For the tests, thirty Common Patterns of the two motifs with the same group of parameters were studied. Also, it study the average time used by the algorithm for obtaining the respective solutions and thus determine when a set of parameters is better than another.

The first test was realized for the parameters set: ant's numbers = 4, cycle's number = 4. It observes that the algorithm converges in 90% of the times, with an average time of 0.76 seconds. The second test was realized for the parameters set: ant's number = 4, cycle's number = 8. It observes that the algorithm converges in a 92.33% of the times . In addition, the algorithm had an average time of 1.10 seconds. The third test was realized for the parameters set: ant's number = 8, cycle's number = 4. The algorithm converges in 96.66% of the times and it presents a best runtime with respect to the previous tests, the average time is 0.89 seconds. The last test is for the next parameters: ants number = 8, cycles number = 8, obtaining with this set a better precision level, since it has a convergence of 100% compared to the expected pattern. Additionally, the average time was 1.09 seconds.

The best resultant pattern of the two motif proteins is "[ST]-x(3)". It conclude that It can obtain the common pattern of the motifs [ST]-x (2)-[ST] and [ST]-x-[RK] in a very short time, when the ants and cycles number are equal to the positions number of motif1; Nevertheless, the best performance is obtained when the ants and cycles number duplicates the positions number of motif1 (the selected pattern to construct the route graph) with a very similar runtime.

4.2 Biological Patterns

Now perform the Construction Common Patterns of two motifs and analyze its biological sense, to see if the patterns generated by the system are useful for the study of protein chains. The first motif is:

K-x-G-S-L-[DGK]-N-[AIV]-T-H-V-[AP]-G-G-G-[AHN]-[KV]-[KQ]-I-E-[NST]-[HR]-K-L-[DST]-F-[RS]-x-[AN]-[AS]-[KP]-x-[KV]-[GT]-[DS]-[HK]-[GT]-[AN]-[EY]-[IQ]-[PV]-x-K-S-[DP]-[GV]-[HKV]

The second motif is:

G-S-[KT]-D-N-[IM]-[KNR]-H-x-P-G-G-G-[KNS]-V-Q-I-[FV]-[DHY]-[EK]

The parameters are taken from the Table IV. In addition, 94 ants and equal number of colony cycles used. The system execution is realized 3 times and the following results are obtained (Table V).

TABLE V
RESULTANT COMMON PATTERN

Run	Pattern
1	x(2)-G-S-x-[DGK]-N-[AIV]-T-H-x-[AP]-G(3)-[HN]-[KV]-Q-I-x(2)-[HR]-K-(24)
2	x(2)-G-S-x-[D**G**K]-N-[AIV]-T-H-x-P-G(3)-[**A**HN]-V-Q-I-x(2)-[H**R**]-K-x(24)
3	x(2)-G-S-x-[DK]-N-[AIV]-T-H-x-[**A**P]-G(3)-[**A**HN]-V-Q-I-x(2)-H-K-x(24)

5 Conparison with other works

The Table VI compared our proposal with work based Ant Colony that use protein motifs and /or DNA sequences. Our approach suggests a motif for the construction of the route graph and other motif defines the route map that the ants used for walking. In addition, ants executed the transition function for each node that can be visited in the next position using the similarity index between amino acids map and graph nodes.

To a qualitative comparison of our method with previous work, we carry out experiments on real datasets previously constructed in [27] (Table VII). In this case we compare S1 with S2, the resultant motif with S3, and so on.

The subsequence ATCCGT is the consensus sequence. This study suggests that the results provided by our system are similar to the results that are found in [27], with the additional advantage that our system does not require the use of post – processing. We carry out a second qualitative comparison with real datasets of the Escherichia coli sequences (they have two highly conserved parts, called the -35 and -10 regions) [28]. The fusion of a set of these sequences is shown in Table VIII. In [28] is not presented the consensus motif of each

fusion. In our case, we fuse the motif resulting of the two previous rows with the following until the end.

TABLE VI
COMPARISON WITH OTHER WORKS

Characteristic	Our Approach	Bouamama S et al, 2010	CHEN-HONG Y ET AL, 2011
Process Modeling	Weighted directed graph. The positions in the node representing each amino acids in the motifs. Special nodes for specific cases.	Weighted directed graph. Where there are 4 possible letters representing nucleotides for each position	Weighted directed graph. Where there are 4 possible letters representing nucleotides for each position
Motif Elements	20 letters (A, C, E, F, G, H, I, K, L, M, N, P, Q, R, S; T, V, W, Y) that represent the amino acids and "x" represent a gap	4 nucleotides (A, C, G, T)	4 nucleotides (A, C, G, T)
Similarity Measure	Expected Similarity index in a node	Consensus score and contained information	Consensus score and contained information
Post-Processing	No	No	Yes

TABLE VII
MOTIFS FUSION

Sequences	[Wei Z and Jensen T, 2006]	Our Approach
S1 : ATCATCCGTGTAGCTCAAAA S2: : ATCATCCGTGTAGCTCAAAA	ATCATCCGTGTAGCTCAAAA	ATCATCCGTGTAGCTCAAAA
S3 : AGATCCGTAACGAAGTTTAC	ATCCGT	AxxATCCGTxxxGxxxxxxA
S4 : CCCCATCCGTAATTACCTAT	ATCCGT	xxxxATCCGTxxxxxxxxxxx

According to [28] the consensus sequences are TTGACA and TATAAT. In our case, the consensus sequence is TGTGA. In contrast with [28], we obtained a consensus sequence for all sequences in the Table VIII. Our system features well-conserved positions, in [Stormo G. and Hartzell G, 1989] it is unclear what positions are absolutely conserved.

TABLE VIII
FUSION RESULTS OF THE SEQUENCES OF ESCHERICHIA COLI.

Sequences	Common Patterns
Bgl R mut : A A C T G T G A G C A T G G T C A T A T T T	A(2)-C-T-G-T-G-A-G-C-A-T-G(2)-T-C-A-T-A-T(3)
Deo P2 site 1 : A A T T G T G A T G T G T A T C G A A G T G	A(2)-x-T-G-T-G-A-x(6)-T-C-x(2)-A-x-T-x
Lac site 1: T A A T G T G A G T T A G C T C A C T C A T	x-A-x-T-G-T-G-A-x(6)-T-C-x(6)
Lac site 2: A A T T G T G A G C G G A T A A C A A T T T	x-A-x-T-G-T-G-A-x(14)
Mal k: T T C T G T G A A C T A A A C C G A G G T C	x(3)-T-G-T-G-A-x(14)
Mal T: A A T T G T G A C A C A G T G C A A A T T C	x(3)-T-G-T-G-A-x(14)
Tna A: G A T T G T G A T T C G A T T C A C A T T T	x(3)-T-G-T-G-A-x(14)
Uxu AB: T G T T G T G A T G T G G T T A A C C C A A	x(3)-T-G-T-G-A-x(14)
pBR P4: C G G T G T G A A A T A C C G C A C A G A T	x(3)-T-G-T-G-A-x(14)
Cat site 2: A C C T G T G A C G G A A G A T C A C T T C	x(3)-T-G-T-G-A-x(14)
Tdc: A T T T G T G A G T G G T C G C A C A T A T	x(3)-T-G-T-G-A-x(14)

6 Conclusions

Our approach can the construction common patterns denoted as regular expressions composed of 20 amino acid (denoted by the letters A, C, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y) using the rules PROSITE. The [21] and [22] approaches only used to fuse motifs in DNA sequences that contain four nucleotides (A, T, C, G). We propose a motif for the construction of the route graph and other motif defines the route map that the ants use to walk. In addition, the ants execute the transition function to each one of the nodes that it can visit in the next position using the similarity index between the nodes of the route map and of the route graph. The algorithm based on Ant Colony Optimization is a good solution for the construction common patterns. It's possible to find common patterns for groups of protein motifs with biological sense. Our approach can build common patterns of protein motifs without a perceptible cost, the processes is quick. The main parameters of our algorithm are determined for the collective behavior of the ant: cycle's number of the colony, ant's number.

Acknowledgment

This work was supported for the CDCHT project I-1407-14-02-B of the Universidad de Los Andes. Dr Aguilar has been partially supported by the Prometeo Project of the Ministry of Higher Education, Science, Technology and Innovation of the Republic of Ecuador.

7 References

[1] Dorigo M., Birattari M, Stützle T. **"**Ant colony optimization: Artificial ants as a computational intelligence technique" **.** *IEEE Computational Intelligence Magazine,* vol. 1 n° 4: pp. 28-39. 2006

[2] *Mathura V., Kangueane P. "Bioinformatic A Concept-Based Introduction". Springer. 2009.*

[3] *Pevsner J. Bioinformatics and Functional Genomics. Second Edition. Wiley – Backitll. 2009.*

[4] *Srinivas V. Bioinformatics. A modern Approach. Eastern Economy. 2005.*

[5] *Searls D., "A primer in macromolecular linguistics" Biopolymers. Special Issue: PDB40: vol. 99, Issue 3. pp. 203-217. 2013*

[6] *Dyrka W., Nebel J., A stochastic context free grammar based framework for analysus if protein sequences BMC vol. 10, n°323. 2009*

[7] *Eddy, S. The HMMER User's Guide (Howard Hughes Medical Institute and Dept. of Genetics Washington* University School of Medicine, 660 South Euclid Avenue, Box 8232 Saint Louis, Missouri 63110, USA, vesion 2.3.2 edition. hmmer.wustl.edu. 2003

[8] Sandve G., Drablos F. A survey of motif discovery methods in an integrated framework, Biology Direct, vol 1. n° 11. 2006

[9] Habib N., Kaplan T., MArgalit H., Friedman N. A novel Bayesian DNA Mofit Comparison Method for clustering and retrieval, Plos *Comput. Biol, vol.* 4. n° 2, pp. 1-17, 2008

[10] Pratt Pattern Matching. *Available in: http://www.ebi.ac.uk/Tools/pratt/.*

[11] Gusfield D., *Computer Science and Computational Biology*, Press University of Cambridge, 1999.

[12] Teiresias. *Available in: http://cbcsrv.watson.ibm.com/Tspd.html.*

[13] Meme. *Available in: http://meme.sdsc.edu/meme/doc/examples/meme_example_output_files/meme.html.*

[14] Bailey TL., Boden M., Buske FA., Frith M., Grant CE., Clementi L., Ren J., Li WW., Noble Ws."MEME Suite: tools for motif discovery and searching", *Nucleic Acids Research.*, 37, pp. W202-W208, 2009.

[15] Dogruel M., Down T., Hubbard T. "NestedMICA as an ab initio protein motif discovery tool,. *BMC Bioinformatics*, 9(19), pp 1-12. 2008.

[16] Corne D., Meade A., Sibly R."Evolving core promoter signal motifs", *Proc. 2001 Congress on Evolutionary Computation*, pp. 1162-1169. 2001

[17] Fogel G., Itekes D., Varga G., Dow E., Harlow H., Onyia J.,Su C.,*"Discovery of sequence motifs related to coexpression ofgenes using evolutionary computation," Nucleic Acids Research 32:*3826-3835. 2004

[18] Edwards R., Davey N., Shields D. "CompariMotif: quick and easy comparisons of sequence motifs", *Bioinformatics*,24(19), pp. 1307-1309, 2008.

[19] FunClust. *Available in: http://pdbfun.uniroma2.it/funclust/.*

[20] Bio.Motif. *Available in: http://www.biocloud.info/Biopython/en/ch13.html#motifobjects.*

[21] Bouamama S., Boukerram A. and Al-Badarneh A., Motif Finding using Ant Colony Optimization, Dorigo M. et al (Eds.). *ANTS 2010,* Springer-Verlang Berlin Heidelberg, LNCS, 6234, 464, 2010.

[22] Chen-Hong Y., Yu-Tang L., and Li-Yeh C. DNA Motif Discovery Based on Ant Colony Optimization and Expectation Maximization. *IMECS 2011*, 1, 169, 2011

[23] S. Pawlicki, A. Le Béchec. C. Delamarche. AMYPdb : A database dedicated to amyloid precursor proteins, BMC Bioinformatics, Vol. 9, 273-28 200

[24] J. Sipe, A. Cohen, Review : History of the Amyloid Fibril, Journal of Structural Biology, Vol. 130, pp. 88-98 2000

[25] P. Itstermark. Classification of amyloid fibril proteins and their precursors: An ongoing discussion. Amyloid: the Journal of Protein Folding Disorders, Vol. 4, pp. 216-218. 1997

[26] W. Xia, H. Xu, Amyloid Precursor Protein. A Practical Approach, CRC Press 2005.

[27] Wei Z., Jensen T. "GAME: detecting cisregulatory elements using a genetic algorith". *Bioinformatics*, Vol. 22, pp. 1577-84. 2006

[28] Stormo G., Hartzell G. "Identifing proteinbinding site from unaligned DNA fragments". *Proc. Natl. Acad. Sci*, Vol. 86(4), pp. 1183-1187. 1989

Revealing protein-ligand interaction patterns through frequent subgraph mining

Sabrina A. Silveira[1], **Alexandre V. Fassio**[2], **Carlos H. da Silveira**[3] **and Raquel C. de Melo-Minardi**[2]
[1]Department of Informatics, Universidade Federal de Viçosa, Viçosa, Minas Gerais, Brazil
[2]Department of Computer Science, Universidade Federal de Minas Gerais, Belo Horizonte, Minas Gerais, Brazil
[3]Advanced Campus at Itabira, Universidade Federal de Itajubá, Itabira, Minas Gerais, Brazil

Abstract— *Molecular recognition plays an important role in biological systems. In this paper, we are interested in receptor (protein) and ligand (non-protein) interactions what consists of noncovalent bonding such as aromatic stackings, hydrogen bonds, hydrophobic interactions and salt bridges. Understanding and predicting protein-ligand interactions is a complex task and in this paper we propose a model and algorithms to understand why different small molecules are recognized by a specific protein. Our model is based on graphs. Each protein-ligand complex is a bigraph where nodes are atoms and edges depicts interactions between protein and ligand. The proposed algorithms aim to detect conserved subgraphs in the dataset of graphs representing protein-ligand complex interactions. We also propose a visual interface where users can find general statistics about the dataset, the type of atoms and interactions established as well as select and analyze the generated patterns. We show an example of use of this methodology with Ricin and CDK datasets, both with their respective ligands.*
Availability: *A prototype of the visualization tool with the examples mentioned in the paper can be found at*
`http://www.dcc.ufmg.br/~alexandrefassio/biocomp`
Supplementary file:
`http://homepages.dcc.ufmg.br/~alexandrefassio/biocomp/files/suplementar.pdf`

Keywords: Protein, Clustering, SVD, Graph, Pattern, Visualization

1. Introduction

Molecular recognition plays an important role in biological systems. It refers to interactions between two or more molecules through noncovalent bonding such as aromatic stacking, hydrogen bonding, hydrophobic forces and salt bridges. Solvent can play a dominant indirect role in driving molecular recognition in solution as well. The conditions responsible for the binding and interaction of two or more molecules are a combination of conformational and physicochemical complementarity [1]. Understanding and predicting protein-ligand interactions are essential steps towards ligand prediction, target identification, lead discovery and drug design [2].

In this paper we propose a model and algorithms to understand why different small molecules are recognized by a specific protein structure. It can be quite tricky because of protein promiscuity [3], [4], [5] what leads to very dissimilar molecules being recognized by the same protein as its substrate or even acting as an inhibitor.

Despite the existence of several methods designed to predict protein ligands, few methodologies are devised to identify and describe what intelligible factors that imply in protein ligand affinity.

A straightforward approach would be based on the description of a protein ligand recognition as a bigraph where nodes represent protein atoms labeled by their physicochemical properties and edges depict possible interactions between them. A bigraph or a bipartite graph is a graph whose vertices can be divided into two disjoint sets U and V (that is, U and V are each independent sets) such that every edge connects a vertex in U to one in V. Vertex set U and V are often denoted as partite sets. In this case, protein atoms would compose set U and ligand atoms, set V.

Having this model based in the explained graphs, a possible algorithm to solve the problem of selecting key factors to describe molecular recognition can be based on graph mining and searching for frequent subgraphs in the set of graphs representing several protein ligand interactions. In this approach, conserved subgraphs would represent emerging patterns responsible for protein ligand interaction and molecular recognition.

2. Methods

In this section we detail the modeling and the experiments performed, which are a clustering analysis and a frequent subgraph pattern mining, as well as the proposed visual interactive representations.

2.1 Data

We collected our Ricin and CDK datasets from Protein Data Bank (PDB) [6] in August 2014. In the case of ricin, we searched for key words *ricin*, *ricin-like* and *ribosome inactivating protein* and obtained 136, 126 and 163 results respectively, totaling 266 PDB structures, as there is overlap between the results for each keyword.

Sequences from all 266 PDB entries were aligned using an inhouse implementation of Needleman-Wunsch algorithm [7] against PDB 2AAI [8] chain A, from now on called 2AAI.A, the catalytic subunit of ricin toxin, and those 47

structures which have identity percentage greater than or equal to 50% were considered as our initial ricin dataset.

As we are interested in patterns of interaction between a protein and its ligands, the next step is to select entries which have at least one ligand. So we split entries by chain and computed probable protein-ligand interactions at atomic level using a geometric approach (which is detailed in Section 2.2) to determine ligands that are interacting with protein residues. Only ligands with seven or more atoms were considered, in a similar manner to [2]. This process resulted in 29 PDB chains, from which we selected only interactions between a protein atom and a ligand atom, discarding interactions established among protein atoms.

In the case of CDK dataset, we obtained the 75 chains based on the study conducted by [9] and the same process was applied.

Table 1: PDB entries from ricin dataset

PDB id and chain		
1BR5.A	1RZO.A	3RTI.B
1BR6.A	1RZO.B	3RTJ.B
1IFS.A	1RZO.C	4ESI.A
1IFU.A	1RZO.D	4HUO.X
1IL3.A	2P8N.A	4HUP.X
1IL4.A	3EJ5.X	4HV3.A
1IL5.A	3HIO.A	4HV7.X
1IL9.A	3PX8.X	4MX1.A
1J1M.A	3PX9.X	4MX5.X
1OBT.A	3RTI.A	

Table 2: PDB entries from CDK2 dataset

PDB id and chain				
3QL8.A	3QQF.A	3QQG.A	3QQH.A	3QQJ.A
3QQK.A	3QQL.A	3QRT.A	3QRU.A	3QTQ.A
3QTR.A	3QTS.A	3QTU.A	3QTW.A	3QTX.A
3QTZ.A	3QU0.A	3QWJ.A	3QWK.A	3QX2.A
3QX4.A	3QXO.A	3QXP.A	3QZF.A	3QZG.A
3QZH.A	3QZI.A	3R1Q.A	3R1S.A	3R1Y.A
3R28.A	3R6X.A	3R71.A	3R73.A	3R7E.A
3R7I.A	3R7U.A	3R7V.A	3R7Y.A	3R83.A
3R8L.A	3R8M.A	3R8P.A	3R8U.A	3R8V.A
3R8Z.A	3R9D.A	3R9H.A	3R9N.A	3R9O.A
3RAH.A	3RAI.A	3RAK.A	3RAL.A	3RJC.A
3RK5.A	3RK7.A	3RK9.A	3RKB.A	3RM6.A
3RM7.A	3RMF.A	3RNI.A	3ROY.A	3RPO.A
3RPR.A	3RPV.A	3RPY.A	3RZB.A	3S00.A
3S0O.A	3S1H.A	3SQQ.A		

2.2 Problem Modeling

2.2.1 Proteins and ligands as graphs

We modeled proteins and its ligands as graphs in which atoms are nodes and interactions between atoms are edges. To do so, we computed the interactions among protein and ligand atoms according to a geometric approach, which is cutoff-independent [10]. For each protein chain and its ligands, we used the CGAL software library [11] to build a Voronoi diagram followed by its Delaunay tessellation [12], [13] and, using a distance criteria and physicochemical properties, we labelled nodes and edges. It is important to point out that although Delaunay tesselation gives as a result all potential interactions among atoms, regardless of whether they are from protein or ligand, we filtered only those interactions involving a protein and a ligand atom.

Protein nodes (atoms) were labelled as charged attractive, charged repulsive, aromatic, hydrophobic, donor or acceptor according to [14]. Ligand nodes (atoms) were labelled with the same types using PMapper (Pmapper 5.3.8, 2010, Chemaxon [1]) software as [14] concerns physicochemical properties only for protein atoms.

It is important to note that PMapper computes pharmacophoric properties of atoms of given molecular structures in isolation, which means that the molecule (ligand), when interacting with a protein, can exhibit different properties. We are aware of this issue and working to improve the assignment of ligand atoms properties.

Edges (interactions) were labelled according to both distance criteria (provided in Table 3) and the type of its nodes, as aromatic stacking, hydrogen bond, hydrophobic, repulsive and salt bridge.

Table 3: Distance criteria to compute interactions.

Type of interaction	Distance (d) in Å
Aromatic stacking	$1.5 \leq d \leq 3.5$
Hydrogen bond	$2.0 \leq d \leq 3.2$
Hydrophobic	$2.0 \leq d \leq 3.8$
Repulsive	$2.0 \leq d \leq 6.0$
Salt bridge	$2.0 \leq d \leq 6.0$

Some nodes (atoms) can have more than one label and the same happens to the edges (interactions) as they depend on node labels. For instance, the oxygen OD1 from ASN is labelled as possible acceptor/donor and the O2 from GAL ligand as acceptor/donor. Therefore, these nodes have potential to interact through two hydrogen bonds. This can be seen in 1RZO.C from group 3, where nodes ASN:1010:OD1 and GAL:5501:02 are connected by and edge labelled with HB/HB (two hydrogen bonds).

2.2.2 Counting matrix

We propose a counting matrix to model a protein-ligand dataset in terms of the labels of nodes in the end of edges. In such a matrix, each row represents an instance of a dataset, in this case a protein chain, and each column represents a pair of node labels in the end of an edge. Suppose, for example, that we have, in a certain dataset, node labels A (aromatic), B (acceptor) and C (donor) and that we have protein chains x, y and z. The counting matrix for this dataset is provided in Table 4. The position (i, j), lets say $(1, 5)$ in

[1] http://www.chemaxon.com

this matrix is 3, which means that the protein chain x has 3 edges (interactions) whose ends are labels B (acceptor) and C (donor). For our ricin dataset, we have 29 rows and 79 columns and for CDK dataset we have 73 rows and 55 columns.

Table 4: Example of a counting matrix

Protein chain	AA	AB	AC	BB	BC	CC
x	2	0	1	0	3	0
y	0	0	0	0	1	0
z	0	0	1	0	0	4

2.3 Technique

2.3.1 Dimensionality and noise reduction strategy

After generating the counting matrix, we apply Singular Value Decomposition (SVD) to reduce dimensionality and noise. Then the processed matrix is finally submitted to the clustering algorithm. Although in our case matrix dimension is not critical (79 columns for ricin and 55 for CDK), both matrices are sparse and we have the benefit of noise reduction.

SVD is a linear algebra technique in which an m by n matrix A can be represented by the product $U\Sigma V^T$ where U is an m by m matrix and its columns are the left singular vectors of A; Σ is an m by n diagonal matrix with its values in descending order; and V is an n by n matrix and its columns represents right singular vectors of A. Matrix A can be approximated by matrix A_k (with rank k where k is less than the rank of A) as: $A_k = U_k\Sigma_k V_k^T$.

To achieve A_k, the first k singular values of A and their singular vectors were taken, and thus the resulting matrix has k features: $A_k = U_k\Sigma_k V_k^T = U_k(\Sigma_k V_k^T) = U_k(D_k)$. According to [15], A_k can be computed using only matrix D_k, which is: $D_k = \Sigma_k V_k^T$.

Here matrix A was approximated by D_k. As stated by [16], the choice of k is an empirical matter; therefore approximations with all possible values of k were generated, and the matrix that led to the best clustering result was chosen.

2.3.2 Clustering

Cluster analysis is an unsupervised learning strategy, which means that it analyzes data objects without consulting a known class label [17]. In general, the class labels are not present because they are not previously known. Clusters of instances are formed so that objects within a cluster have high similarity in comparison to one another, but are very dissimilar to objects in other clusters.

Here we used the partitioning algorithm k-medoids that organizes the objects into k partitions ($k < n$ and n is the number of instances in the dataset), where each partition is a cluster. The implementation of k-medoids employed is *pamk* from *rpc* [18] R [19] package version 3.0.2, which performs a partitioning around medoids clustering with the number of clusters estimated by optimum *average silhouette width*.

To perform the clustering, we provided all reduced matrices resulting from SVD, in a similar manner to [20], to k-medoids algorithm and, for each matrix, we varied the number of groups in k-medoids, to choose the best clustering result (with the best groups).

2.3.3 Clustering evaluation strategy

We performed several experiments to choose the optimal number of clusters. We used all matrices resulting from SVD and, for each matrix, we applied the clustering algorithm k-medoids with k varying over the number of instances of each dataset (ricin and CDK) . To assess its performance we employed the metric *average silhouette width* (asw), choosing the group from k-medoids with the highest value for asw. Average $s(i)$ over all data of a cluster can be taken to mean that the clustering algorithm has discovered a very strong clustering structure. When the algorithm does not succeed in clustering, the overall aws tends to become very low. Hence silhouette plots and averages may be used to determine the natural number of clusters within a dataset. We detail how to compute such a metric and its meaning in Section *Average silhouette width* from *Supporting material*.

2.3.4 Frequent subgraph mining

The problem of frequent subgraph mining (FSM) is to find the most frequent subgraph structures in a set of graphs [21]. It can be used to identify relevant patterns in social, biological, chemical, and technological networks and graphs [22]. The core of frequent subgraph mining is subgraph isomorphism test, which is an NP-complete problem. Therefore exact solutions are time consuming and can just mine small input graph datasets.

Here we apply FSM to the groups resulting from the best clustering result to extract relevant and non-trivial patterns from protein-ligand interactions. The FSM algorithm applied was gSpan [23], in which given a graph dataset $D = \{G_0, G_1, ..., G_n\}$, $support(g)$ denotes the number of graphs in D which have g as a subgraph. So gSpan aims to find any subgraph g with $support(g) \geq minSup$ (a minimum support threshold).

Data mining techniques in general demand visual representations that aid to shed light, explore and make sense of important features obtained and to extract meaningful information [24]. Thus we decided to vary the value of gSpan support and show all resulting patterns. Such patterns, when exhibited in an appropriate visual representation, could help the domain expert to make sense of common and relevant aspects of protein-ligand interaction.

For each group (from clustering algorithm), we executed gSpan varying the support from 0.6 to 1.0. As the value of support increases, we find patterns that are in a high number

of input graphs. However, the size of the patterns tends to decrease, as expected because it is difficult to find big graphs which occur in the whole dataset. All frequent patterns extracted were summarized in our interactive visualizations, containing tables, graphs and images. Also, it is possible to visualize each pattern by clicking on a specific position in the table (according to support and number of nodes in the pattern).

2.4 Visual strategies

In this work we proposed a set of visual and interactive interfaces to depict protein-ligand interactions by using techniques of data visualization. We provide a general view (all clusters and its respectives PDB ids and ligands) and present details on demand (patterns for each dataset and graphs and statistics to characterize such patterns) as performed in [25].

We present a table with basic information relative to our dataset and each group of PDB ids obtained. This table has two headers - *Group* and *PDB ids* - and some interactive features like searching for any of its fields and sorting possibilities by its headers.

Fig. 1: Example of visualization screen showing the labeled input graphs for cluster 3.

Also, we designed a visual interactive interface in order to enable users to find the distribution of number of atoms (vertices) and interactions (edges) of a specific type in a histogram. Users can choose to analyze the data by grouping bars by the cluster and coloring them by the support (Figures 2 and 3) or grouping by the support and coloring by the cluster (Figure 4).

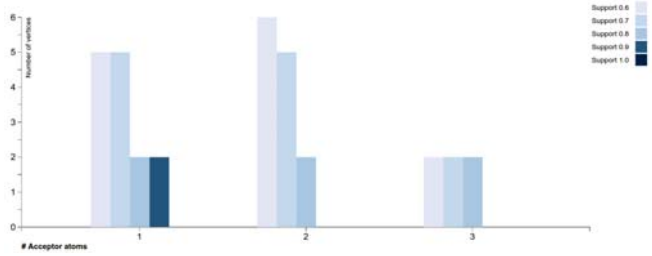


Fig. 2: Grouped bar chart showing the number of vertices representing acceptor atoms grouped by the number of the cluster and colored by the support for the ricin dataset.

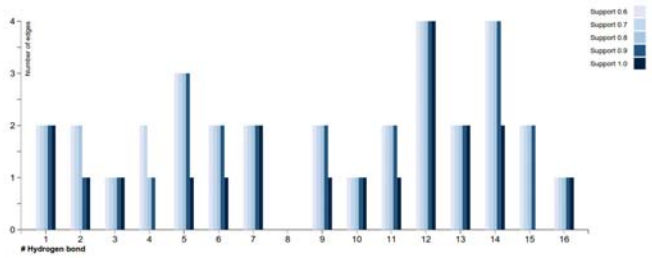


Fig. 3: Grouped bar chart showing the number of edges representing hydrogen bonds grouped by the number of the cluster and colored by the support for the CDK2 dataset (Section 3.3.2).

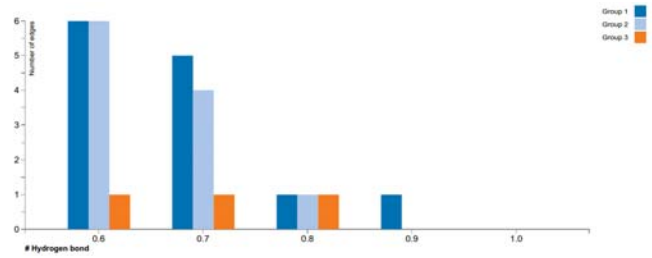


Fig. 4: Grouped bar chart showing the number of edges representing hydrogen bonds grouped by support and colored by the number of the cluster for the ricin dataset.

Concerning the existing patterns relative to the atom types, we developed two interactive tables, which we called *Grouping columns* and *Simple table*, where users can analyze the pattern size and the occurrence of each pattern, i.e., in how many graphs the pattern can be found.

3. Results and discussion

3.1 Clustering

Figure 7 shows a clustering scanning where we can observe the PAM behavior in function of the number of clusters (k) and the number of singular vector (DIM) in SVD dimensionality reduction. The criterion for cluster evaluation was the silhouette average width (avg.width). DIM(x:y) specifies the range of singular vectors used to compose the clustering matrix for PAM. For example, DIM(1:3) indicates that the first 3 highest singular vectors (corresponding to the first 3 highest singular values) were used. DIM(1:1) expresses that only the 1st singular vector was utilized, DIM(2:2) only the 2nd, and so on. The subtitle "all" denotes that no SVD was previously applied and that the whole original matrix was submitted to PAM clustering.

The best silhouette parameter occurred when the 1st or the 3rd highest singular vectors were used, with an avg.width around 0.7. All remaining combinations of singular vectors tested exhibit a lower silhouette profile than these first two. The worst case was when we do not use SVD.

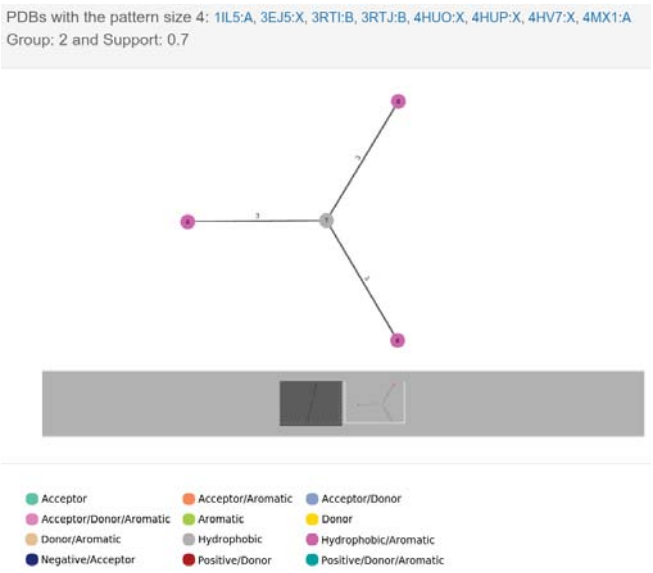


Fig. 5: Screen with examples of frequent patterns.

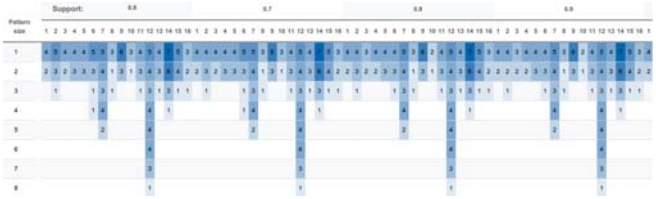


Fig. 6: Summary table presenting the different number of patterns found for different supports (first header line), clusters (second header line) and pattern size (number of nodes) (first header column).

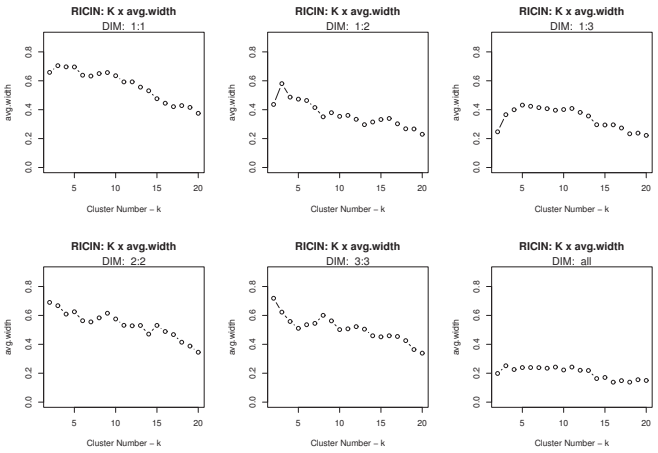


Fig. 7: Clustering Scanning

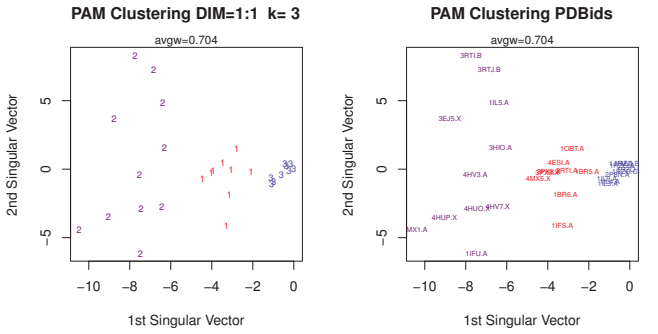


Fig. 8: Best PAM results

In Figure 8 we can have a visual idea of the groups using the two highest singular vectors as plot axes. In the left graphic, we can see the distribution of 3 clusters found by PAM DIM(1:1) with maximum avg.width. In the right graphic, we discriminate the PDB ids to facilitate the identification of ligand entries. As we can observe, it is possible identify two regions of more dense clustering, and a third region with more scattered points. One interesting example, is the cluster number 3 in left graphic formed by PDB ids 1IL9.A, 1IL4.A, 1IL3.A and 2P8N. All are analogous of guanine or adenine. This indicates how our method can be used to suggest potential ligands. Tables 1 and 2 from *Supporting Material* provide the best *asw* and number of groups for k-medoids clustering algorithm applied to ricin and CDK dataset (respectively) for all results from SVD. Figures 1 and 2 from *Supporting Material* show clustering analysis for ricin and CDK dataset respectively.

3.2 Visual analysis

In this section we will present some insights obtained through the interactive visualizations. In Figure 1, we depict one of the clusters used as input to search for frequent patterns. Figure 6 shows an example of a summarized output where users can find out the sizes and frequencies of the patterns found for each of the clusters using all possible supports. It is a simple table but helps to synthesize all possible experiments and decide about what parameters to use in order to achieve a desirable number of patterns

Figures 2 and 3 present a grouped histogram illustrating the frequencies of the number of atoms and edges respectively for aromatic atoms and aromatic stackings. Both are grouped by cluster number and colored by support. Users can invert the analysis, grouping by support and coloring by cluster number as shown in Figure 4.

Finally, Figure 5 presents an example of how the frequent patterns can be visualized. In the screen users can find the list of PDB ids that present the pattern, the number of the cluster and the support of occurrence besides several pictures showing the frequent subgraphs colored by atom types.

3.3 Interaction analysis

In the next two sections we present and discuss some results from clustering analysis and frequent pattern mining

for ricin and CDK datasets in terms of how some relevant interactions known in the literature were grouped and we compared how such interactions were considered as conserved patterns across groups or not. The distribution of protein chains among groups can be accessed in the initial screen of our visualization tool by clicking on *Experiments* and selecting the desired dataset (Ricin with 3 clusters or CDK with 16 clusters). The conserved patterns are provided in the section *Graph patterns table* by clicking on the values from *Occurrences* column.

3.3.1 Ricin

Here we briefly comment on some points our study is in agreement with [26] and also discuss some different aspects to illustrate the potential of our strategy.

In [26] authors presented a structure of ricin A chain in complex with transition state analogue inhibitor which mimic the sarcin-ricin recognition loop of 28S rRNA and the dissociative ribocation transition state established for ricin A chain catalysis. They analyzed how this enzyme works, in terms of molecular recognition and catalytic activity and highlighted some residues that play an important role in ricin activity. Also, authors pointed out the lack of structures with catalytic significance for ribosome-inactivating proteins (RIPs), a problem which impacts in our work, as we have scarce data. The PDB id 3HIO presented by [26] is within our ricin dataset in group 2.

Tyr-123 is considered an important residue of 3HIO.A as it shares a hydrogen bond with catalytic site Glu-177 and forms a quadruple π-stack with the first guanine of the inhibitor (C2X) and Tyr-80. Also Tyr-123 interacts with Arg-134 through a cation-π interaction. In our modeling we focused only on interactions involving a protein and a ligand atom, hence hydrogen bond between protein atoms Tyr-123 and Glu-177 is not represented in our data as expected. The same holds for Tyr-123 and Arg-134 interaction. We do represent the π-stack among protein and ligand atoms (Tyr-80:CE2 and C2X-268:N9G).

In accordance with [26], atoms N1, N6 and N7 of ligand (C2X) participate in hydrogen-bonding network with Val-81 amide and the Gly-121 carbonyl. Our modeling captures this information as in the 3HIO.A graph (group 2) Val-81:N and C2X-268:N1 establish a HB and Gly-121:O can potentially establish 1 HB with C2X-268:N6 and 1 HB with C2X-268:N7.

Arg-258 guanidino group interacts with the second and fourth phosphodiester groups according to [26]. Our modeling captures this interaction as Arg-258:NH1 interacts with C2X-268:O1Z through a salt bridge (SB) and with C2X-268:O2V through a SB and a HB. Also, in our modeling Arg-258:NH2 interacts with C2X-268:O2V through a SB.

Regarding our FSM analysis, considering $support = 0.7$ and patterns with 2 or more nodes for group 2 (where 3HIO.A was clustered) we obtained 3 hydrophobic interactions (HP), 2 hydrogen bonds (HB) and one aromatic interaction or π-stack (AR) as frequent patterns. We believe our HB and AR patterns are relevant in accordance with [26], as showed above. We need further investigation to understand the importance of HP interaction pattern found. Also, there are SB interactions in 3HIO.A that were not found as frequent pattern, which means that although they are relevant for 3HIO.A interaction with C2X, they may not be a relevant pattern for the group 2 (it is not a conserved interaction between ricin ids from group 2 and their respective ligands). We believe we should test if our modeling and results can be improved by considering all interactions established by protein atoms which interacts with a ligand atom (even if such interactions are between protein atoms) because they can disrupt protein-ligand interaction.

The residues and atoms cited in this analysis are highlighted in Figure 3 from Supporting Material. Frequent patterns are provided through our interactive visualization tool in *Graph patterns table*, choosing group 2 and *Minimum pattern size* 2.

3.3.2 CDK2

In [9], authors discovered by high throughput screening the (2-(allylamino)-4-aminothiazol-5-yl-(phenyl)methanone) as an inhibitor of the human CDK2-cyclin A2 complex with an IC_{50} value of 15μM. They also have done a co-crystal structure (3QQK) that confirmed that the compound binds ATP site through hydrogen bonding interactions between the thiazolamine moiety and the hinge region (GLU81-LEU83). The authors designed a set of analogues of the compound such that the hinge-binding functionality of the aminothiazole core remained unchanged while the flanking allyl and phenyl moieties were systematically modified. A total of 95 analogues were synthesized and evaluated using enzymatic assays and crystal structures of 35 CDK2 inhibitor complexes were determined between 1.4 and 2.0 A resolution.

As [9] have pointed out, hydrogen bonds are important interactions in the process of inhibition of CDK2. The ligand core (aminothiazole), for example, interacts with Glu81 and Leu83 in the structure 3QQK by hydrogen bonds. Thus, by using our approach we have found this core interactions as can be seen by the input graph of 3QQK. Moreover, one would expect that this type of interaction always appears in the frequent subgraphs. In fact, by analyzing the Figure 3 we can see that all the clusters have at least one hydrogen bond, except the cluster 8 formed for the PDB ids 3QRU and 3RPR. Although we cannot find any frequent subgraph in this cluster, by analyzing the input graphs of the two PDBs we can see that hydrogen bonds do occur. By comparing the hydrogen bond formed for 3QRU.A and 3RPR.A with its corresponding compound, we can see that the interactions are formed by different vertices type (donor-acceptor for 3QRU.A and acceptor-acceptor/donor for 3RPR.A), so that

none pattern can be found. It indicates that we need to refine our modeling to capture such pattern. As gSpan accepts just one label for nodes and edges, we need to consider using or even propose another FSM algorithm that allows multi-label.

In their work, they also mentioned several hydrophobic interactions being established between the protein residues and its ligands. In agreement with them our work shows that 13 to 15 clusters present hydrophobic interactions as frequent subgraphs. Despite [9] presents different clusters, they were not obtained by grouping similar molecules but they were compiled manually according to modifications done in molecules in order to high its inhibitory activity and, in fact, their clusters are not comparable to ours.

4. Conclusion and future work

In this work we propose a strategy based on frequent subgraph mining to bring out potential conserved patterns in protein-ligand interaction and molecular recognition.

We are on a exploratory phase of our work in which is important to validate our modeling and methodology as well as make some adjustments and refinements. For instance, we want to vary the distance criteria, explore other types of clustering algorithms and metrics to assess the quality of groups. We believe that it is important to test if our modeling can be improved by considering all interactions established by protein atoms which interacts with a ligand atom (even if such interactions are between protein atoms) because they can disrupt protein-ligand interaction. Also, we need to refine our modeling to capture patterns involving nodes and edges with more than one label, thus we need to consider using or even propose a FSM algorithm that allows multi-label.

Understanding and predicting protein-ligand interactions is a complex task and here we propose a model based on graphs to shed some light on why different small molecules are recognized by a specific protein. We believe our data analysis, summarized in our visualization tool, can give some valuable insights in understanding protein-ligand interactions and its conserved patters among clusters of similar proteins.

Acknowledgements

This work was supported by the Brazilian agencies Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG), Financiadora de Estudos e Projetos (FINEP) and Pró-Reitoria de Pesquisa da Universidade Federal de Minas Gerais.

References

[1] A. Kahraman, R. J. Morris, R. A. Laskowski, and J. M. Thornton, "Shape variation in protein binding pockets and their ligands," *Journal of molecular biology*, vol. 368, no. 1, pp. 283–301, 2007.

[2] D. E. Pires, R. C. de Melo-Minardi, C. H. da Silveira, F. F. Campos, and W. Meira, "acsm: noise-free graph-based signatures to large-scale receptor-based ligand prediction," *Bioinformatics*, vol. 29, no. 7, pp. 855–861, 2013.

[3] O. Khersonsky, C. Roodveldt, and D. S. Tawfik, "Enzyme promiscuity: evolutionary and mechanistic aspects," *Current opinion in chemical biology*, vol. 10, no. 5, pp. 498–508, 2006.

[4] K. Hult and P. Berglund, "Enzyme promiscuity: mechanism and applications," *Trends in biotechnology*, vol. 25, no. 5, pp. 231–238, 2007.

[5] O. K. Tawfik and D. S, "Enzyme promiscuity: a mechanistic and evolutionary perspective," *Annual review of biochemistry*, vol. 79, pp. 471–505, 2010.

[6] P. W. Rose, A. Prlić, C. Bi, *et al.*, "The rcsb protein data bank: views of structural biology for basic and applied research and education," *Nucleic acids research*, vol. 43, no. D1, pp. D345–D356, 2015.

[7] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of molecular biology*, vol. 48, no. 3, pp. 443–453, 1970.

[8] E. Rutenber *et al.*, "Crystallographic refinement of ricin to 2.5 å," *Proteins: Structure, Function, and Bioinformatics*, vol. 10, no. 3, pp. 240–250, 1991.

[9] E. Schonbrunn *et al.*, "Development of highly potent and selective diaminothiazole inhibitors of cyclin-dependent kinases," *Journal of medicinal chemistry*, vol. 56, no. 10, pp. 3768–3782, 2013.

[10] C. H. da Silveira, D. E. Pires, R. C. Minardi, *et al.*, "Protein cutoff scanning: A comparative analysis of cutoff dependent and cutoff free methods for prospecting contacts in proteins," *Proteins: Structure, Function, and Bioinformatics*, vol. 74, no. 3, pp. 727–743, 2009.

[11] "CGAL, Computational Geometry Algorithms Library," http://www.cgal.org.

[12] A. Poupon, "Voronoi and voronoi-related tessellations in studies of protein structure and interaction," *Current opinion in structural biology*, vol. 14, no. 2, pp. 233–241, 2004.

[13] A. Okabe *et al.*, *Spatial tessellations: concepts and applications of Voronoi diagrams*. Wiley, 2009, vol. 501.

[14] V. Sobolev and ohters, "Automated analysis of interatomic contacts in proteins," *Bioinformatics*, vol. 15, no. 4, pp. 327–332, 1999.

[15] L. Eldén, "Numerical linear algebra in data mining," *Acta Numerica*, vol. 15, pp. 327–384, 2006.

[16] S. Deerwester, S. Dumais, G. Furnas, *et al.*, "Computer information retrieval using latent semantic structure," June 13 1989, uS Patent 4,839,853.

[17] J. Han, M. Kamber, and J. Pei, *Data mining: Concepts and techniques*, 3rd ed. Morgan kaufmann, 2011.

[18] C. Hennig, *fpc: Flexible procedures for clustering*, 2014, r package version 2.1-9. [Online]. Available: http://CRAN.R-project.org/package=fpc

[19] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2013. [Online]. Available: http://www.R-project.org/

[20] S. A. Silveira, R. C. de Melo-Minardi, C. H. da Silveira, M. M. Santoro, and W. Meira Jr, "Enzymap: Exploiting protein annotation for modeling and predicting ec number changes in uniprot/swiss-prot," *PloS one*, vol. 9, no. 2, p. e89162, 2014.

[21] S. Shahrivari and S. Jalili, "High-performance parallel frequent subgraph discovery," *The Journal of Supercomputing*, pp. 1–21, 2015.

[22] R. Milo, S. Shen-Orr, S. Itzkovitz, *et al.*, "Network motifs: simple building blocks of complex networks," *Science*, vol. 298, no. 5594, pp. 824–827, 2002.

[23] X. Yan and J. Han, "gspan: Graph-based substructure pattern mining," in *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*. IEEE, 2002, pp. 721–724.

[24] F. Stahl, B. Gabrys, M. M. Gaber, and M. Berendsen, "An overview of interactive visual data mining techniques for knowledge discovery," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 3, no. 4, pp. 239–256, 2013.

[25] S. A. Silveira, A. O. Rodrigues, R. C. de Melo-Minardi, C. H. Silveira, and W. Meira Jr, "Advise: Visualizing the dynamics of enzyme annotations in uniprot/swiss-prot," in *Biological Data Visualization (BioVis), 2012 IEEE Symposium on*. IEEE, 2012, pp. 49–56.

[26] M.-C. Ho, M. B. Sturm, S. C. Almo, and V. L. Schramm, "Transition state analogues in structures of ricin and saporin ribosome-inactivating proteins," *Proceedings of the National Academy of Sciences*, vol. 106, no. 48, pp. 20 276–20 281, 2009.

SESSION

GENE EXPRESSION, REGULATORY NETWORKS, MICROARRAY, SEQUENCING, ALIGNMENT, AND RELATED STUDIES

Chair(s)

TBA

De Novo Assembly of Uca minax Transcriptome from Next Generation Sequencing

Hanin Omar[1], Casey A. Cole[1], Arjang Fahim[1], Giuliana Gusmaroli[2], Stephen Borgianini[2], and Homayoun Valafar[1*]

[1]Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208, USA

[2]Department of Natural Sciences, University of South Carolina, Beaufort, SC 29901, USA

[*] Corresponding Author Email: homayoun@cec.sc.edu Phone: 1 803 777 2404 Fax: 1 803 777 3767

Mailing Address: Swearingen Engineering Center, Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208, USA

Abstract – *High-throughput cDNA sequencing (RNA-seq) is a very powerful technique to quantify gene expression in an unbiased way. The Crustacean family is among the groups of organisms sparsely represented in current genomic databases. Here we present transcriptome data from Uca minax (red-jointed fiddler crab) as an opportunity to extend our knowledge. Next generation sequencing was performed on six tissue samples from Uca minax using the Illumina HiSeq system. Six Transcriptome libraries were created using Trinity; a free, open-source software tool for de novo transcriptome assembly of high-throughput mRNA sequencing (RNA-seq) data with the absence of a reference genome. In addition, several tools that aid in management of data were used, such as RSEM, Bowtie, Blast, and IGV; a tool for visualizing RNA-seq analysis results. Fast quality control (FastQC) analysis of the raw sequenced files revealed that both adapter and PCR primer sequences were prevalently present, which may require a preprocessing step.*

Keywords: Uca minax, Trinity, Transcriptome, assembly, Next Generation Sequencing. .

1 Introduction

The advent of Structural Genomics era, marked by completion of the Human Genome project in 2003[1], [2] has introduced many exciting avenues of research in molecular biology and understanding of diseases. Assisted by revolutionizing technologies such as Polymerase Chain Reaction (PCR)[3], and advances is instrumentation[4][5], genome sequencing continues to become a more routine operation compared to other areas of molecular sciences such as protein structure determination[6][7], [8]. While the initial cost of human genome sequencing consisted of $\$10^9$, the new technologies afford a reduced cost of $\$10^3$ for sequencing of the same genome. In addition, from the temporal perspective, the initial human genome project required 10 years of data acquisition and three years of data analysis. In contrast, the use of Next Generation Sequencing (NGS)[9], [10] technology has reduced this time requirement to less than a week for combined data acquisition and analysis.

These significant advances in financial and temporal cost of gene sequencing is rooted in development of massively parallel instance of sequencing shorter sequences (100-10,000 bp). While the new parallel approaches increase the overall throughput by several orders of magnitude, they impose the challenge of gene assembly from short reads. Therefore, in recent years one visible area of research and development is focused on evaluation and development of new assembly techniques.

Computational approaches to assembly of NGS sequences can be place in two distinct categories: "Mapping first", and De novo assembly. The former approach relies on existence of a sequenced genome of an organism closely related to the organism under examination. In this approach the existing genome is used as a template to map, following the assembly of the NGS reads. The former approach aims to assemble the short sequences first without any other a-priori knowledge. Programs such as AbySS[11], SOAPdenovo[12], Oases[13] or Trinity[14] can be cited as De novo approaches to sequence assembly. Programs such as Scripture[15] or Cufflinks[16] can be cited as this category of tools.

In this research we utilized the Trinity software because of its availability, popularity, performance, and hardware requirement in order to assemble the transcriptome of the organism Uca minax. Our selection of Uca minax is based on a number of its unique biological properties including adaptability to a broad spectrum of salinity in its environment. Furthermore, there has been a very poor sampling for the genome of crustaceans and therefore assembled transcriptome of Uca minax will help to close this gap in genetic information. In this report we present the assembly results of transcriptome from six tissues of Uca minax using the software package Trinity.

2 Materials and Methods

2.1 Next generation sequencing data

Messenger RNA (mRNA) was extracted from the following six tissues of Uca minax: anterior gills, posterior gills, gonads (male and female), eye stalk and muscle, first and second Zoea stage, and third Zoea stage using oligodT

primer. The mRNA samples were then fragmented and reverse transcribed using random primers. Next, they are double stranded and ligated with adapters according to the Illumina (http://www.illumina.com) prep library protocol. This process yielded six standard cDNA libraries (one for each tissue) for sequence analysis via Illumina HiSeq™ 2000 (commercial service) at the Genomics LAB of the David H. Murdock Research Institute (DHMRI) in North Carolina (www.http://dhmri.org/). These libraries were then combined together to generate a single library pool for the sequencing exercise. This pool was loaded into one lane of the flow cell. Clusters were generated for a paired end read flow cell and a 100 Bp strand specific paired end read sequencing was performed. This process yielded a total of twelve fastq files; two for each library. Table 1 displays the number of raw reads and corresponding size of each sequenced tissue.

Table 1: Uca minax raw reads sequenced by NGS

Tissue Name	File type	Raw Reads count	Size (GB)
Anterior Gills	R1	7360115	1.9
	R2	7360115	1.9
Posterior Gills	R1	10870260	2.8
	R2	10870260	2.8
Female + Male Gonads	R1	8308628	2.1
	R2	8308628	2.1
Eye Stalk + Muscle	R1	9761180	2.5
	R2	9761180	2.5
1st + 2st Zoea Stage	R1	8481444	2.2
	R2	8481444	2.2
3rd Zoea Stage	R1	9973433	2.6
	R2	9973433	2.6

Image processing and base calling steps were performed at DHMRI to generate the following summary report of sequencing data quality control:

1. Q score: > 80% of bases had a quality score (Q) > 30. The quality or Q_{score} is defined in Eq. (1) and measures the probability that a base is called incorrectly. A Q score of 30 reflects the probability of an incorrect base call of 1 in 1000 for an inferred base call accuracy rate of 99.9%.

$$Q = -10\log_{10}(e) \qquad \text{Eq. (1)}$$

2. Data throughput: The data throughput quality control (QC) threshold was set to be larger than 100 million reads/lane.

2.2 De novo sequence assembly

The Trinity[14][17] software package (release 2013-2-25) optimized with k-mer[18] length of 25 for performing de novo assembly on the raw reads was used in this work. The computational work was performed on a plank cluster with 864 cores, and each node populated with 24 GB of RAM memory. The operating system of this computational facility was CentOS (https://www.centos.org/). The command-line arguments used with Trinity are shown in Dialogue 1. Each of the parameters is briefly described in Table 2.

```
---left "compatible_path_extension_for_reverse_reads"
--right "compatible_path_extension_for _forward_reads"
--seqType fq --SS_lib_type RF --JM 20 --CPU 12 –Output
"compatible_path_extension_for _output_folder".
```

Dialogue 1. Command line arguments used during assembly session with Trinity.

Table 2. Arguments used with Trinity and a brief description of each.

Argument	Brief Description
---left	Input file name for left reads
--right	Input file name for right reads
--seqType	Type of input files; fastq or fasta
--SS_lib_type	Define the left and right files read orientation, RF or FR
--JM	The memory assigned for the kmer dictionary in GB
--CPU	Number of CPU assigned for Trinity to use
-Output	Name of the output folder that contains the assembled transcriptome file

Due to the ambiguity of the protocol used in the preparation of the strand-specific libraries for sequencing, we ran Trinity twice for each tissue. In the first run the –SS_lib_type parameter was set to FR, and on the second run it was set to RF. To explore the performance of Trinity on these datasets we varied the number of cores being utilized by the process and recorded the run time. Table 3 shows the effect of this variation of cores assigned to Trinity on the overall running time performance.

Table 3: Trinity run time analysis

Tissue name	Numbers of cores	Total running time (hours)
Anterior gills	6	< 7
	12	< 3
Posterior gills	6	< 5:30
	12	< 3
Female + male gonads	6	< 8
	12	< 3
Eye stalk + muscle	6	< 26
	12	< 19
1st + 2nd Zoea stage	6	< 11
	12	< 2
3rd Zoea stage	6	< 12
	12	< 6

Bowtie[19] aligner (version 0.12.9) was used to map back

the raw short paired reads to the assembled transcripts produced by Trinity. The command-line parameters used with the *alignReads* script are shown in Dialogue 2. Table 4 provides a brief description of each parameter.

```
---left "compatible_path_extension_for_reverse_reads" --right "compatible_path_extension_for _forward_reads" --seqType fq --SS_lib_type RF --aligner bowtie –retain_intermediate_files –target "compatible_path_extension_for trinity.fasta file" –Output "compatible_path_extension_for _output_folder".
```

Dialogue 2. Command line arguments used with alignReads.pl script.

Table 4. Arguments used with alignReads and a brief description of each.

Argument	Brief Description
--aligner	The choice of aligner used; either bowtie or bowtie2
-target	Path to the desired Trinity assembled transcriptome Fasta file

Next , the Integrated Genomic Viewer (IGV)[20][21] was used to visualize the *Bam* alignment files generated by Bowtie and obtain assembly statistics for the raw reads that were able to be mapped back to one or more of the assembled transcriptomes in each tissue.

RSEM[22](RNA-Seq by Expectation –Maximization) v1.2.20 was used to estimate the gene and isoform expression levels in the assembled transcriptome files generated by Trinity.

2.3 Database and web server software

The original parsing of raw data was done using the Perl scripting language (version 5.12.4). To house our data, we used the MySQL (version 5.1.68) data-warehousing tool.

The project website utilizes a combination of PHP scripting (version 5.3.10), JavaScript (version 1.7.1), and HTML5 and it is powered by Kubuntu 14.04 operating system.

2.4 Blast database of sequences

We created twelve DNA Blast[23] databases for each of the raw (short reads) files that were produced by NGS sequencing. Furthermore, we created twelve DNA Blast databases for the assembled transcriptome files generated by Trinity (note: two runs were performed for each one of the six tissue, each one with a different library type FR/RF). The DNA Blast databases are created using the command-line *makeblastdb* available as part of the blast+[24] package , with the command line parameters shown in Dialogue 3. Table 5 provides a brief description of each parameter.

```
-in "compatible_path_extension_for_input_file", -dbtype nucl, -out "compatible_path_extension_for_blast_database"
```

Dialogue 3. Command line arguments used with makeblastdb.

Table 5. Arguments used with Trinity and a brief description of each.

Argument	Brief Description
-in	Input file name, the file in fasta format
-dbtype	Type of database; nucl for DNA database
-out	Name of blast database created

2.5 Evaluation methods

With the absence of a reference genome, as in the case with non-model organism such as Uca minax, the process of evaluating the correctness and quality of the assembled transcriptomes via de novo sequence assembly methods becomes somewhat ambiguous. There is no definite criteria that can clearly draw a line that separates correct vs incorrect assembled transcriptomes. Hence, the decision was made to use house keeping genes as the criteria to judge the correctness of the assembled transcriptome. We hypothesize that if Trinity's assembly is indeed accurate then a blast search of the transcript database using these house keeping genes should yield alignments across all reconstructed tissues.

The house keeping genes of choice were Histone H3 and Ribosomal protein S16. Histone H3 is one of the five Histone proteins in eukaryotic cells. These proteins are the main components of chromatin which are responsible for packaging and ordering DNA into nucleosomes, as well as having a role in gene regulation. Histone proteins are among the most highly conserved proteins in eukaryotes. Ribosomal protein S16 as its name indicates is one of the proteins that, along with rRNA is responsible for building ribosomal units. S16 is the main protein used for reconstructing phylogenies due to it being highly conserved between different species.

For validation purposes, we used partial sequences of Uca minax Histone H3 and Ribosomal protein S16. These sequences are shown in Table 6. The partial sequences were cloned through the use of RT-PCR with degenerated primers (oligoas). However, considering the absence of a reference genome as well as the fact that the N-term and C-term of the corresponding proteins are rarely conserved, only partial sequences of the corresponding proteins could be retrieved from the Uca minax. Hence, two complete homologous sequences of evolutionary related organisms such as Lice (Pediculus humanus corporis) Histone H3 and Water flea Ribosomal S16 (Daphnia pules S16) were used to retrieve corresponding genes from the Uca minax transcriptome database generated by Trinity.

Table 6. Sequence of Histone H3 and Ribosomal protein S16 genes.

Gene name	Sequence
Histone H3	ATCTGCTCTGCTACCGGAGGAGTCAAGAAGCC CCACCGTTACAGGCCAGGCATCGCCGCACTGC GTGAAATCCGCCGCTACCAGAAGAGCACCGAG CTGCTCATCAGGAAGCTGCCTTTCCAGCGTCT GGTGCGCGAGATCGCCCAGGATTTCAAGACCG ATCTCCGCTTCCAGTCCTCTGCTGTCATGGCT CTCCAGGAGGCCTCAGAGGCTTACCTCGTCGG TCTCTTCGAGGACACCAACCTGTGCGATTTCC ACGCCATAGGGGGGGGAGTATAATAAAAGAGT GGGGTACGTTCACGCCGATTTAAGAAGATAGT GCAAAACGACTGCATAGGTATCCTGCTGTTTG AAGATCACACTCCAGTCTGTTACGCCACTCTT TATAAGACTAGTGGTTTTTGGGCCCGGCA
Ribosomal S16 subunit	TTGAGCCCAGGACACTGCAGTTCAAGTT GATGGAGCCTGTGTCGCTGCTGGGCAAG GAGAGGTTTTCCAATGTGTCCATCCGTG TGCGTGTGAAGGGTGGCGGACACACCTC CCAGGTCTATGCCATCCGTCAGGCCATC TCCAAGTCCCTCGTGGCTTACTACCAGA AGTTTGTGGACGAGGCCTCCAAGAAGG AGATCAAGAACATCCTTATCAACTATGA CAGGTCACTCTTGGTCGCTGACCCCAGG CGGTGTGAGCCCAAGAAGTTCGGAGGTC CTGGAGCCAGGGCACGCTACCAGAA

3 Results and Discussion

3.1 Assembled transcriptome

In total, we created twelve Uca minax transcriptome libraries (six pairs). Each pair corresponds to the same tissue but with different assembly orientation (FR vs RF). As seen in Table 7, the number of transcripts assembled by Trinity for the same tissue slightly differs depending on the orientation. However, we noticed that for almost 80% of the transcripts Trinity assembles a sequence in one orientation (for example FR) and then assembles its reverse complement in the other orientation (RF).This signifies that the distinction of the direction of the raw reads might prove insignificant in our case. Moreover, Table 7 provides the basic statistics on the number of genes, isoforms and contigs assembled by Trinity for each run.

Further analysis of the assembled transcriptomes was needed to determine which transcripts were isoforms of the same gene, for that we used RSEM software. The results of the RSEM analysis are shown in Tables 8 and 9. The focus of the analysis was on two relative measure of transcript abundance: the Transcripts per million (TPM) and the Fragments per kilobase of exon per million reads mapped (FPKM) values. TPM indicates the number of transcripts of a specific type found if one million full transcripts from the sample are sequenced, given the abundances of the other transcripts in the sample. FPKM is the expected number of fragments to be found for each thousand bases in the feature for every $N/10^6$ sequenced fragment if the same RNA pool was sequenced again. The results of the gene quantification analysis for the Uca minax transcriptomes are shown in Table 8. This table enumerates the number of genes as well as their percentage in each library of assembled transcriptome that have FPKM, TPM and expected count values equal zero. On the other hand, Table 9 contains the number of assembled transcripts as well as their percentage in each library of the assembled transcriptome that have FPKM, TPM, expected count and IsoPCT (which stands for the percentage of this transcript's abundance over its parent gene's abundance) values of zero.

Table 7. Basic Trinity statistics

Library name	Read Orientation	Total Trinity Transcripts	Trinity Components	Contig N50
Anterior gills	FR	108674	87591	779
	RF	108651	87569	770
Posterior gills	FR	117370	90888	876
	RF	117173	90832	876
Female + male gonads	FR	118288	93389	708
	RF	118397	93498	695
Eye stalk + muscle	FR	169817	156547	292
	RF	168591	155342	291
1st + 2nd Zoea stage	FR	152489	120625	717
	RF	152197	120744	716
3rd Zoea stage	FR	165495	119072	1088
	RF	165895	119081	1081

Table 8. Uca minax RSEM analysis (genes)

Library name	Read orientation	FPKM =0
Anterior gills	FR	22301 (26%)
	RF	1424 (2%)
Posterior gills	FR	21394 (24%)
	RF	21739 (25%)
Female + male gonads	FR	33608 (38%)
	RF	33509 (38%)
Eye stalk + muscle	FR	58417 (67%)
	RF	58254 (67%)
1st + 2nd Zoea stage	FR	31754 (36%)
	RF	31885 (36%)
3rd Zoea stage	FR	24802 (28%)
	RF	24979 (28%)

Table 9. Uca minax RSEM analysis (isoforms)

Library name	Read orientation	FPKM =0
Anterior gills	FR	30581 (28%)
	RF	7345 (7%)
Posterior gills	FR	29896 (28%)
	RF	30120 (28%)
Female + male gonads	FR	45051 (41%)
	RF	44795 (41%)
Eye stalk + muscle	FR	74543 (67%)
	RF	74339 (67%)
1st + 2nd Zoea stage	FR	42744 (40%)
	RF	42881 (39%)
3rd Zoea stage	FR	34862 (32%)
	RF	35024 (32%)

3.2 Public web resources

All resources and tools are publicly available via our website (www.rdc.cse.sc.edu/Uca_minax/TransNav). This section will highlight the functionality of each tab accessible from the main menu.

Blast—We provide an easy-to-use interface for Blast search conducted on a comprehensive set of databases. A Blast search can be performed on both the raw data and the reconstructed transcriptome data obtained via Trinity. The search tool allows for the user to either upload a file containing their query sequences (in FASTA format) or simply copy and paste their sequences into the provided text box. To perform searches on multiple data sets at once, the user can simply check the boxes beside each desired database. Once the submit button is clicked our engine will perform a Blast search on each database sequentially and display the alignment results for each database below the search tool in separate windows. The user can then choose to download a txt or html version of the results to save for future reference, see Figure 1.

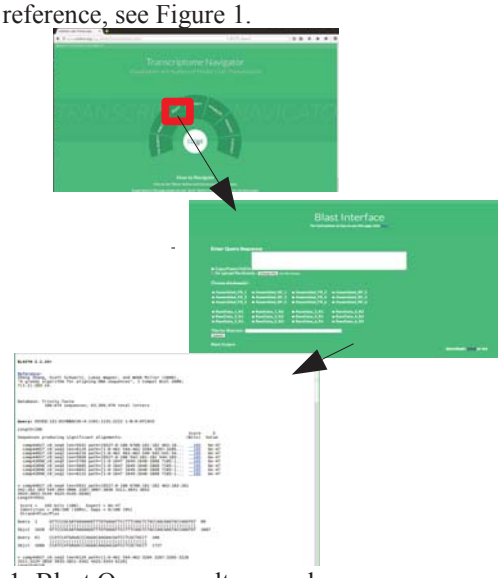


Figure 1: Blast Query result example

Visualize—To evaluate the quality of the reconstructed transcripts and assess results of the Trinity software package, we use Bowtie to align the resulting transcriptome to the original 100 bp reads. IGV (Integrated Genome Viewer)[5][6] is used to view the subsequent alignments. In doing this, we can get an idea of how well Trinity has conserved the original reads in its reconstruction as well as establish a confidence in the reconstructed transcriptome based on the observed coverage. The user can select any of the links on the "Visualization Tool" screen to view the alignment. Doing so will open IGV on their local machine and automatically load the Trinity results along with its alignment to the original reads, see Figure 2. The IGV package can be downloaded from the Broad Institute's website at http://broadinstitute.org/igv/home.

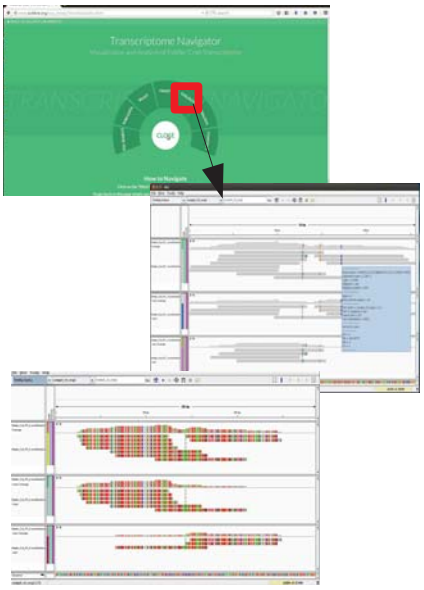


Figure 2: IGV alignment example

Trinity—We provide links for direct download of all Trinity results. On the "Trinity" screen each tissue has two download options. Clicking on the "FASTA" link will download the Trinity assembled transcriptome library for that tissue in FASTA format. Selecting the "Blast searchable database" option will download the data sets in a convenient Blast searchable format. After clicking the links, the files will be downloaded and saved to the user's local machine.

Analysis—On the "Analysis" page of the website the user can find gene and isoform abundance reports generated from the Trinity results for each individual tissue. These files are available for download and are in a spreadsheet-friendly, tab delimited format. The abundance reports were generated using the software package RSEM[7] .

Raw Data—We have also made all of our raw sequenced data available for download. The raw data sets can be downloaded in three forms: FASTA, FASTQ and Blast searchable databases. Each format can be accessed by selecting their respective links under each tissue.

SQL Search—The "SQL Search" tab leads to a page that directs the user to the phpMyAdmin

(http://www.phpmyadmin.net/) view of our databases. Select collaborators have been granted access to perform their own SQL queries on our data. Both the original database and compressed database are available to explore.

Contact—All contact information for all collaborators and labs are available on this page.

3.3 Validation of house keeping genes

Blastx[23] was used to find a matching transcriptome for both the partial sequences and the complete ones. In the Histone H3 case, only two transcriptome libraries produced a match. As shown in Table 10, the two matching transcriptomes were found in both the female and male gonads and the eye stalk and muscle libraries. All matching transcriptomes from the female and male gonad libraries (in both read directions) matched both the complete and partial sequence of the Histone H3. On the other hand, the results from the Eye stalk and muscle libraries varied from one read direction to the other. In the forward/reverse (FR) direction, four transcriptomes were matched but only three of them matched both the complete and partial Histone H3 sequences. The last transcriptome matched only with the partial Histone H3 sequence. In the reverse/forward (RF) direction, three matches were found. Only two matched both the complete and partial H3 sequences while the last transcriptome matched the partial H3 sequence.

In the case of Ribosomal S16 protein, two matches were found in each one of the six libraries for the partial S16 sequence, however none of the assembled transcriptomes in all six libraries matched the complete Water flea sequence, for more details see Table 11. In both Table 10 and 11;the first column identifies the transcriptome library name, the second column specifies the assembly orientation, the third column lists the transcriptome id assigned by Trinity during the assembly process, the fourth column contains the Bp length of the assembled transcriptome, the fifth column is the scoring assigned by Blast, the sixth column is the percentage of matching identities between the transcript and the query sequence.

Table 10: Transcriptome match results for the Histone H3, where RD is the Read Direction and Trans. ID is the Transcriptome ID

Library	RD	Trans. ID	Len.	Score	Identities
Female + male Gonads	FR	comp106449_c0_seq1	227	(Complete) 71	129/158 (82%)
				(Uca) 82	83/84 (99%)
		comp144323_c0_seq1	398	(Complete) 56	92/110 (84%)
				(Uca) 81	83/84 (99%)
	RF	comp107689_c0_seq1	227	(Complete) 71	129/158 (82%)
				(Uca) 116	118/119 (99%)
		comp133719_c0_seq1	398	(Complete) 56	92/110 (84%)
				(Uca) 81	83/84 (99%)
Eye stalk + muscle	FR	comp107816_c0_seq1	384	(Complete) 86	172/215 (80%)
				(Uca) 166	168/169 (99%)
		comp111814_c1_seq2	448	(Complete) 77	146/180 (81%)
				(Uca) 141	143/144 (99%)
		comp111814_c1_seq1	512	(Complete) 77	146/180 (81%)
				(Uca) 141	143/144 (99%)
		comp68227_c0_seq1	257	(Uca) 109	118/122 (97%)
	RF	comp106949_c0_seq1	484	(Complete) 86	172/215 (80%)
				(Uca) 166	168/169 (99%)
		comp112248_c1_seq1	472	(Complete) 77	146/180 (81%)
				(Uca) 141	143/144 (99%)
		comp71139_c0_seq1	258	(Uca) 109	118/122 (97%)

3.4 Analysis of data quality

As a result of the house keeping gene validation step, it was important to investigate the quality of the original raw sequenced data to produce a suitable per-processing template to further improve results of the de novo assembly step. The quality control analysis of the raw sequenced data was performed using FastQC[25] (v0.11.2) software package. All raw sequenced data files used Sanger/Illumina 1.9 encoding. Table 12 contains some of the basic statistics provided by FastQC analysis of the original raw sequenced data. The percentage of nitrogenous bases on a DNA/RNA molecule that are either guanine or cytosine is known as GC content. Determination of this ratio contributes in mapping gene-rich regions of the genome. Overall GC content of all the bases of all sequences and total sequences are two of the parameters that we report in this table. Both of these parameters are the same for the forward read and the reverse reads of the same tissue.

Table 11:Transcriptome match results for the Ribosomal S16

Library	RD	Trans. ID	Len.	Score	Identities
Anterior gills	FR	comp22755_c0_seq1	548	81	301/304 (99%)
		comp22753_c0_seq1	544	295	301/304 (99%)
	RF	comp38036_c1_seq1	547	295	301/304 (99%)
		comp38036_c0_seq1	521	295	301/304 (99%)
Posterior gills	FR	comp25671_c0_seq1	548	295	301/304 (99%)
		comp25635_c0_seq1	550	295	301/304 (99%)
	RF	comp42515_c1_seq1	548	295	301/304 (99%)
		comp42515_c0_seq1	523	295	301/304 (99%)
Female + male gonads	FR	comp41201_c0_seq1	545	298	301/304 (99%)
		comp45019_c0_seq1	545	295	301/304 (99%)
	RF	comp49683_c0_seq1	522	298	302/304 (99%)
		comp49683_c1_seq1	544	295	301/304 (99%)
Eye stalk + muscle	FR	comp85313_c0_seq1	503	295	301/304 (99%)
		comp81645_c0_seq1	459	295	301/304 (99%)
	RF	comp89360_c0_seq2	526	298	302/304 (99%)
		comp89360_c0_seq1	461	298	302/304 (99%)
1st + 2nd Zoea stage	FR	comp52715_c0_seq1	552	295	301/304 (99%)
		comp68619_c1_seq1	522	292	300/304 (99%)
	RF	comp63400_c0_seq1	519	295	301/304 (99%)
		comp63400_c1_seq1	528	292	300/304 (99%)
3rd Zoea stage	FR	comp67340_c1_seq1	542	295	301/304 (99%)
		comp67340_c0_seq1	513	295	301/304 (99%)
	RF	comp54913_c0_seq1	545	295	301/304 (99%)
		comp54911_c0_seq1	543	295	301/304 (99%)

Table 12: FastQC Basic Statistics

Tissue	GC%	Total sequences
Anterior gills	46	7360115
Posterior gills	44	10870260
Female + male gonads	41	8308628
Eye stalk + muscle	39	9761180
1st + 2nd Zoea stage	43	8481444
3rd Zoea stage	47	9973433

Analysis for the original raw sequenced data revealed the same patterns for five of the six tissues sequenced (anterior gills, posterior gills,, female+male gonads, 1st + 2nd zoea stage and 3rd zoea stage). These patterns are listed below.

1. Good scoring results for the following modules: per base sequence quality, per sequence quality score, per base N content, sequence length distribution and adapter content.
2. Warnings were issued for sequence duplication levels, overrepresented sequences modules and some files issued warnings for the per tile sequence quality module.
3. Failed scoring results for the following modules: per base sequence content, per sequence GC content and kmer content.

On the other hand, the eye stalk + muscle tissue deviated from this pattern by having good scoring results for four modules only: per sequence quality score, per base N content and sequence length distribution Warnings were issued for sequence duplication levels, per sequence GC content modules and the R1 file issued warnings for the per tile sequence quality module. All of the remaining modules failed the scoring result.

For the analysis of the five tissues with the same pattern, it showed that the libraries were biased in the first 15bp of the raw sequences and that some contamination of the libraries occurred mainly because of adapter sequences and PCR primer sequences which are normally part of next generation sequencing protocol, existed as part of the raw sequences. This can be easily solved by trimming both the adapter and PCR primer sequences and removing any primer-dimer in the raw libraries, also some sequence truncation may be performed later on. However, for the eye stalk and muscle tissue case, it seems that the adapter contamination is more severe and although it can be lessened by trimming, the resulting raw sequences after trimming may not provide enough coverage for a good transcripts assembly result.

4 Conclusion

Our initial investigation of the Uca minax transcriptome assembly indicates successful reconstruction for five of the six tissue with some potential room for improvement. The five well behaved tissues include: anterior gills, posterior gills,, female+male gonads, 1st + 2nd zoea stage and 3rd zoea stage. Data from these tissue samples can be further improved for a second round of assembly by removal of the primer and adapter sequences. This process will require a preprocessing step where a-priori knowledge of these sequences is used to remove redundant primer/adapter sequences. Upon cleansing of the raw data, we speculate that transcriptome assembly will result in better reconstructed genetic data. The same process will be repeated for the "Eye stalk + muscle" that exhibits the most degree of contamination. We speculate that preprocessing of the data will improve the quality of the final assembled transcriptome, but will likely contain less number and shorter genes. We will publish our final assembled genome at our current website.

5 Acknowledgements

This work was supported by NIH Grant Number P20 RR-016461 to Dr. Homayoun Valafar and INBRE Pilot Project grant to Dr. Giuliana Gusmaroli.

6 Bibliography

[1] M. P. Sawicki, G. Samara, M. Hurwitz, and E. Passaro, "Human Genome Project," *Am. J. Surg.*, vol. 165, no. 2, pp. 258–264, Feb. 1993.

[2] F. S. Collins, M. Morgan, and A. Patrinos, "The Human Genome Project: lessons from large-scale biology.," *Science*, vol. 300, no. 5617, pp. 286–90, Apr. 2003.

[3] H. A. Erlich, "Polymerase chain reaction," *J. Clin. Immunol.*, vol. 9, no. 6, pp. 437–447, Nov. 1989.

[4] M. L. Metzker, "Sequencing technologies - the next generation.," *Nat. Rev. Genet.*, vol. 11, no. 1, pp. 31–46, Jan. 2010.

[5] H. P. J. Buermans and J. T. den Dunnen, "Next generation sequencing technology: Advances and applications.," *Biochim. Biophys. Acta*, vol. 1842, no. 10, pp. 1932–1941, Jul. 2014.

[6] H. M. Berman, J. D. Westbrook, M. J. Gabanyi, W. Tao, R. Shah, A. Kouranov, T. Schwede, K. Arnold, F. Kiefer, L. Bordoli, J. Kopp, M. Podvinec, P. D. Adams, L. G. Carter, W. Minor, R. Nair, and J. La Baer, "The protein structure initiative structural genomics knowledgebase.," *Nucleic Acids Res.*, vol. 37, no. Database issue, pp. D365–8, Jan. 2009.

[7] R. F. Service, "Structural genomics - Tapping DNA for structures produces a trickle," vol. 298, no. 5595, pp. 948–950, 2002.

[8] R. Service, "Structural biology - Structural genomics, round 2," *Science (80-.).*, vol. 307, no. 5715, pp. 1554–1558, 2005.

[9] S. C. Schuster, "Next-generation sequencing transforms today's biology.," *Nat. Methods*, vol. 5, no. 1, pp. 16–8, Jan. 2008.

[10] W. J. Ansorge, "Next-generation DNA sequencing techniques.," *N. Biotechnol.*, vol. 25, no. 4, pp. 195–203, Apr. 2009.

[11] I. Birol, S. D. Jackman, C. B. Nielsen, J. Q. Qian, R. Varhol, G. Stazyk, R. D. Morin, Y. Zhao, M. Hirst, J. E. Schein, D. E. Horsman, J. M. Connors, R. D. Gascoyne, M. A. Marra, and S. J. M. Jones, "De novo transcriptome assembly with ABySS," *Bioinformatics*, vol. 25, no. 21, pp. 2872–2877, 2009.

[12] R. Luo, B. Liu, Y. Xie, Z. Li, W. Huang, J. Yuan, G. He, Y. Chen, Q. Pan, Y. Liu, J. Tang, G. Wu, H. Zhang, Y. Shi, Y. Liu, C. Yu, B. Wang, Y. Lu, C. Han, D. W. Cheung, S.-M. Yiu, S. Peng, Z. Xiaoqian, G. Liu, X. Liao, Y. Li, H. Yang, J. Wang, T.-W. Lam, and J. Wang, "SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler.," *Gigascience*, vol. 1, no. 1, p. 18, Jan. 2012.

[13] M. H. Schulz, D. R. Zerbino, M. Vingron, and E. Birney, "Oases: robust de novo RNA-seq assembly across the dynamic range of expression levels.," *Bioinformatics*, vol. 28, no. 8, pp. 1086–92, Apr. 2012.

[14] B. J. Haas, A. Papanicolaou, M. Yassour, M. Grabherr, D. Philip, J. Bowden, M. B. Couger, D. Eccles, B. Li, M. D. Macmanes, M. Ott, J. Orvis, and N. Pochet, *reference generation and analysis with Trinity*, vol. 8, no. 8. 2014.

[15] G. Chen, C. Wang, and T. Shi, "Overview of available methods for diverse RNA-Seq data analyses.," *Sci. China. Life Sci.*, vol. 54, no. 12, pp. 1121–8, Dec. 2011.

[16] C. Trapnell, B. A. Williams, G. Pertea, A. Mortazavi, G. Kwan, M. J. van Baren, S. L. Salzberg, B. J. Wold, and L. Pachter, "Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation.," *Nat. Biotechnol.*, vol. 28, no. 5, pp. 511–515, 2010.

[17] I. Amit, X. Adiconis, L. Fan, R. Raychowdhury, Z. Chen, E. Mauceli, N. Hacohen, A. Gnirke, N. Rhind, F. Palma, B. W. Birren, C. Nusbaum, and K. Lindblad-toh, "genome from RNA-Seq data," vol. 29, no. 7, pp. 644–652, 2013.

[18] G. Marçais and C. Kingsford, "A fast, lock-free approach for efficient parallel counting of occurrences of k-mers.," *Bioinformatics*, vol. 27, no. 6, pp. 764–70, Mar. 2011.

[19] B. Langmead, C. Trapnell, M. Pop, and S. L. Salzberg, "Ultrafast and memory-efficient alignment of short DNA sequences to the human genome.," *Genome Biol.*, vol. 10, no. 3, p. R25, Jan. 2009.

[20] H. Thorvaldsdóttir, J. T. Robinson, and J. P. Mesirov, "Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration.," *Brief. Bioinform.*, vol. 14, no. 2, pp. 178–92, Mar. 2013.

[21] C. F. Interests, "Integrative genomics viewer," vol. 29, no. 1, pp. 24–26, 2011.

[22] B. Li and C. N. Dewey, "RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome.," *BMC Bioinformatics*, vol. 12, p. 323, Jan. 2011.

[23] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *Journal of Molecular Biology*, vol. 215. pp. 403–410, 1990.

[24] C. Camacho, G. Coulouris, V. Avagyan, N. Ma, J. Papadopoulos, K. Bealer, and T. L. Madden, "BLAST+: architecture and applications.," *BMC Bioinformatics*, vol. 10, p. 421, Jan. 2009.

[25] S. Andrews, "FastQC: A quality control tool for high throughput sequence data.," *http://www.bioinformatics.babraham.ac.uk/projects/fastqc/*, 2010.

A Bitwise Encoding Scheme Designed to Improve the Speed of Large Scale Gene Set Comparison

Tham H. Hoang, Pujan Joshi, Seung-Hyun Hong and Dong-Guk Shin
Department of Computer Science and Engineering, University of Connecticut, Storrs, CT, USA

Abstract— *Comparing one gene set against a large number of gene sets is a method that is ubiquitously used in many high-throughput gene expression studies. A conventional way of comparing one gene set against a large number of target gene sets is through database programming. However, this approach can be inefficient if the sizes of the target gene sets become massively large. We propose a coding scheme designed to convert both query and target gene sets into bit streams and use bitwise Boolean operations such as intersections and set differences for comparisons. Our approach can efficiently represent gene sets containing a small number of gene identifiers, thus suggesting the feasibility of storing a large amount of target gene sets into memory entirely. Our method drastically reduces disk I/Os which are inevitable in the database programming approach. Our method can offer a constant time performance independent of the size of the target gene sets.*

Keywords: Enrichment analysis, bitwise operators, query system, encoding method, gene set interactions.

1. Introduction

The current biomedical research community faces a significant challenge in handling the genomics data that is inundating the field. Particularly, the recent advance in Next Generation Sequencing (NGS) technology has been dramatically changing the way biomedical scientists analyze experimental data. One noticeable trend has been: (i) a group of closely collaborating scientists produces massive amounts of data with a specific agenda in mind, (ii) they deposit the collectively obtained data sets in a public repository, and (iii) the public uses them as references for various types of research inspired by an individual laboratory's interest. There are multiple examples of such community generated genomics data projects. The 1000 Genomes project aims at uncovering structural variations using whole genome sequencing on samples taken from over 2500 individuals with known geographic and ethnic origins [1]. TCGA is to explore the entire spectrum of genomic changes involved multiple human cancer types [2]. ENCODE aims at identifying key functional elements in the human genome [3]. In addition there is a Gene Expression Omnibus (GEO) which is a "catch-all" database repository of many different types of high throughput gene expression data which were submitted by thousands of scientists [4]. All these public data sets wait for utilization by bioinformaticians or computational biologists at large.

We have been analyzing numerous high-throughput gene expression data in attempts to help many local biomedical scientists derive meaningful interpretations from their respective experiments. Steps include, for example, subgrouping gene expression patterns using cluster methods like a Pattern-Based clustering (PBC) program [5] and then performing the subsequent "meta-analyses" [6]. Typically two types of meta-analysis can be attempted: (i) enrichment analysis (e.g., WebGestalt [7]) and (ii) comparing the locally generated data with the similar publicly available or related ones. Both approaches involve gene set comparisons. Enrichment analysis includes comparing a "query gene set" (e.g., differentially expressing "DE" genes) against the "target gene sets" such as well-curated collections of gene sets in Gene Ontology or pathway databases (e.g., KEGG). The goal is to find whether or not query set includes an unexpectedly large number of genes of some a priori known functional group(s). For example, biologists can discover if the identified DE genes clearly point (towards) that "Wnt signaling pathway" is suppressed or whether they are more or less involved in the biological process known as "osteogenesis" and so on. Typically Fisher test is used to produce a p-value which measures the statistical significance of the enrichment. Unlike the first approach, the second approach comparing the DE gene set against the data set available from the public repository is not well-established. In fact, how to carry out the second type of analysis is an active area of bioinformatics research; there are many alternative ways of exploiting the genomics data publicly available and how to use data to better analyze DE genes is a widely-regarded question. Nevertheless, we conjecture that most of the comparison studies will involve comparing DE genes against some target gene sets of many different types (e.g., gene sets derived from ChIP-Seq, ChIA-PET, microRNA, RNA-Seq, DNA-Seq, etc.). As the scale and magnitude for such gene set comparison is rapidly increasing, it is imperative to design an efficient way of carrying out the gene set comparison.

2. Problem Formulation

Figure 1 contrasts the two methods for doing the gene set comparison, one using the conventional database programming and one using the coding scheme we report in this paper. First, lets say there is a very large collection of gene sets, say, a database of target gene sets, which is denoted by $DB_t = \{G_1, G_2, \ldots G_n\}$. Here each $G_i, 1 \leq i \leq n$, includes one or more gene identifiers. Let there be, namely, a query gene set, G_q, which also includes one or more gene identifiers. The goal is how fast one can compare the query set G_q, with each gene set stored in DB_t, which should be typically stored in a database. Figure 1a illustrates the database programming approach. One can write a SQL statement that performs intersection between G_q, and each

gene set in DB_t . In this approach the DBMS will fetch a group of gene sets from the database (via disk I/O) and carry out the intersection in memory. Figure 1b illustrates the scenario in which someone designs an effective data compression method, the entire gene sets of DB_t can be stored in the memory and the entire intersection operations can occur in memory. As the figure suggests, each target gene set of DB_t is encoded once and the encoded gene set is stored in the disk. This encoded gene set of DB_t is read once and can reside in memory and is repeatedly used in comparing a series of input query gene sets. Our coding scheme approach would be far more efficient than the SQL based approach because processing of each target gene set will not incur any additional disk I/O which is not the case in the SQL approach.

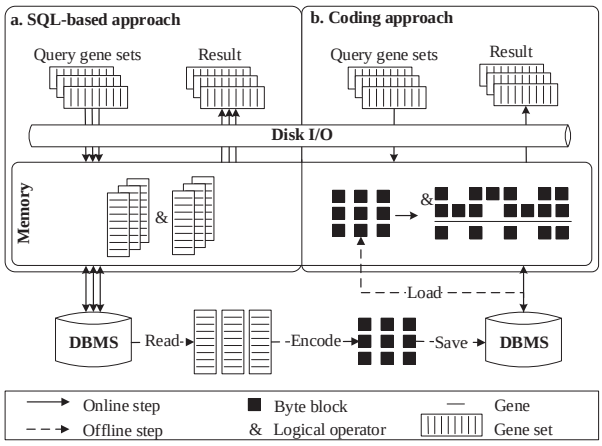


Fig. 1: SQL-based approach and coding approach with memory and disk I/O usage in both approaches.

One issue in our approach is estimating realistically how big a DB_t our coding scheme can store in memory of an ordinary computer workstation. Our plan is given a byte (8 bits), we use one bit to present a gene identifier. Each byte ranges from 00000000 to 11111111 and each bit position in the byte is tied to a gene identifier uniquely. Each byte can then represent up to 8 gene identifiers depending on which bit is set to "1" as opposed to "0". If we assume the size of the entire known human genes is 24000, this coding scheme will entail up to 3000 $bytes$ $(= 24000/8)$ to express a gene set including the entire known human gene identifiers. If we estimate that a typical DE gene set has less than 1000 genes in it, then this coding scheme may need to use up to 1000 bytes. The worst case is when only one bit is set in each byte. A different coding scheme can be used which can squeeze in up to $2^8 - 1 = 255$ gene identifiers in a byte. In this case only about 100 bytes would be enough to store the entire 24000 genes. The former encoding scheme has a lesser data compression factor but the bitwise operation needed in this scheme is far more straightforward than the one needed in the latter coding scheme. In the former scenario, DE sets from one million experiments can be stored in 1GB memory which is affordable in typical workstation computers.

The rest of the paper is organized as follows. In Section 3, we present related works of using bitwise coding schemes in genomics data analysis. In Section 4, we provide the basics of bitwise data structure and the associated Boolean operations. We also discuss other topics related to motivating the use of the encoding scheme. In Section 5, we describe the details of how gene sets are encoded and decoded, and how encoded gene set data structure is processed. The running time complexity of the operation is also analyzed and discussed. Section 6 is the conclusion.

3. Related Works

Two of the tools applying bitwise data structure and operations to perform high-throughput data analysis are BiForce [8] and BOOST (BOolean Operation-based Screening and Testing) [9]. In BiForce, the goal is to uncover gene-gene interactions (epistasis) from genome-wide association studies (GWAS). It uses Boolean bitwise operations and multithreaded parallelization to speed up comparisons of billions of single nucleotide polymorphism (SNP) combinations obtainable from a pair-wise genome scan. Its coding scheme introduces three types of genotype in which the type is represented by the values of the set $\{0, 1, 2\}$. Each value is mapped to a set of 3 bits S_x^0, S_x^1, S_x^2 where S_x^i, $0 \leq i \leq 2$, is set to 0 or 1 among 3 bits depending on whether the genotype is present or missing. By default, all bits are set 0. If a genotype is present, one and only one of 3 bits is turned on as 1, otherwise, none of 3 bits is set. For example, let $S_1^0 = |1\ 0\ 1|$ denoting genotype 0, and $S_2^1 = |1\ 1\ 1|$ denoting genotype 1. Then by using the bitwise AND operation on each pair in S's and counting the 1's of the outcome, one can produce the binary traits table, say, $n_{1,2} = |S_1^0 \wedge S_2^1| = |1\ 0\ 1 \wedge 1\ 1\ 1| = |1\ 0\ 1| = 2$. The authors report that BiForce completed analyses of the eight metabolic traits within 1 day on a 32-node cluster computer, identifying nine epistatic pairs of SNPs in five metabolic traits and 18 SNP pairs in two disease traits using two sets of about 340K SNPs from GWAS cohorts.

BOOST also uses bitwise data structure and operators on epistasis data obtainable from GWAS. The authors introduce a Boolean representation of genotype data. Each row is to represent one specific type of genotype consisting of two-bit strings: one obtained from control samples and the other obtained from test samples. Each SNP has 3 rows for three types of genotypes instead of one row as in BiForce. Each bit in the string represents one sample, and its value (0 or 1) indicates whether the sample has the corresponding genotype or not. The authors give an easy to follow example in [9] illustrating that the method uses the bitwise operation AND on each corresponding row, then counts the number of 1's (e.g., hamming weight) taking the advantage of 64 bit AND operation.

There have been other works in bioinformatics that also exploit bitwise operation. A coding scheme and bitwise operators have been applied to mapping high-throughput bisulfite sequencing reads to the reference genome in the system called BSMAP [10]. BSMAP combines genome hashing and bitwise masking to achieve bisulfite mapping where DNA sequences are converted into binary strings by encoding each DNA nucleotide into two bits (i.e., A:

00, C: 01, G:10, T:11). The authors suggest that due to this encoding scheme high-throughput bisulfite reads can be mapped at whole genome level with feasible memory and CPU usage. We note that none of the aforementioned existing works uses the coding scheme of our work presented in this paper.

High throughput computing requires that the tool should aim to "gene sets" instead of individual gene. Gene set interactions with multiple categories have been integrated into an analysis toolkit called WebGestalt from Wang et al. [7]. Recursively applying on a pair interaction makes it possible to combine information from any number of gene sets. However, it has a limitation of performing multiple combinations at the same time in some experiments. Oftentimes, the comparisons of all combinations will show a bird's eye perspective and better understanding the biological system. We also want to calculate the information content variables (e.g., information gain) for enrichment analysis of gene set interactions. Other interactions such as gene and gene or gene and environment ones have been discussed [11], and in general perspective of genetics [12].

4. Methods

4.1 Bitwise Conversion

Boolean logical operation has been used widely for the query system which computes typical comparisons such as intersection and set difference. In order to compare two gene sets using bitwise operator, we need to convert the gene set into bits. For example, let A and B be two DE gene sets obtained from two experiments. Performing intersection of the two gene sets could be finding the genes that are consistently up- or down-regulated in both experiments. Difference query displays the genes that are expressed in experiment A but not in experiment B, or vice versa [7]. Figure 2 is an example of bitwise encoding and bitwise AND operation between query and target gene sets.

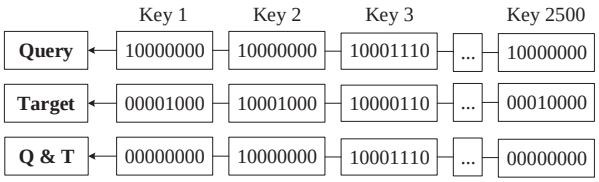


Fig. 2: A hypothetical example of performing logical bitwise AND operator between query gene set and target gene set.

Bitwise encoding is a conversion process of the query and target gene sets into coded key and value map. We will segment the total genes into an eight bit format. For example, let 20000 be the total number of universe genes. Then we will have $20000/8 = 2500$ keys (bytes) with each key representing presence of 8 genes in a gene set. The genes in key k $(1 \leq k \leq 2500)$ represents genes which have index from $(k-1)*8+1$ to $k*8$. If gene i exists, the gene position index will be calculated using modular 8 calculation, because we have an 8 bit structure. In Figure 2, in query gene set, key 1 has one gene (gene 1), key 2 has 1 gene (gene 9), and key 3 has 4 genes (17, 21, 22, 23), and key 2500 has one gene (19993). Similarly target gene set has genes with ids of 5, 9, 17, 22, 23, 19996. We apply AND bitwise operator between value of a specific key of query gene set and the same key in the target gene set. Genes with ids of 9, 17, 22, 23 are in common and there is no common gene in key 2500.

4.2 Bitwise Coding Approach for Gene Set Comparison

In this section, we describe new bitwise coding approach and compare it with the SQL-based approach. The work flow of bitwise approach is shown in Figure 3.

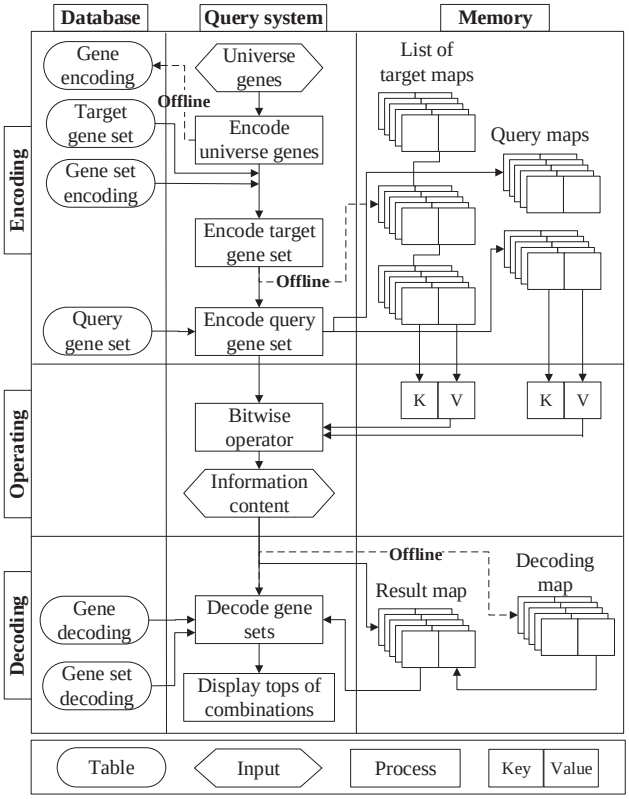


Fig. 3: The work flow of coding approach has three steps: encoding, operating and decoding. Encoding phase has two off-line steps for encoding universe gene set and all the target gene sets once into memory. Decoding phase includes one off-line step to create decoding map.

It consists of three major steps: encoding, operating and decoding. In encoding phase, the universe gene set, target gene set and query gene set are collected and encoded into a bitwise structure with key and value pair as described in Section 4.1. Encoding universe gene set and target gene set is performed off-line. Gene encoding table contains keys and values for every gene. If a gene is present in the set, the corresponding key and value are generated. Encoded gene sets are archived in the database, and loaded into the computer memory when the computation is needed.

Operating phase is performing a comparison between the input query set against the encoded target gene sets to

produce various parameters (e.g., information gain, ratio, count. etc.) which will be needed to derive biological interpretations. One noticeable fact is that such comparison can be performed entirely in memory (see Figure 3). Decoding phase is to translate the comparison output obtained in bit-wise structure into a list of gene identifiers. This translation step will need to use a decoding table that specifies how the key and value of each gene are mapped into gene identifiers. Such decoding will be an one-time process and can be done off-line only when doing so is needed. The decoding process is $(key - 1) * 8 + value$. For example, if the commonly found gene has $key = 10,\ value = 2$ then it is $(10 - 1) * 8 + 2 = 74^{th}$ gene in the universe gene set. Three steps denoted in Figure 3 as "Offline" are processes of encoding the universe gene set, encoding the target gene set and generating the decoding map.

Figure 4 shows the work flow of the alternative SQL-based approach with two major steps: loading phase and operating phase. Loading phase includes reading target and query gene sets. Here, the DBMS query system will be responsible for generating the output of comparing the input query gene set and the database stored target gene sets. This SQL-based approach will incur disk I/Os to bring in blocks of target gene sets into the memory which would become the dominating factor responsible for making the SQL-based comparison process slow.

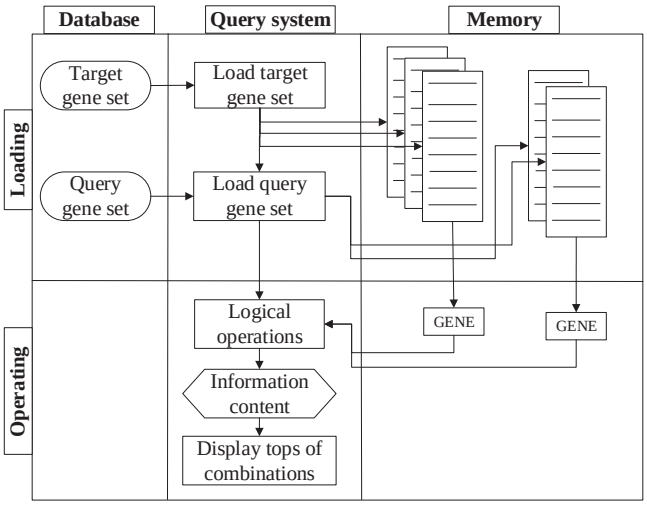


Fig. 4: The work flow of SQL-based approach has two steps: loading and operating. It needs lots of disk I/O usage for real-time computing.

4.3 Computing Information Gain

To illustrate how bitwise operations can be used on coded data structure, we show an example of computing information gain by comparing a pair of gene sets. Figure 5 depicts the overview of the process, which is analogous to construct a decision tree [13]. In Figure 5, each node of the tree contains a question. Every internal node points to one child node as a possible answer to its question and the questions are formed into a hierarchy. In decision trees, one of the most commonly used measurements is entropy. To estimate how a pair of gene sets are inter-related, let $H(X)$ and $H(Y)$ be gene set entropies for X and Y, respectively. Overlapped region in Figure 6 is $I(X;Y)$, the mutual information derived from both patterns (e.g., joint distribution) and $H(X|Y)$ and $H(Y|X)$ denote conditional distributions (see Figure 6).

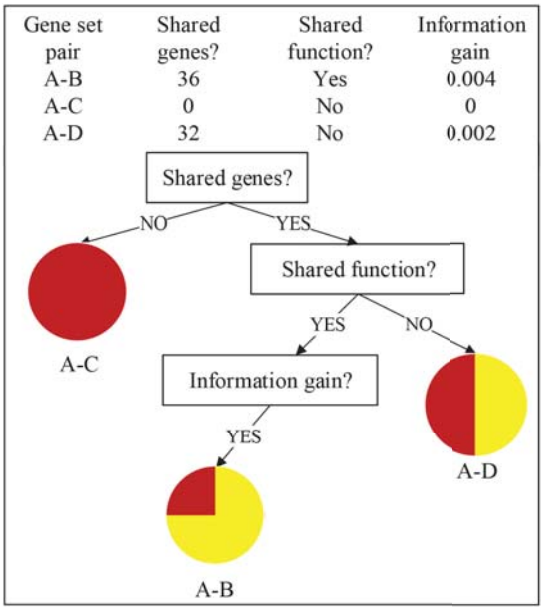


Fig. 5: Example of three gene set pairs A-B, A-C and A-D. Decision tree goes from one node to other nodes that depends on the constraints of the nodes. It chose gene set B as the best predictor of given gene set A when comparing A with B, C and D.

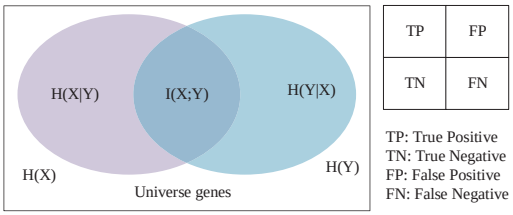


Fig. 6: Venn diagram for entropies H(X), H(Y), the mutual information I(X;Y) and conditional distribution table.

X is a set of possible values $x_1, x_2, \ldots x_n$ and corresponding probability distribution $p(x_i) = P(X = x_i),\ i = 1, 2, \ldots n$. The entropy of X is H(X), calculated as in Equation 1.

$$H(X) = -\sum_i p(x_i) log_2 p(x_i) \quad (1)$$

Mutual information is defined as the mutual information in both random gene sets X and Y in Equation 2.

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \quad (2)$$

Target gene set and query gene set are assumed to be statistically independent. The conditional entropy X given Y can be measured. Likewise, the conditional entropy Y given X can be computed using Equation 3.

$$H(X|Y) = H(X) - I(X;Y) = H(X,Y) - H(X)$$
$$H(Y|X) = H(Y) - I(X;Y) = H(X,Y) - H(Y) \quad (3)$$

Figure 5 is given to illustrate the use of information gain in solving the following problem: Given a query gene set A and the three target gene sets (B, C, D), can one determine which target gene set is more predictable of the given query gene set? To solve this problem we need to build the contingency table as illustrated in Figure 6 for each pair of gene sets (e.g., A-B, A-C and A-D). The table has four components which are labeled, respectively, true positive (TP), true negative (TN), false positive (FP) and false negative (FN). Given the pair of target and query gene sets, TP value is number of common genes between the two gene sets. TN/FP are number of genes present in query/target gene set but absent in target/query gene set. FN value is the number of universe genes that are neither in the target gene set nor in the query gene set. We discuss in the subsequent section how fast information gains can be computed using our proposed encoding scheme.

5. Case study and Discussion

In our case study, we have implemented bitwise coding method and compared the results with SQL approach using a randomly generated dataset of 30 query gene sets $(Q_1, Q_2, \ldots, Q_{30})$ and 1000 target gene sets $(T_1, T_2, \ldots, T_{1000})$. The number of genes in each set varies from 51 to 499 and is randomly selected from universal pool of 20000 genes. The target gene sets are encoded and stored in variable number of blocks ranging from 73 to 143. We have analyzed this dataset by running both SQL and bitwise approach to identify top five pairs P_i (Q_j, T_k) for two comparison cases: Intersection and Mutual Information Content. Our goal is to identify intersection between query genes sets (Q_s) and target genes sets (T_s) and also to identify mutual information contents (difference) in each pair. To balance the environmental factors, we run each experiment 10 times and report average running time. We also discuss time complexity of the methods and briefly explain the performance benefit.

5.1 Gene Set Intersection

Algorithm 1 identifies top five pairs of Q_s and T_s for highest intersection of genes. In this algorithm, gene sets are compared using conventional SQL-based approach where two tables are joined and row counts are generated using a Oracle DBMS. We compare all the possible pairs and rank the pairs based on the number of common genes.

In algorithm 2, we have used bitwise approach to achieve the same goal as Algorithm 1. The target gene sets are assumed to be already encoded and stored in the database. The encoded sets are loaded into the memory in key-value HashMap for each block. This facilitates efficient comparison between two sets using the block. A fast bitwise AND operation is performed for every common block of both sets. A result map is created and result of bitwise operation is stored as values for those corresponding keys.

Algorithm 1 Generate_Top5_Intersection_use_SQL

```
1: Initialize variables and database connection.
2: Read target gene set.
3: Read query gene set.
4: for (all queries) do.
5:     Read target gene set and match gene id and output the ordered list of gene
   set id and count.
6:     Get the first element of list
7:     Look up the Map with this element and store query list.
8: end for
9: Sort the query list.
10: Output top five of gene set combinations.
```

Algorithm 2 Generate_Top5_Intersection_use_Bitwise

```
1: Initialize variables and environment, database connection.
2: Read target gene set.
3: Read query gene set.
4: Encode list of target gene set maps
5: for (all queries) do.
6:     Encode query map of index and value.
7:     Retrieve target map given target id.
8:     Get AND value of pair-wise of query map and target map into a list.
9:     Sort the list descending.
10:    Get the first element of list
11:    Look up the map with this element and store query list.
12: end for
13: Sort the query list.
14: Output top five of gene set combinations.
```

Encoding and loading of target sets are considered offline because they can be pre-loaded into the program. Query sets need to be encoded once, and can be compared with all the loaded target sets. Loading of decoding table can also considered offline because it is pre-loaded into the program. Quick decoding is performed for all the keys in the result map and number of intersecting gene is computed by adding decoded values for all the existing keys in the result map.

5.2 Mutual Information Content

Mutual information content is computed in terms of information gain (IG) using both the methods. Algorithm 3 illustrates SQL-based approach to compute information gain between each possible pairs of query sets and target sets. We first create conditional table using set of SQL queries and use this table to compute information gain. Each pair of sets and their information gain are stored in a list and top five pairs are extracted by sorting the list in descending order of their corresponding IGs. Information gain is also computed using bitwise approach for the same dataset.

As illustrated in algorithm 4, conditional table is generated using bitwise AND operation and the values are used in computation sequence of the algorithm for final result.

Figure 7 shows average running time for one pair of query set and target set for two approaches. It is clear that TP in conditional table is computed significantly faster in bitwise approach than SQL approach. Even with decoding overhead, it is observed that True Positive (TP) calculation in bitwise approach takes seven times less than that of SQL approach. IG time only includes the calculation time when having all elements in conditional table.

Algorithm 3 Generate_Top5_Difference_use_SQL

```
Initialize variables and database connection.
Read target gene set (target id and count).
Read query gene set (query id and count).
Encode list of target gene set maps
for (all queries) do
    Read target gene set and match gene id and output the ordered list of gene set id and count.
    Calculate the conditional table values and information gain, store into a list
    Sort the list descending.
    Get the first element of list
    Look up the Map with this element and store query list.
end for
Sort the query list.
Output top five of gene set combinations.
```

Algorithm 4 Generate_Top5_Difference_use_Bitwise

```
Initialize variables and database connection.
Read target gene set (target id and count).
Read query gene set (query id and count).
Encode list of target gene set maps
for (all queries) do
    Encode query map of index and value.
    Retrieve target map given target id.
    Get AND value of pair-wise of query map and target map into a list.
    Calculate the conditional table values and information gain, store into a list.
    Sort the list descending.
    Get the first element of list.
    Look up the map with this element and store query list.
end for
Sort the query list.
Output top five of gene set combinations.
```

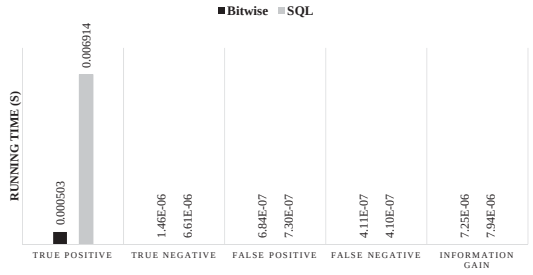


Fig. 7: Comparison of average running time for calculation of four elements in conditional table and information gain for a combination of target gene set and query gene set.

SQL approach is directly dependent on the number of genes in each set and also number of sets that are being compared against. As shown in Figure 8, running time for SQL approach is increasing linearly with the number of target gene sets. On the other hand, performance of bitwise approach is very consistent and is not affected much by the number of target sets. For a small number of comparisons, bitwise method performance is comparable with SQL approach, however, it performs significantly better as the number of comparison increases.

We also experimented with the efficiency of coding approach by checking the time taken by the program for sets with various number of genes. Conventionally with SQL, bigger sets will make the process slower. However, the proposed bitwise approach is not much affected by the number of genes in the set because the encoded data structure size is more or less fixed.

Figure 9 shows the speed comparison for entire information gain computation. When calculating information gain

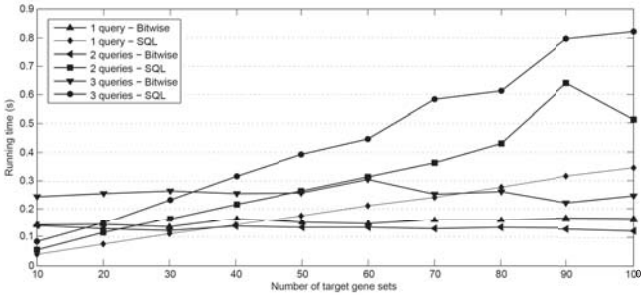
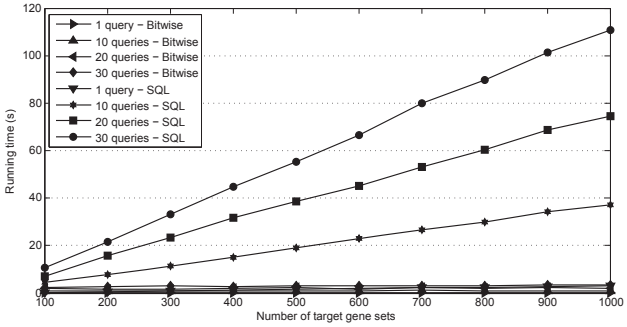
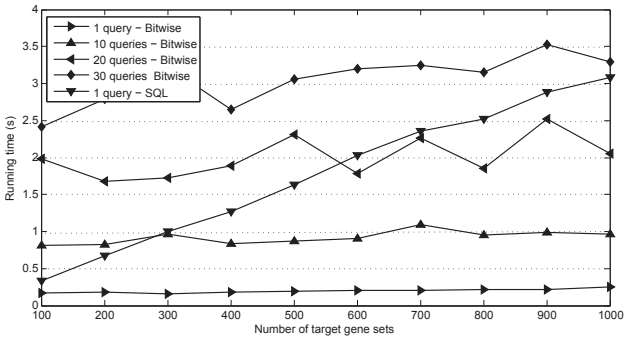


Fig. 8: Running time of 1 query, 2 query and 3 query gene sets in comparisons with 100 target gene sets.



(a) Running time of 30 query gene sets in comparisons with 1000 target gene sets.



(b) Running time of 4 types of bitwise queries (1 query, 10 queries, 20 queries and 30 queries) and 1 query using SQL in comparisons with 1000 target gene sets.

Fig. 9: Running time of 30 query gene sets in comparisons with 1000 target gene sets. Each comparison includes a query gene set and a target gene set.

with SQL, average running time increases linearly to the number of target gene sets. However with the proposed bitwise method, it virtually unaffected by the number of target gene sets. Figure 9(a) is the average running time of the program of 30 query gene sets in combinations with 1000 target gene sets. Figure 9(b) shows the average running time of 1 query, 10 queries, 20 queries and 30 queries using bitwise approach and 1 query using SQL approach. We can see that the speed of bitwise computation is very less affected by the size of input sets and comparison sets.

5.3 Time Complexity

In this section, we analyze time complexity of both approaches. The running time analysis for a single interaction between a query gene set and a target gene set for our bitwise coding approach can be explained as follows. Let n_q be the number of genes in query set, B_q be the number of blocks in this set and B_t be the number of blocks in the target gene set. The total time can be calculated by aggregating running times to encode query set, compare query gene set and target gene set, and decode the result (see Equation 4).

$$C_{bitwise} = C_{enc} + C_{comp} + C_{decode} \quad (4)$$

C_{enc} is the time to read all genes in a query set and encode into key and value format. The complexity is equal to the input size when reading all the indexes into an array. Therefore, $C_{enc} = \mathcal{O}(n_q)$ since n_q is the number of genes in a query set. C_{comp}, is the time taken to compare the blocks with indexes that are common in both sets. In order to do this, we only need to scan $\min(B_q, B_t)$ number of blocks. We have $C_{comp} = \mathcal{O}(\min(B_q, B_t))$. When $B_t = B_q$, the computation time is linear with the function of B_t (or B_q). And finally, decoding time C_{decode} is the time to look up decoding table which has the same data structure of blocks, i.e. $C_{decode} = \mathcal{O}(\min(B_q, B_t))$. Thus, average running time (same order as Equation 4) is presented below.

$$C_{bitwise} = \mathcal{O}(n_q) + \mathcal{O}(\min(B_q, B_t)) + \mathcal{O}(\min(B_q, B_t)) \quad (5)$$

For SQL approach, time complexity is the time to look up the database to retrieve the result (see Equation 6). Let n_t be the maximum number of genes in a target gene set.

$$C_{SQL} = \mathcal{O}(n_q) + \mathcal{O}(\max(n_q \log(n_q), n_t \log(n_t))) \quad (6)$$

In this approach, both sets have to be sorted and that takes time $\mathcal{O}(\max(n_q \log(n_q), n_t \log(n_t))$ since the average sorting time of n indexes in an array is $\mathcal{O}(n \log n)$. When $n_t = n_q$, the running time will be $\mathcal{O}(n_q) + \mathcal{O}(n_q \log n_q) \approx \mathcal{O}(n_q \log n_q)$. Now if we compare the time complexities of bitwise approach and SQL approach, we have $C_{bitwise} \ll C_{SQL}$. Bitwise coding method which encodes the universe gene set and target gene sets offers a way to compare two gene sets with the minimum number of blocks (bytes). In this approach the missing blocks (e.g., block with all zero values.) can be totally ignored in the comparison, thus offering a greater memory space saving and reduction in bitwise operation.

6. Conclusion

Gene set comparison is one of the key essential operations that is universally needed in many high-throughput genomic data analyses. Examples include functional enrichment analyses, ChIP-Seq peak finding, gene-gene interaction analysis, and so on. One important observation in these gene set comparison practices is that the majority of the processes is finding and comparing memberships of gene identifiers (in hundreds) in a large number of gene sets (in millions). We presented a gene identifier encoding scheme which could store a very large number of gene sets entirely in memory. This coding scheme suggests the feasibility of storing the target gene sets in memory and using it over and over again for the analysis of the high quantity of arriving query sets. Such an arrangement can be very useful in supporting learning methods and data mining methods which need to test a large number of gene identifier combinations against a fixed albeit large number of target gene sets. Our case studies demonstrated our coding scheme could offer linear time performance proportional to the number of arriving input query sets.

7. Acknowledgment

Work by TH was funded by a grant from the Vietnam Education Foundation (VEF). Work by PJ and DGS was supported in part by NIH Grant R21AA023212. Work by SHH was supported in part by NIH Grant R01AR063702-01A1. The opinions, findings, and conclusions stated herein are solely of the authors and do not necessarily reflect the official view of VEF or NIH.

References

[1] N. Siva, "1000 genomes project," *Nature biotechnology*, vol. 26, no. 3, pp. 256–256, 2008.
[2] C. G. A. R. Network *et al.*, "Integrated genomic analyses of ovarian carcinoma," *Nature*, vol. 474, no. 7353, pp. 609–615, 2011.
[3] E. P. Consortium *et al.*, "The encode (encyclopedia of dna elements) project," *Science*, vol. 306, no. 5696, pp. 636–640, 2004.
[4] R. Edgar, M. Domrachev, and A. E. Lash, "Gene expression omnibus: Ncbi gene expression and hybridization array data repository," *Nucleic acids research*, vol. 30, no. 1, pp. 207–210, 2002.
[5] D.-G. Shin, S.-H. Hong, P. Joshi, R. Nori, B. Pei, H.-W. Wang, P. Harrington, L. Kuo, I. Kalajzic, and D. Rowe, "Pbc: A software framework facilitating pattern-based clustering for microarray data analysis," in *Bioinformatics, Systems Biology and Intelligent Computing, 2009. IJCBS '09. International Joint Conference on*, Aug 2009, pp. 30–36.
[6] P. Joshi, B. Pei, S.-H. Hong, I. Kalajzic, D.-J. Shin, D. Rowe, and D.-G. Shin, "A software framework integrating gene expression patterns, binding site analysis and gene ontology to hypothesize gene regulation relationships," in *Bioinformatics and Biomedicine (BIBM), 2013 IEEE International Conference on*, Dec 2013, pp. 210–213.
[7] B. Zhang, S. Kirov, and J. Snoddy, "Webgestalt: an integrated system for exploring gene sets in various biological contexts," *Nucleic acids research*, vol. 33, no. suppl 2, pp. W741–W748, 2005.
[8] A. Gyenesei, J. Moody, C. A. Semple, C. S. Haley, and W.-H. Wei, "High-throughput analysis of epistasis in genome-wide association studies with biforce," *Bioinformatics*, vol. 28, no. 15, pp. 1957–1964, 2012.
[9] X. Wan, C. Yang, Q. Yang, H. Xue, X. Fan, N. L. Tang, and W. Yu, "Boost: A fast approach to detecting gene-gene interactions in genome-wide case-control studies," *The American Journal of Human Genetics*, vol. 87, no. 3, pp. 325–340, 2010.
[10] Y. Xi and W. Li, "Bsmap: whole genome bisulfite sequence mapping program," *BMC bioinformatics*, vol. 10, no. 1, p. 232, 2009.
[11] R. Fan, M. Zhong, S. Wang, Y. Zhang, A. Andrew, M. Karagas, H. Chen, C. Amos, M. Xiong, and J. Moore, "Entropy-based information gain approaches to detect and to characterize gene-gene and gene-environment interactions/correlations of complex diseases," *Genetic epidemiology*, vol. 35, no. 7, pp. 706–721, 2011.
[12] C. Wu, S. Li, and Y. Cui, "Genetic association studies: an information content perspective," *Current genomics*, vol. 13, no. 7, p. 566, 2012.
[13] C. Kingsford and S. L. Salzberg, "What are decision trees?" *Nature biotechnology*, vol. 26, no. 9, pp. 1011–1013, 2008.

Exploring Effects Of Various Data Preprocessing Methods On The Classification Of DNA Microarray Data

Samuel Dong[1]
[1]Department of Computer Science, Western Kentucky University,
Bowling Green, Kentucky, United States of America

Abstract— *DNA microarray data can be very useful in the prediction of diseases. There have been several studies into machine learning to predict a disease from microarray data, but the high-dimensionality and imbalanced classes make it difficult. In this paper, we investigate the effectiveness of sampling to improve class balance and feature selection to reduce the dimensionality by testing them on prostate cancer microarray data. We used two sampling methods, one oversampling and one undersampling, as well as five feature selection methods with five subset sizes each. For all combinations, three classifiers were trained and then evaluated to see which combination yielded the best results. The empirical results show that SMOTE is significantly useful, as well as using any feature selection method with a low subset size such as 50. While classifiers were not the focus of this study, results also show that k-nearest neighbors outperforms both Naive-Bayes and decision trees when classifying microarray data.*

Keywords: DNA Microarray, Feature Selection, Sampling, SMOTE, WEKA, Prostate Cancer

1. Introduction

The use of computers to aid in cancer research has recently been very active. One such pursuit is using computers to take in various data and make predictions on whether or not a patient has cancer. Our research is interested in DNA microarray data, which is a technique that involves having a plate of many different genes. The sampled DNA is then placed in to see how well it binds with the various genes. Genes that exactly match those in the DNA bind perfectly while genes that are similar bind partially. The amount of binding is recorded for each gene. This produces a large feature vector consisting of tens of thousands of genes and how well the DNA bound with each. This high dimensionality makes it very difficult to identify patterns in the data. The other issue is the imbalance of cancer and non-cancer patients. In a sample population, only a few people have cancer while the rest do not. This class imbalance proves to be an issue for classification.

There have been numerous studies on using machine learning and data mining techniques on DNA microarray data. Many studies either focus on sampling or feature selection, but few focus on both. Our study analyzes the combined effect of both sampling and feature selection. We do this by taking the data and applying sampling and feature selection and analyzing the processed data with classifiers. We then evaluate the classifiers on the processed data in order to determine the effectiveness of the processing methods.

In this paper, we investigate two sampling methods, resampling and SMOTE, to correct the class imbalance and five feature selection methods, Chi-Squared (CS), Information Gain (IG), Gain Ratio (GR), ReliefF (RF), and Symmetric Uncertainty (SU), to reduce the dimensionality. We also investigate the number of features selected and its impact on the effectiveness of the classification. We use three classification methods to learn the data. We perform cross-validation for each classifier and average them.

Our study showed that the feature selection, especially when 50 attributes were selected, has a significant positive impact on the performance of the classifiers, but no individual feature selection method was significantly better than the others. Sampling also significantly improved performance, with resampling (undersampling) performing better than none, and SMOTE (oversampling) performing the best. In terms of classifiers, k-nearest neighbors (IBK) performed the best, followed by Naive-Bayes (NB) and finally decision trees (DT).

This paper is organized as follows. Section 2 briefly discusses previous work done on the subject. It is followed by Section 3 which describes in detail the methods that were used in the study. Section 4 gives information about the dataset used in the experiment, the design of the experiment, and the results and analysis of the experiment. This is followed by a brief conclusion and future work in Section 6.

2. Related Work

Due to the high dimensionality of DNA Microarray datasets, it has become necessary to include dimension reduction techniques. Researchers and practitioners are interested in which feature subset is best suited for the model building process when high-dimensional datasets are considered. Guyon and Elisseeff [1] outlined key approaches used for attribute selection, including feature construction, feature ranking, multivariate feature selection, efficient search methods, and feature validity assessment methods. They conclude

that variable and feature selection improves the prediction performance of the inductive models. Liu and Yu [2] provided a comprehensive survey of feature selection algorithms and presented an integrated approach to intelligent feature selection.

Class imbalance is another problem within DNA Microarray datasets, especially where there are few cases of the positive class. This may result in a large number of misclassifications of instances of the positive class. Lusa et al. [3] found that there is a bias towards the majority class when working with imbalanced data. They also stated that unless we take steps to combat class imbalance, the performance of the analysis will suffer. The primary technique used to deal with this problem is known as sampling, which generates more balanced datasets by adding or removing instances. A comprehensive study on different sampling techniques was performed by Kotsiantis [4] and Guo [5].

3. Methodology

Our research focused on three main steps: sampling, feature selection, and classification. To perform these operations, we used the University of Taikato's WEKA Java API.[6] We used a total of two sampling methods, five feature selection methods, and three classification methods.

3.1 Sampling

Sampling is used to correct the class imbalance issue. This can either be done by oversampling, in which more instances are added to the minority class, or undersampling, in which instances are removed from the majority class.

To undersample, we used WEKA's Resample filter which takes a random subsample. By setting the bias toward uniform class to 1, we ensured that the output subsample was balanced.

To oversample, we used WEKA's SMOTE filter. SMOTE stands for Synthetic Minority Oversampling Technique. It works by generating synthetic instances based on the existing instances in the minority class to balance the data. The synthetic instances are generated by taking random points along a line between an existing minority instance and its nearest neighbors. It has been shown to improve the accuracy of classifiers for a minority class.[7]

3.2 Feature Selection

Feature selection is used to reduce the high dimensionality of the data. There are two types of feature selection methods: subset evaluation feature selection and ranking feature selection. Subset evaluation feature selection analyses all possible subsets of the attributes and chooses the best subset. Ranking feature selection provides a rank for each attribute which can then be used to choose the top N attributes to use. For our research, we only used ranking feature selection because the high dimensionality makes subset evaluation feature selection impractical. We used the following feature selection metrics:

1) Chi-Squared (CS) - This metric provides a measure of how similar the probability distribution of two attributes. It is used to determine how independent two attributes are. [8]
2) Information Gain (IG) - This metric provides a measure of the decrease in entropy when an attribute is learned. [9] It is calculated using the equation below:

$$IG(T,a) = H(T) - \sum_{v \in values(a)} \left(\frac{|\{x \in T | x_a = v\}|}{|T|} \cdot H(\{x \in T | x_a = v\}) \right) \quad (1)$$

   where T is the set of all instances, a is an attribute, and H() is entropy.
3) Gain Ratio (GR) - This metric divides information gain by intrinsic value to counteract the bias toward attributes with a large number of values that information gain has. Intrinsic value is calculated using the equation below:

$$IV(T,a) = - \sum_{v \in values(a)} \left(\frac{|\{x \in T | x_a = v\}|}{|T|} \cdot \log_2 \left(\frac{|\{x \in T | x_a = v\}|}{|T|} \right) \right) \quad (2)$$

   Gain ratio is then calculated as follows:

$$GR(T,a) = \frac{IG(T,a)}{IV(T,a)} \quad (3)$$

4) ReliefF (RF) - This metric uses an iterative approach to calculating the relevance of attributes. [10] It is slightly improved version of RELIEF by Kononenko et al [11]. It starts with a weight vector W which starts at zero for each attribute. It then goes through each instance once and finds k-nearest neighbor instances of the same class and k-nearest neighbor instances of the opposite class based on the Manhattan distance. The closest instances that are the same class are called near-hits and the closest instances that are the opposite class are called near-misses. For each instance, the following equation is used to update the weight vector:

$$W_i = W_i - \frac{\sum_{n=0}^{k-1} |x_i - nearHit_n| - |x_i - nearMiss_n|}{k} \quad (4)$$

   The weight vector is then divided by the number of instances to get the relevance vector. Each value in the vector represents the relevance of that particular attribute.
5) Symmetrical Uncertainty (SU) - This metric represents the weighted average of two uncertainty coefficients.

It is based on the mutual information between two attributes which is a measure of how independent two attributes are. [12] The mutual information is calculated as follows:

$$I(X;Y) = H(X) - H(X|Y) \tag{5}$$

where X and Y are two attributes and H() is the entropy. Symmetric uncertainty is then calculated as:

$$SU(X,Y) = 2\frac{I(X;Y)}{H(X) + H(Y)} \tag{6}$$

The attributes with higher independence and therefore less redundancy have higher symmetrical uncertainty values.

3.3 Classification

We used three classification methods: decision tree (DT), Naive-Bayes (NB), and k-nearest neighbors (IBK). In WEKA, J48, which is a Java implementation of C4.5, is used for the decision tree. J48 builds a decision tree based on determining the best attributes to split the set into subsets using information gain. The Naive-Bayes classifier determines the probability of a instance belonging to a class based on the values of its attributes. WEKA also uses IBk for the k-nearest neighbors classifier. IBk finds the k-nearest neighbors and finds the majority class with more weight on closer instances.

3.4 Evaluation Metric

We used the area under the ROC (Receiver Operator Characteristic) curve or AUC (Area Under Curve) values to evaluate the classifiers. The ROC curve shows the rate of true and false positives as the decision threshold varies. The area under the curve represents the performance of the classifier under all thresholds. The value ranges from 0 to 1 where 1 is a perfect classifier. This metric is well suited for unbalanced data and has been shown to be more stable than other performance metrics.

4. Experiment

4.1 Dataset

The dataset we used in our experiments came from Kent Ridge Biomedical Dataset [13] which contains numerous high-dimensional datasets. We used their Prostate Cancer DNA microarray data. Originally, it had 12600 attributes and 132 instances. 75 of the instances were tumor and 59 of the instances were non-tumor. In order to test class imbalance, we randomly removed tumor instances to reach a ratio of 80% non-tumor to 20% tumor. The final dataset used in the experiment contains 59 non-tumor and 15 tumor instances.

4.2 Experimental Design

Using Java and WEKA's API, the experiment was performed in three stages:

1) Sampling - The original data is taken and sent through through two sampling methods, SMOTE and Resample. SMOTE's percent parameter was set to 300% and Resample's sample size percent set to 40% and bias to uniform class parameter to 1. These produce new datasets that are balanced. After this stage, there are three datasets, the control dataset with no sampling, the oversampled dataset with SMOTE, and the undersampled dataset with Resample.
2) Feature Selection - We used five feature selection methods: Chi-squared, information gain, gain ratio, ReliefF, and symmetrical uncertainty. Since these are all ranker feature selection methods, we had to choose the number of top attributes to choose. We used the following subset sizes: 50, 100, 200, 1000, and 2000. With five feature selection methods with five subset sizes each, there are 26 new datasets (one extra without feature selection for control) from each of the three datasets after sampling, which is a total of 78 datasets.
3) Classification - Each of the resulting datasets are used to train three classifiers and test them with cross-validation. We used J48 decision tree, Naive-Bayes classifier, and IBk for the 5 nearest neighbors weighted by 1 / distance. We performed ten runs of three-fold cross-validation. This resulted in 30 AUC (Area Under Curve) results, which were then averaged to get the final value. This results in 234 (78 datasets X 3 classifiers) total values.

5. Results and Analysis

Tables 1 through 3 show the averaged AUC (Area Under Curve) values for different sampling, feature selection, and classification methods. The first two columns represent the feature selection method and the size of the subset of attributes chosen (No feature selection cannot have a subset size). The last three columns represent each of the classifiers that were used in the experiment. DT stands for decision tree, NB stands for Naive Bayes, and IBk stands for k-nearest neighbors. Figures 1 and 2 show AUC (Area Under Curve) versus subset size for a specific sampling method and classifier. Figure 2, which represents SMOTE, has higher AUC values than Figure 1, which represents no sampling. Also, all feature selection methods and subset sizes outperform the baseline of no feature selection. Looking at each of the tables, k-nearest neighbors seems to perform the best, followed by Naive-Bayes, and lastly decision trees. An interesting exception is when there is no feature selection, decision trees outperform Naive-Bayes. This is most likely due to the high-dimensionality which renders the Naive part

of Naive-Bayes (Conditional independence for all attributes) unrealistic.

In order to confirm our findings, we used Analysis of Variance (ANOVA) tests and multiple comparisons with a significance level of $\alpha = 0.05$. Table 4 shows the results from applying one-way ANOVA test on subset size (Factor A). Since the p-value of 0.18E-08 is less than 0.05, subset size has a statistically significant impact on classifier performance. Figure 4 shows that a subset size of 50 performs better than the other subset sizes (subset sizes of 200 and 1000 are not significantly different as shown).

We then only considering subset sizes of 50. We ran three-way analysis of variance on the modified data. Table 5 shows the results of that test. Factor A is sampling method, factor B is feature selection method, and factor C is classification method. The p-value for every combination except all three are 0, which is statistically significant. This signifies that there are significant effects on the performance when any of the three factors are modified. The last combination, A x B x C, is not statistically significant, but this is most likely because there is only one data point for each combination, which would lead to a high variance. In figure 3, the three factors are shown separately. As expected, figure 3(a) shows that SMOTE significantly outperformed the baseline of no sampling. Resample also outperformed the baseline but underperformed SMOTE. In figure 3(b), all feature selections methods significantly outperformed the baseline of no feature selection, however, there is no significant difference between which feature selection method we used. In figure 3(c), k-nearest neighbors is shown to perform the best, followed by Naive-Bayes, and decision trees as expected.

Table 1: No Sampling

FS	Subset	Classifier		
		DT	NB	IBk
CS	50	0.7869	0.9564	0.9755
	100	0.7848	0.9639	0.9671
	200	0.7608	0.9355	0.9417
	1000	0.7243	0.8778	0.9603
	2000	0.7243	0.8507	0.9491
GR	50	0.8032	0.9897	0.9311
	100	0.7643	0.9725	0.9225
	200	0.7384	0.9530	0.9189
	1000	0.7255	0.8800	0.9603
	2000	0.7206	0.8503	0.9491
IG	50	0.7854	0.9766	0.9718
	100	0.7945	0.9548	0.9723
	200	0.7724	0.9398	0.9640
	1000	0.7206	0.8810	0.9602
	2000	0.7206	0.8552	0.9491
RF	50	0.8105	0.9792	0.9877
	100	0.7984	0.9689	0.9837
	200	0.7915	0.9489	0.9697
	1000	0.7646	0.8161	0.9160
	2000	0.7641	0.7618	0.9071
SU	50	0.7852	0.9655	0.9491
	100	0.7771	0.9790	0.9629
	200	0.7606	0.9415	0.9538
	1000	0.7292	0.8814	0.9602
	2000	0.7206	0.8540	0.9491
None	full	0.6965	0.5832	0.8177

Table 2: Resampling

FS	Subset	Classifier		
		DT	NB	IBk
CS	50	0.8863	0.9161	0.9995
	100	0.8737	0.9135	1.0000
	200	0.8147	0.9099	1.0000
	1000	0.8147	0.8876	0.9928
	2000	0.8147	0.8672	0.9670
GR	50	0.8704	0.9396	0.9967
	100	0.8307	0.9357	0.9871
	200	0.8168	0.9111	0.9900
	1000	0.8147	0.8659	0.9629
	2000	0.8147	0.8575	0.9694
IG	50	0.8863	0.9309	0.9998
	100	0.8863	0.9185	0.9995
	200	0.8525	0.8951	1.0000
	1000	0.8147	0.8883	0.9984
	2000	0.8147	0.8686	0.9877
RF	50	0.7592	0.9098	1.0000
	100	0.7739	0.8946	0.9985
	200	0.7980	0.8858	0.9907
	1000	0.7707	0.7849	0.9690
	2000	0.8245	0.6941	0.9616
SU	50	0.8307	0.9413	0.9989
	100	0.8147	0.9022	0.9975
	200	0.8147	0.8986	0.9922
	1000	0.8147	0.8908	0.9866
	2000	0.8147	0.8684	0.9678
None	full	0.8018	0.6610	0.9260

Table 3: SMOTE

FS	Subset	Classifier		
		DT	NB	IBK
CS	50	0.9210	0.9878	0.9968
	100	0.9130	0.9885	0.9981
	200	0.9043	0.9851	0.9988
	1000	0.8832	0.9724	0.9953
	2000	0.8740	0.9636	0.9939
GR	50	0.9162	0.9939	0.9934
	100	0.9098	0.9886	0.9961
	200	0.9085	0.9860	0.9968
	1000	0.8848	0.9787	0.9972
	2000	0.8745	0.9704	0.9964
IG	50	0.9264	0.9888	0.9964
	100	0.9143	0.9890	0.9971
	200	0.9000	0.9924	0.9988
	1000	0.8780	0.9739	0.9952
	2000	0.8724	0.9664	0.9940
RF	50	0.9003	0.9892	0.9988
	100	0.8999	0.9883	0.9979
	200	0.9035	0.9859	0.9968
	1000	0.9028	0.9747	0.9936
	2000	0.8968	0.9622	0.9920
SU	50	0.9192	0.9926	0.9928
	100	0.9126	0.9850	0.9979
	200	0.9063	0.9831	0.9975
	1000	0.8770	0.9786	0.9964
	2000	0.8724	0.9703	0.9947
None	full	0.8698	0.7029	0.9855

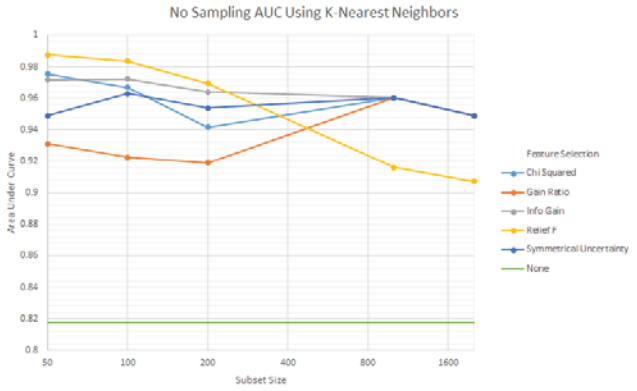


Fig. 1: No Sampling Using K-Nearest Neighbors AUC Graph

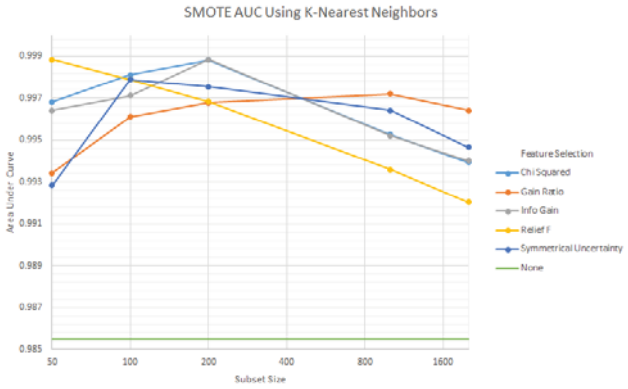


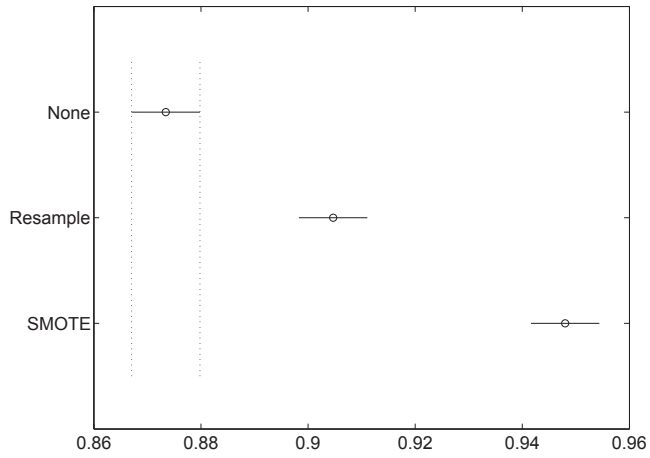
Fig. 2: SMOTE Using K-Nearest Neighbors AUC Graph

Table 4: One-way Analysis of Variance

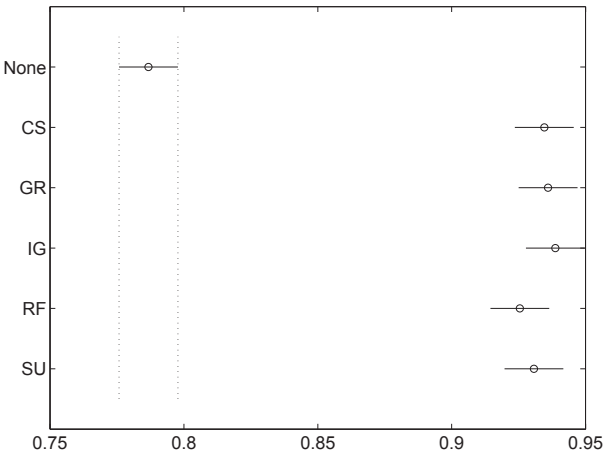
Source	Sum Sq.	d.f.	Mean Sq.	F	p-value
A	0.597	4	0.14932	9.84	6.18E-08
Error	102.356	6745	0.01518		
Total	102.954	6749			

Table 5: Three-way Analysis of Variance

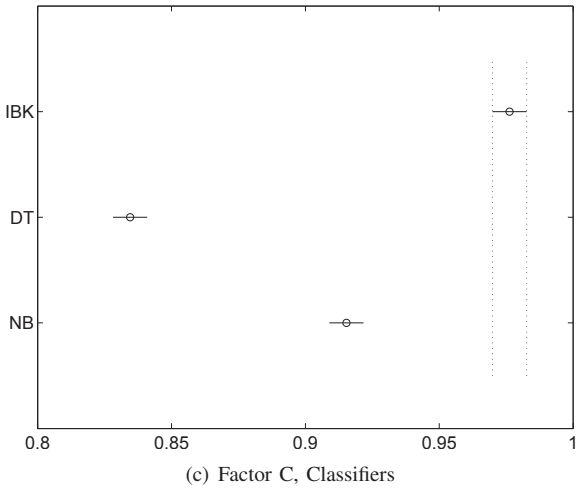
Source	Sum Sq.	d.f.	Mean Sq.	F	p-value
A	1.5143	2	0.75717	94.38	0
B	4.845	5	0.96901	120.79	0
C	5.4557	2	2.72787	340.04	0
A×B	0.4614	10	0.04614	5.75	0
A×C	0.6852	4	0.1713	21.35	0
B×C	3.504	10	0.3504	43.68	0
A×B×C	0.1652	20	0.00826	1.03	0.4226
Error	12.5629	1566	0.00802		
Total	29.1938	1619			



(a) Factor A, Samplings



(b) Factor B, Filters



(c) Factor C, Classifiers

Fig. 3: Tukey's HSD, Classification

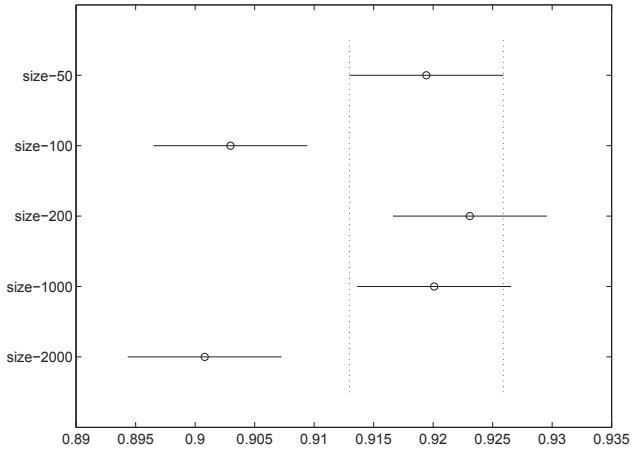


Fig. 4: Tukey's HSD, Size

6. Conclusion

DNA microarray data has high-dimensionality and is usually imbalanced. These two challenges can be remedied with feature selection and sampling respectively. We used two sampling methods, five feature selection methods, and three classifiers to test the effectiveness of feature selection and sampling in dealing with the challenges of microarray data.

Our results show that sampling, especially SMOTE, provides a very effective way of dealing with class imbalance. Feature selection, especially with a subset size of 50, is also very effective in dealing with high-dimensionality. All of our feature selection methods were equally effective in increasing the performance of the classifiers. While classification was not the main focus of our experiment, k-nearest neighbors provides a signification better performance than either Naive-Bayes or decision trees. In the future, we would like to test this approach with more datasets and possibly live data, as well as test different classifiers.

References

[1] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, March 2003.

[2] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 4, pp. 491–502, 2005.

[3] L. Lusa and R. Blagus, "The class-imbalance problem for high-dimensional class prediction," in *Machine Learning and Applications (ICMLA), 2012 11th International Conference on*, vol. 2, Dec 2012, pp. 123–126.

[4] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, "Handling imbalanced datasets: A review," in *GESTS International Transactions on Computer Science and Engineering*, vol. 30(1), Dec 2006, pp. 25–36.

[5] X. Guo, Y. Yin, C. Dong, G. Yang, and G. Zhou, "On the class imbalance problem," in *Natural Computation, 2008. ICNC '08. Fourth International Conference on*, vol. 4, Oct 2008, pp. 192–201.

[6] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed. Morgan Kaufmann, 2005.

[7] K. W. Bowyer, N. V. Chawla, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *CoRR*, vol. abs/1106.1813, 2011. [Online]. Available: http://arxiv.org/abs/1106.1813

[8] A. C. Cameron, P. Hammond, A. Holly, and P. K. Trivedi, *Regression Analysis of Count Data*. Cambridge: Cambridge Univ. Press, 1998. [Online]. Available: https://cds.cern.ch/record/997610

[9] J. R. Quinlan, "Induction of decision trees," *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, Mar. 1986. [Online]. Available: http://dx.doi.org/10.1023/A:1022643204877

[10] K. Kira and L. A. Rendell, "A practical approach to feature selection," in *Proceedings of the Ninth International Workshop on Machine Learning*, ser. ML92. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1992, pp. 249–256. [Online]. Available: http://dl.acm.org/citation.cfm?id=141975.142034

[11] I. Kononenko, E. Simec, and M. Robnik-Sikonja, "Overcoming the myopia of inductive learning algorithms with RELIEFF," *Appl. Intell.*, vol. 7, no. 1, pp. 39–55, 1997. [Online]. Available: http://dx.doi.org/10.1023/A:1008280620621

[12] M. A. Hall and G. Holmes, "Benchmarking attribute selection techniques for discrete class data mining," *IEEE Trans. on Knowl. and Data Eng.*, vol. 15, no. 6, pp. 1437–1447, Nov. 2003. [Online]. Available: http://dx.doi.org/10.1109/TKDE.2003.1245283

[13] J. Li and H. Liu, "Kent ridge bio-medical dataset," http://datam.i2r.a-star.edu.sg/datasets/krbd/, 2003, [Online; accessed 3-February-2015].

A Simplified Linear Model for Reconstruction of Gene Regulatory Networks

X. Wu[1], B. Yang[1], W. Koh[1], P. Gong[2], and C. Zhang[1]
[1]School of Computing, University of Southern Mississippi, Hattiesburg, MS, U.S.
[2]Environmental Laboratory, U.S. Army Engineer Research and Development Center, Vicksburg, MS, U.S.

Abstract - *Many inference methods have been proposed to reconstruct unknown Gene Regulatory Networks (GRN) using microarray datasets. State Space Model (SSM) is a method that can be used to infer GRNs from a time series dataset. However, there exist two difficulties in SSM when it is applied for GRN reconstruction: how to estimate initial values of parameters and how to learn the gene-gene interactions in the networks. We introduce a Simplified Linear Model (SLM) which uses Principle Component Analysis (PCA) to reduce the dimension of the dataset and the noise. A total of 20 synthetic GRNs were generated by GeneNetWeaver and they were used to test the average inference accuracies of both SSM and SLM. Our results show that SLM is more stable with different length of hidden variables and it can be applied to a smaller dataset. The proposed SLM significantly improved the performance of GRN reconstruction.*

Keywords: Gene Regulatory Networks, State Space Model, Simplified linear model, Principle Component Analysis

1 INTRODUCTION

Gene-gene interactions cannot be directly measured. The microarray technology measures intermediate products of genes such as concentrations of proteins and mRNAs. How to reconstruct the Gene Regulatory Network (GRN) from measurement is a difficult problem. GRN can be represented by a directly connected graph that a vertex stands for gene and directed edge stands for regulation relationship. In this paper we investigate how to reconstruct GRN from only time series data of concentration of mRNAs. Several methods have already been proposed, such as Dynamic Bayesian Network (DBN) [1, 2], Probability Boolean Network (PBN) [3], Information Theory Models (ITM) [4]. DBN and PBN are time consuming and ITM cannot give direction of regulation relationship. State Space Model (SSM) [5-9] can conquer those difficulties. As discussed in [10], SSM can have almost the same inference accuracy compared with DBN and has much less computational time. However, SSM still has its disadvantages. First, it needs to guess the initial values of parameters to start the iterations [7-9]. Second, it is hard to determine the gene-gene interactions in SSM. Rangel introduced an extra direct gene-gene interaction term and used bootstrap analysis to infer GRN in SSM [7]. The bootstrap analysis can slow down the speed of SSM, which makes the advantage of SSM insignificant, compared with other methods. Osamu proposed how to extract network from the standard SSM, but it still used permutation method to get the matrix of network, which needs many repeated calculations on the permutation datasets [8]. Kojima introduced a variation of SSM, which does not have observation matrix; but it still used an iterative method and needed to guess initial value of parameters [9]. Wu introduced the probabilistic principal component analysis (PPCA) to compute the values of hidden variables first, which solves the problem of determining initial values of parameters [5, 6]. However, it does not address how to infer the network. Holter used the Linear Model to obtain module-module interactions instead of GRN [11]. In this paper, we combine the advantages of PCA [16] and our previous work in SSM [10] and develop a simplified linear model (SLM) for GRN reconstruction, in which we first use PCA to compute the values of hidden variables and observation matrix, then use standard Linear Model to fit the transformed data. After fitting process we use the similar equation in [8, 15] to approximately extract the network without repeated calculations. Twenty synthetic networks generated by GeneNetWeaver [12] are used to demonstrate the performance of SLM. In addition, this combined method is simpler and faster, compared with standard SSM.

2 METHODS

2.1 State space model

SSM [5-9] has two kinds of variables; one is hidden variable x_t and the other one is measurement variable y_t. x_t and y_t are vectors, standing for values at t-th time point. The length of x_t and y_t are m and k, respectively. y_t is the measurement of microarray data in time point t. Since x_t is hidden variable, its value cannot be directly measured. The purpose of introduction of hidden variables is to reduce the number of parameters in this model [5-9]. If the length of hidden variable m is less than observed variable k, the number of parameters is decreased compared with model which does not introduce hidden variables, such as Linear Model.

The equations of SSM are,

$$x_t = Fx_{t-1} + w_t, \quad (1)$$

$$y_t = Hx_t + v_t. \quad (2)$$

Both equations are linear. The first equation describes the transition of hidden variable x_t where w_t is Gaussian noise. F is the matrix introduced to let x_t be the linear combination of x_{t-1}. The second equation describes the relationship between hidden variable x_t and observed variable y_t where v_t is Gaussian noise. y_t is linear combination of x_t. The transition equation (1) has less number of parameters compared with Linear Model since the length of x_t is less than the length of y_t. Since x_t is hidden variable, how to determine the optimum length of x_t becomes a problem. As discussed in [10], the length m should set to be a small integer number, since if m is too large, the purpose of reducing number of parameters will fail and the accuracy will be poor.

The inference process is complicated, and the analytic solution is usually not available. One will use an iterative Expectation-Maximization (EM) method [13, 14] to infer parameters. The EM method needs initial values of parameters to start iteration and it is also time consuming. After inference process, the H and F matrices can be obtained. What we intend to discover is the relationship between genes; or more specially, the relationship between y_t and y_{t-1}. After simple derivation, one can get the approximate equation,
$y_t = HF(H'H)^{-1}H'y_{t-1}$. So $C = HF(H'H)^{-1}H'$ can be used to determine inferred GRN and compared with realistic network [8, 15]. The value of each entry in the matrix C can be considered as the possibility of having a connection [9, 15].

2.2 Simplified linear model

The SLM can be described by the following equations:

$$y_t = Hx_t \quad (3)$$

$$x_t = Fx_{t-1} + \Omega + \sigma_t, \quad (4)$$

where x_t and y_t are vectors, representing the real and observed gene expression levels at time point t. H is a transformation matrix, obtained when PCA is applied before fitting the linear model (4). F is transition matrix, obtained by fitting the linear model (4) with x_t. Ω is a constant vector, having the same length as x_t. σ_t is Gaussian noise.

PCA is used to transform the original data y_t to x_t. x_t has the same length k as y_t. This step does not introduce any loss of information. However, the first few components of x_t are the most important for reconstruction of original data y_t. So, if we only retain the first few components of x_t, then we successfully reduce the number of parameters. After the data dimension reduction process, the usual fit of Linear Model is applied to x_t.

Equations (3) and (4) are similar to equations (1) and (2) in SSM. The difference is that matrix H is not a parameter in SLM; it can be determined by using PCA. Now the analytic solution of fitting linear equation (4) is available and one does not need to use iterative method anymore. Another advantage with analytic solution is that there is no need for initial values.
Similar with SSM, the approximate relationship between y_t and y_{t-1} is, $y_t = HF(H'H)^{-1}H'y_{t-1} = HFH'y_{t-1}$ [8, 15]. The second equation uses the fact that H is orthogonal matrix in SLM obtained using PCA. The network is a simpler equation compared with SSM; it does not include the inversion operation of matrix. Moreover, in SLM H is no longer a parameter matrix; it is only determined by the data. Since SLM has fewer parameters than SSM, the inference process is simpler and it is faster than SSM.

3 Results And Discussion

We use synthetic data to compare SLM and SSM methods. The advantage of synthetic data is that the true GRN is known so we can compare the inferred GRN with the true GRN. The number of time points can be as large enough so we can investigate the influence of number of time points on inference accuracy. The software used to generate synthetic data is GeneNetWeaver [12]. We generated 20 GRNs with 30 genes and 501 time points from *Ecoli* dataset. We used those two methods to infer the 20 GRNs and compare the average accuracy. The definition of accuracy is the number of correctly inferred edges over the number of totally inferred edges. The number of edges retained in inferred GRN is set as the same as true GRN for those two methods. The number of edges in true GRN is usually between 30 and 60 in our examples.

As mentioned before, equation (3) describes the data dimension reduction process by applying PCA. We use some examples to investigate if we can approximately reconstruct original data by only retaining the first few components of x_t. Fig. 1 is an example including both reconstructed and original data. The number of most important components retained in x_t is 2. It is smaller compared with the total number of genes, 30. One can see that, the most important structure, which is suddenly decreasing at around 250-th time point, is still reserved. At the same time, the noise level of reconstructed data is less than the original data. This is understandable, since less important components of x_t contain many high frequencies information. Here the number of remained components of x_t is set as a small number 2. The effects of different numbers on accuracy are discussed later.

Now we set the length of hidden variables x_t as 2 for both SSM and SLM and use this setting to test the reconstruction of 20 networks. The average accuracies for SSM and SLM are 10.8% and 9.2%. Since the average accuracy of random guess is around 4.6%, SSM and SLM can capture some truly interconnections of GRN from time series dataset. The accuracy of SSM is slightly higher than SLM, but the total calculation time for SSM and SLM is 471 seconds and 11 seconds, respectively. SLM is much faster than SSM. Since the convergence of SSM depends on the convergent criteria and initial values,

the above calculation time of SSM is only a typical one; it may vary.

Fig. 2 shows detailed accuracies of SSM and SLM with respect to 20 different networks using the data with all 501 time points. In some cases, SSM performs better than SLM while in other cases SLM is better. While the performance is dependent on the specific datasets, the overall performance of SSM is slightly better than SLM.

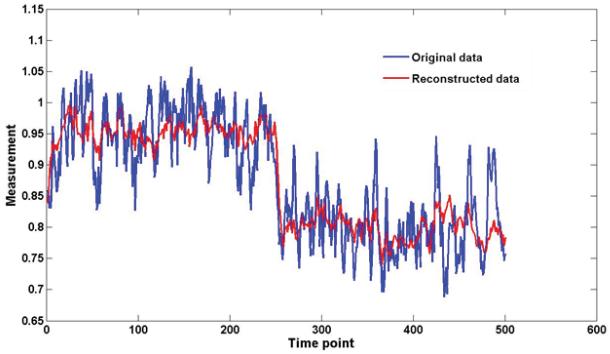


Fig. 1 The original data with 30 genes and 501 time points is depicted in blue line. The reconstructed data with only retaining 2 most important components of x_t after usage of PCA is depicted in red line.

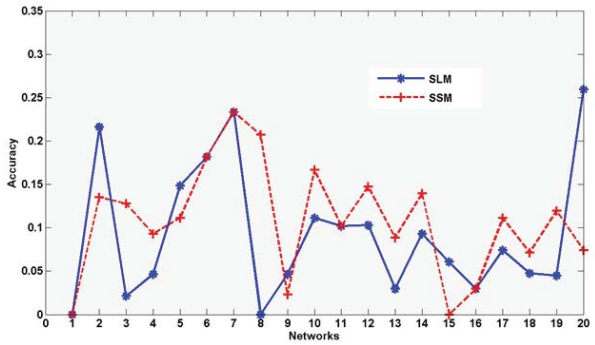


Fig. 2 The detailed accuracies of 20 networks with 30 genes and 501 time points by using SSM and SLM. The length of hidden variables in both SSM and SLM are set as 2.

In reality the measurement of 501 time points is not available. Usually there are only dozens of time points. There is a need to investigate the impact of different number of time points on the accuracy of GRN reconstruction. These time points are indexed as 1, 2 ... 501. We can choose a subset of data that are uniformly distributed among the original data. For example, one subset of data is chosen from the original data with time points 1, 11 ... 491, 501 so that the length of interval is 10.

The data with larger intervals have the same noise level which is how GeneNetWeaver generated those data [12]. We use different length of intervals from 1-30 to obtain 30 different datasets. The sizes of these datasets are from 501 to 17 time points. The GRNs are inferred using these datasets with different numbers of time points. For each dataset, the average accuracy is calculated based on the 20 networks. Fig. 3 shows the inference results (average accuracy of 20 networks) of SSM and SLM with respect to different numbers of time points. For datasets (1, 2, 3) whose sizes are from 501, 251 and 167 time points, SSM performs better than SLM. For datasets (4 - 17), whose sizes are from 126 to 30 time points, SLM performs better than SSM. For datasets (18 – 30), whose sizes are from 28 to 17, SSM performs better than SLM. For both SSM and SLM, larger sizes (number of time points) of dataset do not always lead to better performance. This is interesting, and we argue that this is because those two models are linear, so a larger dataset lets the fitting process more unreliable. In the same time the dataset with small size may cause the algorithm not to converge. So measuring a reasonable size of data is important. In our examples, more than 30 time points will be appropriate for inferring a network with 30 genes.

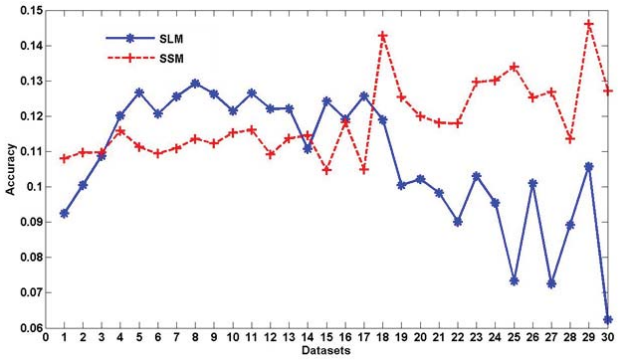


Fig. 3 The average accuracies over 20 networks by using SSM and SLM. Every dataset has 20 networks with different number of time points. The length of hidden variables in both SSM and SLM is set as 2.

In above examples, we let the length of x_t equal to 2 for both SSM and SLM. Now we investigate the average inference accuracies for setting length of x_t as different numbers. We choose an appropriate size of dataset, which has 101 time points to test. Fig. 4 shows that, for most settings of length of x_t, SLM performs better than SSM. The average accuracy of SLM is stable when length of x_t changes. In contrast, the average accuracy of SSM decreases fast when the length of x_t becomes larger. That is understandable since a larger length of x_t means a larger number of parameters; and the convergent results of SSM depend on how to guess initial values. A larger number of guessed initial values increase the difficulty of inferring true GRNs.

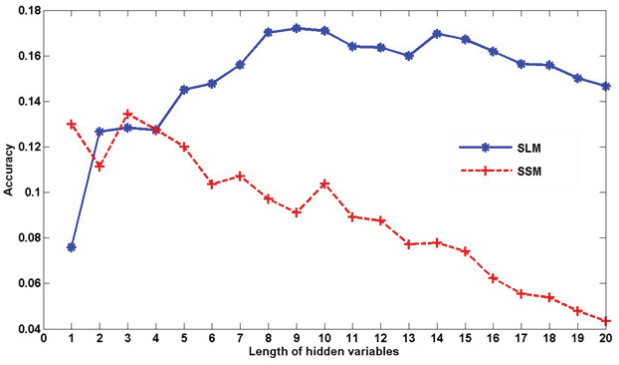


Fig. 4 The average accuracies over 20 networks with 101 time points for different length of hidden variables using SSM and SLM.

Fig. 5 gives the same results as Fig. 4, but uses all 501 time points. When one uses all 501 time points, the average accuracy of SLM is lower and it decreases fast like SSM. We argue that this is because the time interval of 501 time points is too small; and the noise added to the measurement has the same distribution, no matter what the length of time interval is. That implies that the time interval is not large enough so the measurement does not have obvious difference compared with the previous time point. After adding noise to those consecutive measurements without obvious differences, the dataset seems to show more randomness.

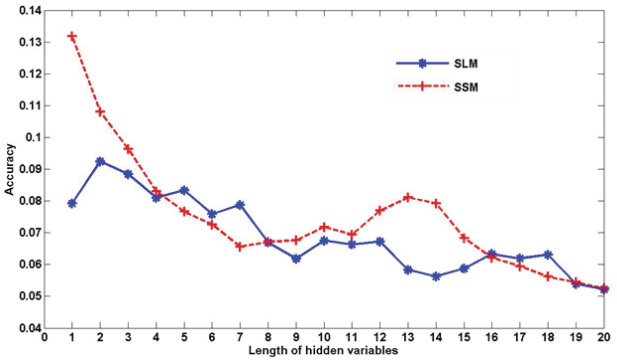


Fig. 5 The average accuracies over 20 networks with 501 time points for different length of hidden variables using SSM and SLM.

We use datasets with different time intervals investigate if it necessary to use PCA to reduce the dimension of a dataset. Without using PCA the SLM becomes standard Linear Model. Since the overfitting problem also exists in Liner Model, we use datasets with different time intervals to obtain a more comprehensive result. It is similar to the test in Fig. 3. The difference is that the length of x_t in Fig. 3 is 2, and here the length is set as 30 without using PCA. Only the first 15 datasets with a larger number of time points work, since the number of parameters is large when the length of x_t equals to 30 and more data is needed for successful inference.

Fig. 6 shows that, due to the large number of parameters, the average accuracy is really low even if the solution of Linear Model exists. For most of the datasets, the average accuracy is even below the accuracy of a random guess. The reason may be that the noise in the data let the fitting process have large bias, which implies that the introduction of PCA is necessary and useful.

Our previous work demonstrates that SSM is much faster and has almost the same inference accuracy compared with DBN [10]. However, since the equations in SSM are complicated, the inference process of SSM is difficult. That means we can only use iterative method to get an approximate result [7-9]. An initial guess of parameters is needed for iterative methods, and an inaccurate initial guess may lead to an inaccurate inference result. Due to these reasons, SLM was introduced to reduce the dimension of measured variables for inferring GRNs.

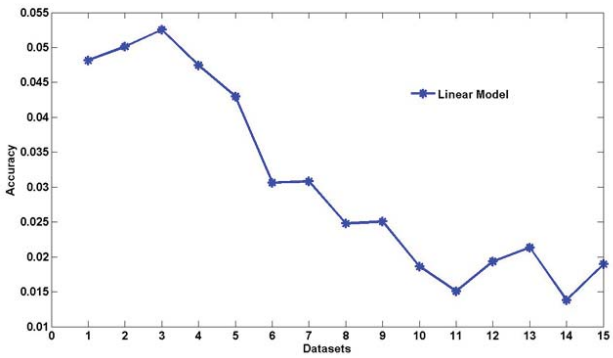


Fig. 6 The average accuracies over 20 networks by using Standard Linear Model. Every dataset has 20 networks with different number of time points.

4 CONCLUSION

SLM uses PCA to reduce the size of dataset first before inference calculation. The usage of PCA reduces the number of parameters and noise level in datasets. In SLM the observed matrix H is no longer a parameter. It is determined only by dataset itself. The analytic solution of SLM exists, so there is no need to guess an initial value of parameters, which results in a better convergence property. The results show that, if the number of time points is appropriately chosen, SLM can give a result with compatible accuracy. SLM is less sensitive to the change of the length of hidden variables than SSM. The average accuracy of SSM decreases fast when the length of hidden variables grows, but SLM almost gives the same average accuracy for a large range of lengths of hidden variables. At last, the need of using PCA was investigated. The result shows that without PCA, only using standard Linear Model cannot give a good performance. Moreover, SLM is much simpler and faster than SSM.

5 ACKNOWLEDGMENT

This work was supported by a National Science Foundation award (EPS 0903787) and an intramural grant from the US Army Environmental Quality/Installation (EQ/I) Basic Research Program to PG. Permission was granted by the Chief of Engineer to publish this paper.

6 REFERENCES

[1] K. Murphy and S. Mian, "Modeling gene expression data using dynamic Bayesian networks," Technical Report, Computer Science Division, University of California, Berkeley, CA 1999.

[2] M. Zou and S. Conzen, "A new dynamic Bayesian network (DBN) approach for identifying gene regulatory networks from time course microarray data," Bioinformatics, 2005, vol. 21, pp.71-79.

[3] F. Shmulevich, E. Dougherty, S. Kim, and W. Zhang, "Probabilistic Boolean networks: A rule-based uncertainty model for gene regulatory networks." Bioinformatics, 2002, vol. 18, no. 2, pp.261–274.

[4] V. Chaitankar, P. Ghosh, E. J. Perkins, P. Gong, and C. Zhang, "Time lagged information theoretic approaches

to the reverse engineering of gene regulatory networks," BMC Bioinformatics, 2010, 11(Suppl 6):S19.

[5] F. Wu, W. Zhang, and A. Kusalik, "Modeling gene expression from microarray expression data with state-space equations," Pac. Symp. Biocomput, 2004, 9, pp.581–592.

[6] F. Wu, "Gene Regulatory Network modelling: a state-space approach," Int. J. Data Mining and Bioinformatics, 2008, vol. 2, no. 1, pp.1–14.

[7] C. Rangel, J. Angus, Z. Ghahramani, M. Lioumi, E. Sotheran, A. Gaiba, D. Wild, and F. Falciani, "Modeling T-cell activation using gene expression profiling and state space modeling," Bioinformatics, 2004, vol.20, no. 9, pp.1361–1372.

[8] O. Hirose, R. Yoshida, S. Imoto, R. Yamaguchi, T. Higuchi, D. Charnock-Jones, C. Print, and S. Miyano, "Statistical inference of transcriptional module-based gene networks from time course gene expression profiles by using state space models," Bioinformatics 2008, vol. 24, pp.932–942.

[9] K. Kojima, R. Yamaguchi, S. Imoto, M. Yamauchi, M. Nagasaki, R. Yoshida, T. Shimamura, K. Ueno, T. Higuchi, N. Gotoh, and S. Miyano, "A state space representation of VAR models with sparse learning for dynamic gene networks," Genome Inform, 2009, vol. 22, pp.56–68.

[10] X. Wu, P. Li, N. Wang, P. Gong, E. Perkins, Y. Deng, and C. Zhang, "State Space Model with hidden variables for reconstruction of gene regulatory networks," BMC Systems Biology, 2011, 5(Suppl 3):S3.

[11] N. Holter, A. Maritan, M. Cieplak, N. Fedoroff, and J. Banavar, "Dynamic modeling of gene expression data," Proc. Natl Acad. Sci. USA. 2001, vol. 98, pp.1693–1698.

[12] D. Marbach, T. Schaffter, C. Mattiussi, and D. Floreano, "Generating realistic in silico gene networks for performance assessment of reverse engineering methods," Journal of Computational Biology 2009, vol. 16, no. 2, pp.229-239.

[13] C. Bishop, "Pattern Recognition and Machine Learning." Springer 2006.

[14] A. Dempster, A. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," Journal of the Royal Statistical Society, 1977, Series B, 39(1), pp.1–38.

[15] P. Li, "Inferring Gene Regulatory Networks From Time Series Microarray Data," Ph.D. dissertation, The University of Southern Mississippi, 2009.

[16] S. Raychaudhuri, J. Stuart, and R. Altman, "Principal components analysis to summarize microarray experiments: application to sporulation time series." Pac. Symp. Biocomput, 2000, pp.455–466.

KLAST: fast and sensitive software to compare large genomic databanks on cloud

I. Petrov[1], S. Brillet[1], E. Drezen[1], S. Quiniou[2], L. Antin[2], P. Durand[2] and D. Lavenier[1]
[1]INRIA/IRISA/GenScale, Campus de Beaulieu, 35042 Rennes cedex, France
[2]Korilog, 4, rue Gustave Eiffel, 56230 Questembert, France

Abstract - *As the genomic data generated by high throughput sequencing machines continue to exponentially grow, the need for very efficient bioinformatics tools to extract relevant knowledge from this mass of data doesn't weaken. Comparing sequences is still a major task in this discovering process, but tends to be more and more time-consuming. KLAST is a sequence comparison software optimized to compare two nucleotides or proteins data sets, typically a set of query sequences and a reference bank. Performances of KLAST are obtained by a new indexing scheme, an optimized seed-extend methodology, and a multi-level parallelism implementation. To scale up to NGS data processing, a Hadoop version has been designed. Experiments demonstrate a good scalability and a large speed-up over BLAST, the reference software of the domain. In addition, computation can be optionally performed on compressed data without any loss in performances.*

Keywords: Cloud computing – bioinformatics – Intensive sequence comparison – High Throughput Sequencing

1 Introduction

The extraordinary progresses of sequencing technologies currently lead to generate very huge amount of data. Processing these data are performed by specializing bioinformatics pipelines that depend of the biological question to answer. Such pipelines include many specific bioinformatics tools but also – most of the time - intensive sequence comparison steps to extract similarities between raw sequencing data and fully annotated DNA or protein databanks. Comparing large sequence datasets to genomic banks (DNA or protein) can thus be extremely time-consuming, especially if a minimum of sensitivity is required.

Speeding up the comparison process can be done using different directions. A very efficient heuristic is the seed-extend approach that first detects similar *seeds* (i.e. short words) between query and genomic databank, and then performs left and right extensions to generate relevant alignments. BLAST family software [1], RAPSEARCH [14] or DIAMOND [2] are based on this concept. Another possibility is to count the number of identical short words in common between two sequences and to decide whether it's worth to continue the search according to a threshold value. USEARCH [5] follows this interesting strategy.

The exploitation of internal parallelism of modern processors is also a convenient way to gain performances: processors now include several cores that can be simultaneously activated through multithreading. They also integrate powerful instruction sets allowing basic comparison operations to be simply vectorized. SSEARCH [6] and SWIPE [11] makes an intensive use of this technique to produce a very efficient fine-grained parallelization of the Smith and Waterman algorithm based on dynamic programming [13]. Algorithm optimization and parallelization techniques can of course be mixed together to improve efficiency.

KLAST has been primarily designed to compare a query set of sequences against a DNA or protein databank. It follows the BLAST strategy by proposing all the possible query/database combinations: DNA/DNA, DNA/protein, protein/DNA, protein/protein, DNA-translated/DNA-translated. Performances of KLAST are obtained by a new indexing scheme, an optimized seed-extend methodology, and a multi-level parallelism implementation (multithreading and vectorization). Tuning the seed-extend heuristic of KLAST allows the users to precisely define the sensitivity/speed tradeoff.

Compared to BLAST (the gold standard of search alignment tools), and considering equivalent sensitivity, speed-up ranges from 5 to 10 according to the nature of the datasets, and to the amount of data to process. For very large problems, this speed-up is far from negligible. It can save million hours of computation and significantly reduce the number of nodes in a cloud infrastructure.

The KLAST cloud implementation highlights this aspect. Similar to existing BLAST cloud solutions [8][10] our implementation provides efficient scalability thanks to the nature of the sequence comparison problem that is an embarrassingly parallel problem. Distributing the computation on a cluster, especially for this specific problem, is thus straightforward and doesn't present any theoretical difficulties. On the other hand, scalability can be limited by data accesses, I/O transfers or sequential sections of the algorithms. Cloud implementation must consequently be

highly optimized on these points to maintain a good scalability.

The KLAST cloud implementation also addresses the problem of managing large data files. The genomic banks represent hundred of Giga bytes of data. They have to be stored near the hardware computing resources to make them available when a job is run. They also have to be efficiently routed through the cloud network to the computing nodes. The first point requires consequent space storage. The second point can be critical for I/O bound problems. Working on compressed data, as proposed in the KLAST cloud implementation, is a way to release the pressure when tackling big data domains.

The next section briefly describes the KLAST methodology and provides a performance overview compared to other software. Section 3 is dedicated to the cloud implementation of KLAST with Hadoop. Section 4 concludes the paper.

2 KLAST SOFTWARE

KLAST is issued from PLAST, a sequence comparison software previously developed for protein alignments only [9]. It has been extended to DNA comparison by adapting the ORIS algorithm [7] to the double indexing seed scheme. Furthermore, for nucleotides search, it includes a new filtering step strategy that makes KLAST very competitive to process NGS data.

The global KLAST strategy, for protein or DNA sequences, is based on the following steps:

1. *Seed indexing*: In this step, the query and the subject bank are both indexed. For performance purpose, the two indexes completely fit into the computer memory. However, for very large banks, a bank splitting is automatically performed.
2. *Ungap alignment search*: This step acts as a filter to limit the search space. The idea behind this filtering step is that if no similarity is found in the immediate neighborhood of the seed, then the probability to find a significant alignment in this region is low, and won't necessitate further processing.
3. *Gap alignment search*: The ungap alignments calculated on the previous step are extended to include gap errors. This is done by dynamic programming technique over a restricted search area.
4. *Alignment sorting*: The multi-threaded implementation of the steps 2 & 3 generate alignments in a random order. They need to be reordered before to be displayed.

Depending of the nature of the data (protein or DNA), the methodologies used in step 1 and 2 are different. The two next subsections explain the way indexing and ungap search are implemented according to the data type.

2.1 Seed indexing

Protein sequences: A subset seed model is used [12]. Such seeds are more convenient than standard seeds for indexing purpose [9]. The protein index is an array that stores all the positions of the seeds in the banks.

DNA sequences: A conventional seed model is used (words of N consecutive nucleotides). By default, a seed size of 11 is used. Unlike the protein index that only store the seed positions, the DNA index also memorizes neighborhood information: 3-mers that are present in the left and right neighborhood of 20 nucleotides are tagged into two 64-bit set vectors.

2.2 Ungap alignment search

As both banks are indexed in the same way, the ungap alignment search step consists in performing a loop over all possible seeds. For a given seed, the indexes of the two banks provide two lists of positions from which an all-vs-all computation is done. More precisely, each element of one list is compared to all elements of the other one. In this step, gap errors are not allowed.

Protein sequences: A search over a fixed size area near the seeds is performed. This predefined search region allows computation to be parallelized with SIMD instructions. 16 8-bit alignment scores are simultaneously computed.

DNA sequences: This step is split into two tasks. First, logical operations between the two 64-bit vectors embedded in the index are performed to determine the number of identical 3-mers. If it overcomes a threshold value, then an ungap search is launched using the ORIS algorithm. This algorithm computes a score by a left and right extension starting from the seed, but returns only an alignment if this extension doesn't use words smaller than the seed. See [7] for a complete description of the algorithm.

2.3 Parallelism

The KLAST algorithm exhibits three levels of parallelism. The first one is linked to the capacity of KLAST to split the banks into chunks of data that are fully indexed into the computer memory. These features have two main advantages:

1. Huge computer memory configurations are not required to process large genomic banks. The computation is performed sequentially on each chunk of data.

2. Pieces of banks can be dispatched and independently processed over a grid or a cluster infrastructure. The merging step is cheap and doesn't penalize the overall performances.

The second level of parallelism comes from the double indexing approach. A large number of seeds can be analyzed

simultaneously using the multithreading possibilities of today multi-core processors. The programming model is a producer /consumer model. One thread manages the overall computation and request many threads to compute ungap and gap alignments.

The last level of parallelism is the use of the SIMD paradigm for very "regular" computation. For protein sequences it is intensively used in the ungap and gap alignment search. The ungap step parallelizes 16 computations of scores simultaneously. The gap search step, which is more complex, run only 8 score computations in parallel.

2.4 Performances

This section provides a brief overview of KLAST performances, both in terms of quality and speed-up compared to other similar software. In the context of NGS data, we have compared a subset of a RNA-seq dataset (50K Illumina sequences) from a microbiome project with a Human proteome databank (71,338 proteins). Experiments have been conducted on a 2.67 GHz Intel Xeon E5640, 8 cores, 48 GBytes of RAM, Debian 4.6.3-13 Linux version. The following software, which handle the comparison of DNA sequence with protein sequences, are considered: BLAST, UBLAST, DIAMOND and KLAST.

Table 1 reports the results. The Align column represents the number of alignments found by the software. Two alignments are considered as identical if the two sequences overlap at 80% [4]. The Hit column is the number of query-subject pairs reported without any control on the alignment boundaries. It just tells that a specific DNA sequence has a significant match with a specific protein. This type of information can be sufficient to answer many biological questions.

	Align	Hit	Exec. Time
BLAST	2934080	465376	2539 sec.
UBLAST	308826	221551	65 sec.
DIAMOND	968886	402211	44 sec.
KLAST	2902727	463934	283 sec.

Table 1: Nucleic/Protein search (e-value = 10-3)

From a quality point of view, BLAST and KLAST generate similar results. Differences come from the search methodologies that are not exactly identical. But both rely on heuristics and are statistically equivalent [9]. KLAST is however much faster (speed up = 9). UBLAST and DIAMOND are very fast but only 10% and 35% of the alignments are reported, respectively.

If less sensitivity is required, KLAST can be tuned to reach DIAMOND sensitivity. The tuning is performed by considering only a subset of seeds, from 100% to 1%. In that case, the execution time of KLAST significantly decreases. DIAMOND is generally 50% to 100% faster but requires computers with a large memory. As a matter of fact, DIAMOND speed comes from a sophisticated – but costly – indexing scheme that absolutely needs to fit into memory

3 KLAST on CLOUD

3.1 Parallelization Strategy

The problem of finding alignments between two sets of sequences is embarrassingly parallel. If N and M are respectively the size (in terms of number of sequences) of the query set and the reference bank, then we have to solve NxM independent problems. N and M can be very large (a few tens of millions), leading to 10^{14} to 10^{15} elementary tasks that could be ultimately processed in parallel. As an example, a NGS data set may represents 10^8 sequences of length 100 (10 GB) and the non-redundant uniprot protein bank contains 92 x 10^6 proteins (~35 GB).

The strategy, here, is to process independently chunks of sequences, and to run KLAST on these data. The query set and the reference bank are thus split into packets of a few mega bytes. Each run of KLAST generates a list of sorted alignments, which are pushed to the storage system.

The final step is to reorder the lists of alignments. Actually, depending of the downstream processing, this task may not be essential. Hence, it is optionally done when reading back the results.

3.2 Hadoop Implementation

The task of processing a large number of independent chunks of data can be solved efficiently using a Map-Reduce approach [3]. The most famous open source implementation of this approach is Hadoop (http://hadoop.apache.org/). The Hadoop strategy allows very good job scheduling on a large number of nodes to be done very efficiently in regard of many aspects: physical location of the data, good utilization of the hardware resources, High Availability (HA), dynamic modification of available nodes, etc.

Here, we have to process NxM chunks of data, each chunk being a set of sequences that do not have exactly the same size. Furthermore, for a specific job, Hadoop manages only the split of a single file among all the cluster nodes. Thus, our strategy is the following:

- The subject bank is managed by the Hadoop splitting mechanism. N is the number of chunks.
- The query bank is split independently into M sub query banks, and M Hadoop jobs are launched.

As Hadoop splits data into chunks of identical size, an additional step is required when the data is being read. It

consists of a special treatment of the first and the last sequence in each chunk, so that they are exactly read once, even when a part of any of them is placed in one split and another part of it is placed in the next chunk. One solution, for a chunk, is to skip the first partial sequence and ensure that it is read as part of the previous chunk. In that way, all sequences are considered. Hadoop provides internal features to deal with this kind of data adjustment.

As previously mentioned, a Hadoop job handles the splitting of only one single file. For that reason, the query bank is cut into sub banks of nearly identical size. It is then possible to force Hadoop to place specific files on each node, and to run a Hadoop job for each query split. In this way, both the subject and the query are split and all the pairs of query and subject chunks are processed independently. In order to improve further the nodes utilization, two jobs are started at the same time on each node. In that way, the I/O and scheduling times for one comparison are used for calculations of the others.

Klast computes an e-value for each alignment. This e-value depends of the size of the subject bank. The e-value computation based on a reduced section of the subject bank will lead to incorrect statistical information. To avoid such a situation, and to output identical results compared to a sequential execution, the real size of the subject bank is considered through the use of a specific parameter of KLAST that is set automatically by our implementation.

The actual computation is performed in the Hadoop mappers. They execute a KLAST command with the appropriate arguments and save the output on the Hadoop distributed file system. If special merging of the results is needed, it is performed in a single Reducer task.

Having the databases in raw fasta format may take a considerable amount of space storage and may also take time during data transfers. To make these aspects less constraining as possible, data can be uploaded in a compressed format. When this option is selected, bz2 is currently used for storing the data. Fortunately, this is a splitable format that can benefit of the parallel cloud environment. We are looking into adding other splitable archive formats, as bz2 is computationally expensive in decompress mode. As Hadoop can support a number of archive formats this is not expected to present any difficulty.

3.3 Experimentation

Tests have been performed on the IFB (French Institute of Bioinformatics) cloud. The virtual CPU cores are simulated by qemu and report a working frequency of 2.6 GHz. 2 or 4 core nodes configuration have 8 GB RAM, whereas 8 core configuration have 16 GB RAM. The OS is CentOs 6.5.

We first tested the scaling of KLAST by comparing the full yeast proteome (6640 proteins, 3.2 x 10^6 amino acids) with the NCBI nr protein database (release 67, ~61.3 x 10^6 sequences, 18.7 x10^9 amino acids).

Different nodes/CPU configurations have been tried. Figure 1 plots the speed-up of KLAST in the 2 CPU/node configurations ranging from 1 to 40 nodes. The 4 or 8 CPU/node configurations provide similar performances (same scaling and same execution time: ~1h30).

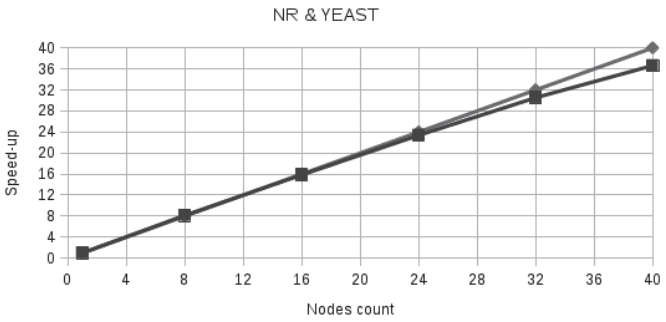


Figure 1: KLAST speed-up on the IFB cloud with 2 CPU per node. The Yeast proteome (6640 proteins is compared with the NCBI nr protein databank (61.3 millions of proteins)

As it can be seen, the speed-up is nearly linear, allowing an execution time of 60 hours on one 2 CPU node to be lower down to 1h30 with 40 nodes. Further experimentations with a higher number of nodes would be necessary to better analyze the scalability of the KLAST Hadoop implementation.

We also compare the parallel BLAST and KLAST Hadoop implementation. Instead of running KLAST, BLAST is simply launched with the same splitting parameters. The following table reports the execution times. On average, KLAST is about 7 times faster and the speed-up tends to increase with the number of nodes.

#nodes	**10**	**20**	**40**
BLAST	35h35	19h40	10h58
KLAST	5h08	2h45	01h29
Speed-up	6.9	7.2	7.4

On this specific experiment, if the data compression mode is activated, the overall processing time remains the same. Actually, the time for transferring the data is balanced out by the time for decompressing the data. The great advantage is that a lot of space is saved without any loss in performances. The size of ncbi nr bank is 36.3 GB. Its compressed version with bz2 is 13.2 GB. Hence, on that example, 23 GB is saved.

4 Conclusions

KLAST is a bank-to-bank comparison software designed and optimized to process large genomic and metagenomic data sets. Like BLAST, KLAST is based on the *seed-and-extend* heuristic strategy, but includes various improvements that favour NGS data processing. It also exploits all parallel resources of modern computers (multi-cores and aggressive use of SSE instruction set) that allow execution time to be significantly reduced.

A KLAST Hadoop version has been designed to tackle large sequence comparison problems, i.e. problems requiring millions hours of CPU time. The current implementation, over concurrent approaches, mainly bring two advantages:

- *Significant speed up* compared to the gold standard of the domain (BLAST) and a good scalability. An average speed up of 7 if often measured if equivalent sensibility to BLAST is wanted. If a loss of sensibility is accepted, then much high speed up is achieved (X50). For huge instances of genomic sequence comparison problems, and from an economic point of view (in a cloud context), the efficiency of KLAST may save a lot of time and money.

- *Efficient storage* through the direct use of compressed data. Cloud services dedicated to the bank-to-bank comparison have to propose a large panel of public banks. Having the banks in a compressed format save both storage space and communication bandwidth within a cluster.

Short-term perspectives are to increase scalability measurements. We are currently limited by the IFB cloud infrastructure that is still in its starting phase, but that should progressively grow to 10,000 cores in the next 2 years.

Longer-term perspectives are to split the reference banks in a much clever way. The current splitting generates raw chunks of data independently of any knowledge that could be extracted from the genomic banks. Actually, banks content a lot of redundancy coming from the orthologous nature of the sequences. On the other hand they contain a huge diversity that makes the search computationally intensive: a query often matches with a very tiny fraction of sequences of the reference bank, meaning that exploring the entire search space could be avoided.

Similarly, in the case of a NGS query bank, a pre-processing step to exploit the coverage redundancy would be an efficient way to reduce the complexity of the problem. A solution is to assemble the NGS data into contigs and then to perform a comparison between these contigs and the reference bank. In that way, the number of query sequences could be significantly reduced, and similar computation avoided.

KLAST AVAILABILITY

KLAST is jointly developed by the INRIA/IRISA GenScale research team and the KORILOG Company through a common research lab (KoriScale Lab). A free academic version is available and can be downloaded at: http://koriscale.inria.fr/klast-download

ACKNOWLEDGMENT

The authors would like to thank the IFB (Bioinformatics French Institute) for the availability of computing resources. *Funding*: This work is supported by the ANR (French National Research Agency), ANR-14-CE23-0001-01 and by the Brittany Region (KoriPlast2 project).

5 References

[1] Altschul S.F. et al. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res., 25, 3389–3402.

[2] Benjamin Buchfink, Chao Xie & Daniel H. Huson, Fast and Sensitive Protein Alignment using DIAMOND, Nature Methods, 12, 59–60 (2015) doi:10.1038/nmeth.3176

[3] Dean J., Ghemawat S., (2005) MapReduce: Simplified Data Processing on Large Clusters, OSDI'04

[4] Drezen E, Lavenier D., (2014) Quality metrics for benchmarking sequences comparison tools, Advances in Bioinformatics and Computational Biology, LNCS 8868, pp.144-153

[5] Edgar R.C. (2010) Search and clustering orders of magnitude faster than BLAST, Bioinformatics 26(19), 2460-2461.

[6] Farrar M., (2007) Striped Smith–Waterman speeds database searches six times over other SIMD implementations, Bioinformatics (2007) 23 (2): 156-161.

[7] Lavenier D., (2008) Ordered Index Seed Algorithm for Intensive DNA Sequence Comparison, IEEE International Workshop on High Performance Computational Biology, Miami, Florida, USA

[8] Matsunaga A., Tsugawa M., Fortes J., (2008) Cloudblast: Combining mapreduce and virtualization on distributed resources for bioinformatics applications eScience, 2008. eScience'08. IEEE Fourth International Conference

[9] Nguyen V.H., Lavenier D., (2009) PLAST: parallel local alignment search tool for database comparison, BMC Bioinformatics, vol 10, no 329

[10] O'Driscoll A. et al., (2015) HBLAST: Parallelised sequence similarity – A Hadoop MapReducable basic local alignment search tool, Journal of Biomedical Informatics, Volume 54, April 2015, Pages 58–64

[11] Rognes T., (2011) Faster Smith-Waterman database searches with inter-sequence SIMD parallelisation, BMC Bioinformatics 12, 221

[12] Roytberg M, Gambin A, Noe L, Lasota S, Furletova E, Szczurek E, Kucherov G., (2009) On subset seeds for protein alignment, EEE/ACM Trans Comput Biol Bioinformatics 2009, 6(3): 483-494.

[13] Smith T.F., Waterman, M. S. (1981). Identification of Common Molecular Subsequences, Journal of Molecular Biology 147: 195–197

[14] Zhao Y., Tang H. and Ye Y., (2012) RAPsearch2 : A fast and memory efficient protein similarity search tool for next generation sequencing data, Bioinformatics (2012) 28 (1): 125-126

[15] Sul, S-J., Tovchigrechko A., (2011) Parallelizing BLAST and SOM algorithms with MapReduce-MPI library, 2011 IEEE International Parallel & Distributed Processing Symposium

Model-Driven Analysis of Gene Expression Data

Application to Metabolic Re-programming during T-cell Activation

Alyaa M Abdel-Haleem[1,2], Ayman F Abuelela[1], Victor Solovyev[1], Neema Jamshidi[*3] & Nathan E Lewis[*3,4]

[1] King Abdullah University of Science and Technology, Thuwal, Saudi Arabia.

[2]Computational Bioscience Research Centre, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia

[3] University of California San Diego, La Jolla, CA, USA.

[4]Novo Nordisk Foundation Center for Biosustainability at the University of California, San Diego School of Medicine, San Diego, La Jolla, CA, USA.

* Corresponding authors: Nathan E Lewis nlewisres@ucsd.edu & *Neema Jamshidi* neema@ucsd.edu

***Abstract*—For decades, metabolism had been appreciated to be intimately integrated with many physiological and disease processes. Recent advances in systems biology have leveraged this fact and used genome-scale metabolic networks to understand how mutations and changes in gene expression impact the whole cell or organism. In particular, constraint-based modeling methods systematize biochemical, genetic and genomic knowledge into a mathematical framework that enables a mechanistic description of metabolic physiology. In this perspective, we demonstrate how constraint-based model-driven analysis of transcriptomics data can advance our understanding of physiological processes, such as the metabolic reprogramming events involved with T-cell activation. By developing context-specific models for activated and naiive T-cells, the activated T-cell model predicted mitochondrial activation of glycerol-3-phosphate dehydrogenase (GPD2) as well as providing enzyme-efficiency related mechanistic insights into usage of amino acid transporters that take place upon T-cell activation while it shifts from a small resting cell to a rapidly cycling cell.**

Keywords—genome-scale models, transcriptomics, T-cell

I. INTRODUCTION

Biology is increasingly becoming a data-rich field. As such it has become an increasing challenge to organize, sort, interrelate, and contextualize all of the high-throughput datasets in a manner that would allow data interpretation and the elucidation of novel discoveries. In particular, it has become increasingly apparent that this process is greatly accelerated when carried out in the context of all the relevant prior knowledge [1].

In systems biology, both bottom-up and top-down approaches are central to assemble information from all levels of biological pathways that must coordinate physiological processes. A bottom-up approach involves draft reconstruction, manual curation, network reconstruction through mathematical methods, and validation of these models through literature analysis (i.e., bibliomics) [2, 3]. However, top-down approaches leverage metabolic network reconstructions and 'omics' data (e.g., transcriptomics, proteomics, fluxomics) using appropriate statistical and bioinformatics methodologies [4]. In top–down modeling, determination of network structure poses a major technological and computational hurdle. However, many challenges facing top–down modeling, such as confidence in the statistically inferred pathways, can be alleviated by the incorporation of carefully reconstructed bottom–up models. Genome-scale network reconstructions of metabolism are built from all known metabolic re actions and metabolic genes in a target organism [3]. Networks are constructed based on genome annotation, biochemical characterization, and the published scientific literature on the target organism.

Metabolic reconstructions contain all known metabolic reactions of a particular system. The reactions are charge and mass balanced, and linked to the enzymes catalyzing them through gene-protein-reaction (GPR) annotations. GPRs mechanistically connect the genome sequence with the proteome and the enzymatic reactions [5]. Thus, reconstructed networks, or an assembled reactome, for a target organism, represent biochemically, genetically, and genomically structured knowledge bases, or BiGG k-bases [6]. A network reconstruction can be converted into a mathematical format (constraint-based model or CBM) and thus, lend itself to mathematical analysis and computational treatment [2]. Network reconstructions have different biological scope and coverage. They may describe metabolism, protein-protein interactions, regulation, signaling, and other cellular processes, but they have a unifying aspect: an embedded, standardized biochemical and genetic representation amenable to computational analysis [2].

By serving as a framework on which other data types can be overlaid, metabolic reconstructions have served as a powerful tool for contextualizing high-throughput data and aiding top–down approaches [3]. Omics data have been used both to constrain calculated flux distributions and for comparison and validation of model predictions [7]. For example, this can be done by directly imposing bounds on individual reaction fluxes in the genome-scale network reconstruction, based on the values in the experimental datasets (e.g. gene expression data, protein expression data, or C13 flux data).

Despite a non-perfect correlation between gene expression and protein expression [8, 9] gene expression data can be used to constrain metabolic fluxes and provide insight into

condition-specific changes in metabolic activity or capabilities. One analysis goal of gene expression data experiments (i.e., RNA-Seq and microarray) is to compare expression levels between two conditions, e.g., stimulated versus un-stimulated or wild type versus mutant to identify genes that are significantly altered under certain condition. Such genes are traditionally selected based on a combination of expression change threshold and score cutoff, which are usually based on p values generated by statistical modeling [10]. The interpretation of RNA-Seq data, for instance, requires the definition of a computational pipeline that comprises several steps: read mapping, count computation, normalization and testing for differential gene expression [11]. However, turning huge and complex differentially expressed genes data sets into biologically meaningful findings is not trivial. Also, the number of differentially expressed genes is pipeline dependent and also false positives have been reported [12, 13]. Further, analysis of transcription data in general is often hindered by the low signal-to-noise ratio and by the limitation that post-transcriptional regulation is not captured in these data sets. These limitations can be ameliorated through contextualization of the gene expression data within the CBM framework.

When analyzing transcriptomics data, the changes of expression levels of some genes are disconcerted, i.e., transcripts are found within the same pathways but the changes in the direction of their expression level that are inconsistent with the activity of pathway. This might be caused by a false-positive measurement or by post-transcriptional regulation. Disconcerted transcripts in metabolic pathways can be tuned by overlaying them on the cellular pathways they are involved in to see how expression changes perturb the entire metabolic network. In this way, CBMs can be used to analyze complex omics data for an organism or tissue of interest, thus exposing different perspectives into how the expression of each gene contributes to the overall physiology of the cell [7, 14].

In this study, we demonstrate this process to study the metabolic reprogramming that occurs during T-cell activation. By integrating RNA-Seq data from naiive T-cells and activated T-helper (TH) cells, with the human metabolic network [15], Genome-scale metabolic network models of naiive and activated T-cells were developed. These cell-specific models allowed us to simulate the metabolic activities of each cell type in order to study the re-routing program that an inactivated T-cell follows on the way to become activated. Using activated and inactivated TH-cell-specific metabolic models, we were able to capture the role of glycerol-3-phosphate dehydrogenase (GPD2) in the TH-cell activation program while showing no difference in gene expression level between the naiive and activated states. The models also predicted enzyme-efficiency related mechanistic explanation for the altered amino acid transporters usage that comes into place as the cell progresses towards an active state. In addition, contextualization of the gene expression data shed some light on the potential similarities between the processes of tumorigenesis and T-cell activation. Overall, fostering the notion that model-driven analysis of gene expression data provides novel insights complex biological processes.

II. Application to Metabolic Reprogramming of T-Cell Activation

A. *Construction of naiive and activated TH-cell metabolic models*

Context-specific models for naive and activated T-helper cells (memory CD25$^-$ TH1 and TH2) were generated using the integrative metabolic analysis tool (iMAT) [16] and previously published gene expression data (GSE55320) [17]. iMAT was used to integrate the expression data with the human metabolic network (recon2.v04; http://humanmetabolism.org). Briefly, iMAT computes a flux distribution which best uses reactions that are associated with up-regulated genes and which avoids using reactions that are associated with down-regulated genes, thereby predicting differential reaction use between conditions. The output is a reduced model of the organism's metabolic state, showing the most likely predicted metabolic fluxes across its reactions [16]. Because iMAT is from the family of integrative methods that doesn't require a definition of a biological objective, the resulting models don't necessarily include the biomass objective function that was present in the starting model (recon2.v04 in this case). An objective function, such as the biomass objective function, details the composition of the cell and energetic requirements necessary to generate biomass content from metabolic precursors [18]. Hence, the biological objective function of the human metabolic network was added to the TH and naive-cell models. The minimum number of reactions required for maintaining growth in each network was added subsequently. This resulted in two models (Table 1) of different size and content for the TH- and naive cells.

TABLE 1. Summary statistics for the models

	TH-cell model	*Naiive-cell mdoel*
Reactions	4836	4848
Metabolites	3642	3634
Unique genes	129	133
Common Reactions	**4623**	

B. *Metabolic genes are classified by their use in cell-type-specific metabolic models*

To simulate and interpret the metabolic pathway usage of these models, we used parsimonious flux balance analysis (pFBA) [19], a bi-level linear programming optimization that uses FBA to compute the optimal growth rate followed by a minimization of the sum of all gene-associated reaction fluxes while maintaining optimal growth. This proxy computes the pFBA optima, representing the set of genes associated with all maximum-growth, minimum-flux solutions, thereby predicting the most stoichiometrically efficient pathways [19]. In addition to simulating pathway usage, pFBA classifies the metabolic genes into five classes, associated with reactions that (1) are essential for optimal and suboptimal growth 'essential', (2) are inside the pFBA optima 'pFBAoptima', (3) are ELE, requiring more enzymatic steps than alternative pathways that meet the same cellular need, (4) are 'MLE', requiring a reduction in

growth rate if used, or (5) cannot carry a flux in the given environmental condition/genotype (pFBA 'no-flux' or 'blocked') [19].

pFBA optima were computed for the TH-cell and naive-cell models. Genes were assigned to one of the five categories (essential, pFBA optima, ELE, MLE or blocked). The sets of pFBA genes were compared with the differential gene expression analysis (DEG) sets. The total number of metabolic genes that was accounted for in the RNA-Seq data was n = 1488. Of these, n = 360 were significantly differentially expressed (n = 235 significantly up-regulated & n = 125 significantly down-regulated) and n = 1128 were invariant.

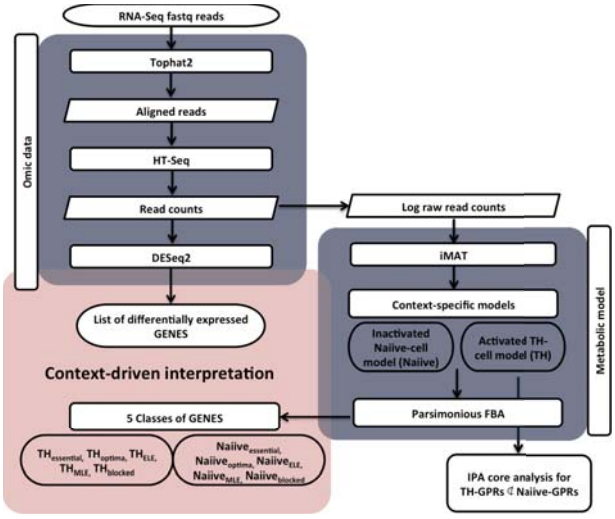


Fig .1. The implemented workflow introduced in this study to provide context for gene expression data analysis and interpretation. Archived SRA files were downloaded from Gene Expression Omnibus (GEO) database (GSE55320) and converted to fastq files using SRA Toolkit v2.4.2. Reads were mapped against human reference genome (hg19, GRCh37 downloaded from the UCSC genome browser) using TopHat v2.0.13 [20]. HTSeq v0.6.1p1 [21] was used to compute raw read counts which were passed to DESeq2 for differential gene expression analysis [22]. Log read counts for the activated and naiive T-cells were then used to generate context-specific models using iMAT and recon2 (v2.04). A modified version of pFBA [19] was then used to classify genes from each model for contextualizing the gene expression data. IPA (Ingenuity Pathway Analysis) software core analysis was then conducted on the list of genes that were uniquely included in the TH-cell model but not in the naive-cell model.

C. Models demonstrate differential activity of glycerol-3-phosphate dehydrogenase 2 (GPD2)

While 28% of the genes that were unique to the TH-cell model (reactions retained in iMAT-reduced TH-cell model while being excluded from naive-cell model) are significantly differentially expressed between the TH and naive cells (FDR adjusted p value < 0.05), several genes are predicted to be of higher importance to the activated TH-cell model while being non-differentially expressed according to the DEG data. Several of these have been previously implicated in T-cell activation. For instance, glycerol-3-phosphate metabolism is crucial for T cell activation [23] as demonstrated by a recent study reporting significant up-regulation of mitochondrial glycerol-3-phosphate dehydrogenase 2 (GPD2) upon T-cell activation [24]. While the DEG failed to detect the differential expression of GPD2, context-specific models predictions showed that GPD2-associated reaction is active in the TH-cell model but not in the naiive-cell model.

D. Metabolic efficiency explains transporters usage in TH-cell activation

Amino acids (AAs) are also key nutrients for T cells, because they can serve as both a fuel source and a pool of biosynthetic precursors for protein and nucleic acid biosynthesis. The alanine, serine, and cysteine (ASC) system AA transporter 2 (ASCT2) as well as the T cells System L ('leucine-preferring system') transporter, SLC7A5 play critical roles in promoting the differentiation of the helper T cells [25]. In addition, several other AA transporters like the sodium-coupled neutral AA transporters SNAT1 and SNAT2 are also expressed by activated T cells [24]. Model-driven analysis of the expression data hints to some mechanistic insight as to why SLC7A5 and ASCT2 are significantly overexpressed (FDR adjusted p value < 0.0001, FC > 2). In the naive-cell model, SLC7A5 and ASCT2 are classified as ELE, which means they are sub-optimal and require more enzymatic steps than alternative pathways that might meet the same cellular needs of the cell in the inactivated state. However, upon activation, both transporters are required for maximizing growth (optima). Interestingly, while the differential expression doesn't show increased levels of expression for SNAT1, SNAT2 or SLC3A2, they are predicted to be 'optima' in both the TH-cell and naive-cell models.

E. Systems analysis shows tumor-like metabolism seen in TH-cell activation

Although at the functional level, TH cells and tumors have little in common, both T cells and cancer cells have strong proliferative signals and undergo metabolic reprogramming during their respective life cycles, and there exist clear functional and mechanistic similarities between the reprogramming events in each cell type [26]. Consistent with this, a system level analysis of the metabolic genes expression data in activated vs. inactivated TH-cells showed the activation (or overexpression) of enzymes implicated in tumorigenesis. The list of genes that were exclusively included in the TH-cell model (TH not subset naive), showed enrichment in terms such as 'tumorigenesis of carcinoma cells', 'advanced lyompocytic leukemia (ALL)', and 'childhood acute T-cell ALL' (Ingenuity Pathway Analysis (IPA) software core analysis, Fisher exact test p value < 0.05). Further filtering for enzymes only present in the TH-cell model and associated with cancer related terms resulted in 11 enzymes (Table 2). For instance, Dopa decarboxylase (DDC) (Table 2) catalyzes the decarboxylation of L-3,4-dihydroxyphenylalanine (DOPA) to dopamine, L-5-hydroxytryptophan to serotonin and L-tryptophan to tryptamine. This enzyme is predicted by the model to be active in TH-cell model only and not in the naive-cell model, while DEG didn't reveal similar info although there have been several reports supporting a regulatory role for Phenylalanine and Tyrosine metabolism regarding the proliferation and activation of TH-cells and subsequent immune responses [27]. DDC has also been previously implicated in childhood ALL [28]. Also, tyrosinase (TYR) (Table 2), which induces specific T-cell activation in vitro [29], is also associated with melanoma

[30]. Thus, a systematic analysis of the metabolic similarities and distinctions between activated TH cells and cancer cells would provide insights into how and why T cells adopt a cancer cell-like metabolic profile, while relying exclusively on DEGs can miss some valuable insights.

TABLE 2. Enrichment analysis results for genes exclusively included in the TH-model and involved in carcinogenesis

	Deg	model	pvalue
RRM1	0.1696056	optima in both	1.46E-04
RRM2	1.31E-10	optima in both	1.46E-04
RRM2B	0.5142105	optima in both	1.46E-04
ACP5	0.1039702	TH_only	4.84E-03
DDC	NA	TH_only	4.85E-04
ST6GALNAC3	0.1194187	TH_only	4.85E-04
HSD3B2	NA	TH_only	4.98E-03
SRD5A1	0.1555058	blocked in naiive, ELE in TH	7.94E-03
SRD5A2	NA	blocked in naiive, ELE in TH	7.94E-03
SLC19A1	0.4845694	TH_only	1.72E-02
TYR	NA	TH_only	1.98E-02

III. Genome-scale reconstructions provide mechanisitic insights into complex biological systems

Metabolic network reconstruction has matured into a methodological, systematic process with quality control and quality assurance steps that can be carried out according to standardized detailed protocols. However, the overall procedure for multi-omic integration with genome-scale models is an iterative workflow [2]. Once experimental data from the particular biological system under study is obtained, it is converted into constraints on model function followed by the successive application of experimentally derived constraints to the reaction network and eventually, generating a cell-type- and a condition-specific model. In this study, we have demonstrated how integrative analysis of gene expression data in the context of a genome-scale metabolic network can provide new insights and perspectives than using the gene expression data solely. Using activated and inactivated TH-cell-specific metabolic models, we have shown how the models correctly predicted glycerol-3-phosphate dehydrogenase (GPD2) role in the TH-cell activation program while having similar expression levels in both the activated and inactivated states. In addition, the model-driven analysis presented here provided some insights into the mechanistic explanation for reprogrammed amino acid transporters usage that takes place upon TH-cell activation. Further, our systematic analytical approach has shed some light onto similarities between TH-cell activation process in analogy to tumorigenesis. Taken together, our results lend support to the potential application of metabolic systems biology towards the field of immunometabolism.

In summary, metabolic networks provide a mechanistic scaffold for interpreting condition- and tissue-specific gene expression data and bridge the gap between the disparate levels of high-throughput data. CBM with metabolic networks thus enable better understanding of complex biological systems.

IV. Acknowldegmenet

Research reported in this publication was supported by competitive research funding from King Abdullah University of Science and Technology (KAUST). We also acknowledge generous funding from the Novo Nordisk Foundation Center for Bio-sustainability at the Technical University of Denmark.

References

[1] Ideker, T., J. Dutkowski, and L. Hood, *Boosting signal-to-noise in complex biology: prior knowledge is power.* Cell, 2011. **144**(6): p. 860-3

[2] O'Brien, E.J., J.M. Monk, and B.O. Palsson, *Using Genome-scale Models to Predict Biological Capabilities.* Cell, 2015. **161**(5): p. 971-987.

[3] Oberhardt, M.A., B.O. Palsson, and J.A. Papin, *Applications of genome-scale metabolic reconstructions.* Mol Syst Biol, 2009. **5**: p. 320.

[4] Shahzad, K. and J.J. Loor, Application of Top-Down and Bottom-up Systems Approaches in Ruminant Physiology and Metabolism. Curr Genomics, 2012. **13**(5): p. 379-94.

[5] Bordbar, A., N. Jamshidi, and B.O. Palsson, iAB-RBC-283: A proteomically derived knowledge-base of erythrocyte metabolism that can be used to simulate its physiological and patho-physiological states. BMC Syst Biol, 2011. **5**: p. 110.

[6] Jamshidi, N. and B.O. Palsson, Investigating the metabolic capabilities of Mycobacterium tuberculosis H37Rv using the in silico strain iNJ661 and proposing alternative drug targets. BMC Syst Biol, 2007. **1**: p. 26.

[7] Bordbar, A., et al., Constraint-based models predict metabolic and associated cellular functions. Nat Rev Genet, 2014. **15**(2): p. 107-20.

[8] Ideker, T., et al., Integrated genomic and proteomic analyses of a systematically perturbed metabolic network. Science, 2001. **292**(5518): p. 929-34.

[9] Chechik, G. and D. Koller, *Timing of gene expression responses to environmental changes.* J Comput Biol, 2009. **16**(2): p. 279-90.

[10] Anders, S., et al., *Count-based differential expression analysis of RNA sequencing data using R and Bioconductor.* Nat Protoc, 2013. **8**(9): p. 1765-86.

[11] Finotello, F. and B. Di Camillo, *Measuring differential gene expression with RNA-seq: challenges and strategies for data analysis.* Brief Funct Genomics, 2015. **14**(2): p. 130-42.

[12] Rapaport, F., et al., Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data. Genome Biol, 2013. **14**(9): p. R95.

[13] Consortium, S.M.-I., *A comprehensive assessment of RNA-seq accuracy, reproducibility and information content by the Sequencing Quality Control Consortium.* Nat Biotechnol, 2014. **32**(9): p. 903-14.

[14] Lobel, L., et al., *Integrative genomic analysis identifies isoleucine and CodY as regulators of Listeria monocytogenes virulence.* PLoS Genet, 2012. **8**(9): p. e1002887.

[15] Thiele, I., et al., *A community-driven global reconstruction of human metabolism.* Nat Biotechnol, 2013. **31**(5): p. 419-25.

[16] Zur, H., E. Ruppin, and T. Shlomi, *iMAT: an integrative metabolic analysis tool.* Bioinformatics, 2010. **26**(24): p. 3140-2

[17] Seumois, G., et al., *Epigenomic analysis of primary human T cells reveals enhancers associated with TH2 memory cell differentiation and asthma susceptibility.* Nat Immunol, 2014. **15**(8): p. 777-88.

[18] Feist, A.M. and B.O. Palsson, *The biomass objective function.* Curr Opin Microbiol, 2010. **13**(3): p. 344-9.

[19] Lewis, N.E., et al., Omic data from evolved E. coli are consistent with computed optimal growth from genome-scale models. Mol Syst Biol, 2010. **6**: p. 390.

[20] Kim, D., et al., *TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions.* Genome Biol, 2013. **14**(4): p. R36.

[21] Anders, S., P.T. Pyl, and W. Huber, *HTSeq--a Python framework to work with high-throughput sequencing data.* Bioinformatics, 2015. **31**(2): p. 166-9.

[22] Love, M.I., W. Huber, and S. Anders, *Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2.* Genome Biol, 2014. **15**(12): p. 550.

[23] Collison, L.W., E.J. Murphy, and C.A. Jolly, *Glycerol-3-phosphate acyltransferase-1 regulates murine T-lymphocyte proliferation and cytokine production.* Am J Physiol Cell Physiol, 2008. **295**(6): p. C1543-9.

[24] Poffenberger, M.C. and R.G. Jones, *Amino acids fuel T cell-mediated inflammation.* Immunity, 2014. **40**(5): p. 635-7.

[25] Sinclair, L.V., et al., *Control of amino-acid transport by antigen receptors coordinates the metabolic reprogramming essential for T cell differentiation.* Nat Immunol, 2013. **14**(5): p. 500-8.

[26] Macintyre, A.N. and J.C. Rathmell, *Activated lymphocytes as a metabolic model for carcinogenesis.* Cancer Metab, 2013. **1**(1): p. 5.

[27] Sikalidis, A.K., Amino acids and immune response: a role for cysteine, glutamine, phenylalanine, tryptophan and arginine in T-cell function and cancer? Pathol Oncol Res, 2015. **21**(1): p. 9-17.

[28] Trevino, L.R., et al., Germline genomic variants associated with childhood acute lymphoblastic leukemia. Nat Genet, 2009. **41**(9): p. 1001-5.

[29] Alijagic, S., et al., *Dendritic cells generated from peripheral blood transfected with human tyrosinase induce specific T cell activation.* Eur J Immunol, 1995. **25**(11): p. 3100-7.

[30] Yan, J., et al., Novel and enhanced anti-melanoma DNA vaccine targeting the tyrosinase protein inhibits myeloid-derived suppressor cells and tumor growth in a syngeneic prophylactic and therapeutic murine model. Cancer Gene Ther, 2014. **21**(12): p. 507-17.

SESSION

SIGNAL PROCESSING, IMAGING SCIENCE, AND DATA QUALITY ENHANCEMENT + HEALTH INFORMATION SYSTEMS

Chair(s)

TBA

The Stacked Frame Display for Optimizing the Display and Interpretation of Analog Data

Robert A. Warner, M.D.
Tigard Research Institute
Tigard, OR USA

1. Abstract

This paper describes the stacked frame display (SFD), a method for improving the accuracy and speed of interpreting analog data. The SFD involves the replacement of sequentially recorded traditional analog waveforms with narrow single lines. The amplitude and duration information that had been shown in the traditional analog waveforms is provided in the SFD by encoding this information using colors or the brightness of pixels in the linear display of the data. The method was tested by comparing its ability to interpret ambulatory electrocardiographic (ECG) data to that of traditional methods of interpreting such data. Compared to the traditional method of interpreting ambulatory ECG data, The SFD resulted in more accurate and efficient interpretations of these data. The SFD improves the accuracy and efficiency of interpreting sequentially recorded analog data.

Keywords: analog data, stacked frame display

2. Introduction

In medicine and in other fields such as seismology and engineering, it is frequently necessary to examine large amounts of data to detect important patterns in the information. These data often represent time series in which the data points are acquired sequentially over a given period. For example, many patients with known or suspected cardiac disease undergo ambulatory electrocardiographic (ECG) monitoring to detect, characterize and quantify episodes of arrhythmias or ischemia whose presence may put the patient at increased risk for disability or death. It is widely accepted that ambulatory ECG monitoring, which may last for hours or days, is a useful clinical tool. [1-3] However, studies have shown that current methods of detecting both arrhythmias and ischemia using the ambulatory ECG are suboptimal.[4-6] Improvements in electromagnetic and optical storage have permitted the collection of large amounts of clinically important data generated by ambulatory monitoring. This has made the use of accurate and efficient methods for reviewing these data especially important. For example, consider a three-day recording of data recorded by two standard ECG leads from a patient whose average heart rate is 70 beats per minute. The number of individual ECG complexes to be reviewed is:

2 leads x 70 beats/min. x 60 min./hr. x 24 hrs./day x 3 days = 604,800 ECG complexes

Furthermore, each ECG complex has multiple components, any combination of which may exhibit transient or persistent abnormalities in amplitude and/or duration. The considerable time required to review recordings such as these not only makes the process inefficient and therefore costly, but can also compromise the accuracy of the interpretations. First, the tedium associated with examining very large amounts of data can lead to intermittent inattention to the task at hand. Second, because of the large amounts of time required, it's often decided that it isn't feasible for a physician expert in electrocardiography to review all the ECG complexes that have been recorded during hours or days of monitoring. Therefore, it is common practice to have less highly trained technicians review the data first and then select portions of the original recording for subsequent review by a physician. Therefore, if a technician happens to miss important recorded events because of either the tediousness of the task or a relative lack of expertise in electrocardiography, the physician never even gets to examine the important data. In other words, the current system of reviewing ambulatory ECG data does not provide "full disclosure" of the data to the experts who are the most qualified to interpret the information.

The present paper describes a method of producing a meaningful display of even large amounts of sequentially acquired electronic data in a much smaller space than that which is required to display traditional analog tracings. The display is highly intuitive and, because of its compactness, it enables one to both review recorded data rapidly and to easily identify diagnostically useful patterns that otherwise would have been very difficult to detect. These features of the display eliminate the need for preliminary review of the data by a technician. Therefore it provides full disclosure of the data to the individuals who have the greatest expertise in the interpretation of the data.

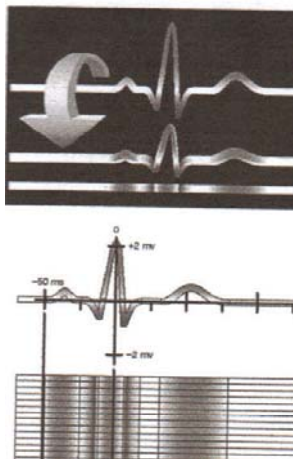
3.0 Methods

3.1 Description of the method

The method to be described and evaluated is called the stacked frame display (SFD) of analog

data and Figure 1 illustrates the rationale upon which it is based. The top panel of Figure 1 shows the P wave, QRS complex and T wave of a typical ECG complex. The SFD rotates an analog display 90^{o} such that the previously upright portions of the original waves now point directly toward the observer and the previously downward portions now point directly away from the observer. This results

Figure 1

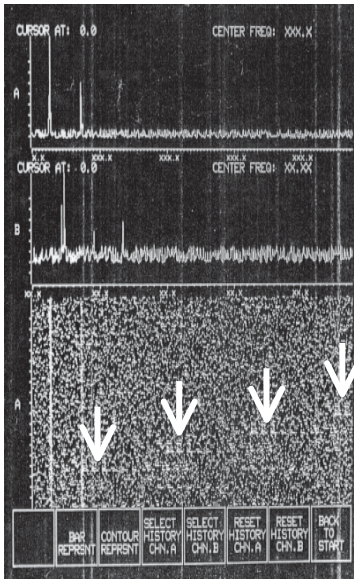


in the total height of the original analog display being reduced to the width of a single line. To restore the amplitude and duration information that had been shown in the pre-rotation analog display, one encodes the amplitude information using a system of colors, shades of gray or, in the case of a dichromatic display, the brightness of the pixels. For example, the portions of the waves that have rotated toward the observer could be colored red and those that have rotated away from the observer could be colored blue. The intensities of each of these colors are proportional to the amplitudes of the waves being depicted. As shown in Figure 1, information about the durations of each part of the original analog display is conveyed by the widths of the colored portions of each line.

In a sequential display of analog data, each set of rotated and encoded waves is placed either directly beneath or directly above its predecessor. In this way, the sequentially acquired data become stacked upon each other in chronological order. Also, the lower half of Figure 1 shows that the resulting rows of data can be aligned vertically along a common fiducial point, e.g. the onset or the peak or nadir of the QRS complex in a long recording of ECG data. Several advantages of the SFD are immediately apparent. First, an entire analog display (e.g. an ECG complex) of any amplitude is now represented on a screen or a printout by a thin single line that is placed immediately below or above the previously acquired data. This permits one to display and review in a given space on a screen or printout much more information than would have been possible using the original analog representation of the data. Second, the color, grayscale or pixel intensity coding of the information retains the amplitude and duration information that was present in the original analog display. Third, as Figure 2 shows, the compactness and arrangement of the display of data not only increases the speed with which the data can be reviewed, but also permit easier identification of important patterns in the data that might otherwise have not been apparent. Figure 2 shows analog displays of sonar signals that were being used to detect an undersea object. Panels 1 and 2 show the traditional appearance of these signals as they sweep from the left to the right side of the screen. In both Panels 1 and 2, the detected signals sweep across the screen and then disappear to accommodate the next set of detected signals. The transient nature of such displays of data on a screen make it difficult to detect patterns in the signals. However, Panel 3 represents a SFD encoded by pixel brightness of a series of screens of data from Panel 1. In this case, the earliest recorded signals are at the bottom and the most recently recorded signals are at the top of Panel 3. Throughout most of the display on Panel 3, the accumulated recorded signals appear random. However, the four vertical arrows show distinct horizontal lines that represent reflected signals of a consistent type from an underwater object that is moving from left to right. In contrast to the traditional analog displays of data in Panels 1 and 2, the SFD makes it possible to identify the presence, direction and (because of the known rate of acquisition of the data) the speed of the underwater object.

An additional feature of an SFD displayed on screen allows the user to click with a mouse on any part of the display and show the more familiar traditional form of the analog signal represented by that part of the SFD. This capability further increases the ease with which SFD displays can be interpreted. In addition, each click of the computer's mouse on a portion of the SFD would show the date and time at which those data were recorded. This feature permits one to measure precisely the times at which of any event of interest began and ended. Besides being presented on the screens of electronic monitors, one can choose to print or make screen shots of any portion of the display. This enables one to analyze data that had previously filled multiple screens, examine the data in sites remote from the electronic monitors and compile permanent records of the data that have been acquired.

Figure 2



3.2 Empirical evaluation of the method

I evaluated the SFD using 24-hour ambulatory ECG data and compared it to the traditional method of reviewing such records with respect to both accuracy and efficiency. I studied 21 randomly selected patients: 11 males and 10 females, ages 18 to 87 years (mean = 56) who had been cared for between September, 1997 and August, 1998 at the Baptist Memorial Hospital of the Bowman-Gray School of Medicine in Winston-Salem, North Carolina. Each patient had received a 24-hour ambulatory 2- channel ECGs for detecting possible arrhythmias and myocardial ischemia. The data from each tape were then transferred to a GE Medical Systems-Information Technologies **(**GEMS-IT)**,** MARS 8000 Arrhythmia Review Station™. A technician then scanned the resultant standard ECG waveforms on the screen of the Holter review station and selected segments of the 24-hour record for subsequent review by one of several cardiologists. Each cardiologist then generated a diagnostic report as part of each patient's medical record. Each of these diagnostic reports was based entirely on the traditional method of reviewing ambulatory ECG data aided by sophisticated commercial diagnostic algorithms. These reviews, analyses and reports of the ECG findings were augmented by the system's automated arrhythmia and ischemia detection algorithms and by graphs that depicted any changes in cardiac rate and ST segment displacement. I then reviewed the same 24-hour record after the data had been converted to the SFD for display on an electronic monitor. I was blinded to the standard ECG waveform, the above graphs and the diagnostic reports before I examined the SFD for the patients. During my initial review of the SFD, I recorded both my diagnostic findings and the amount of time required to review the entire 24-hour record using the SFD. Finally, I compared the findings obtained by reviewing only the SFD to those recorded in the original diagnostic report. To determine whether I had correctly identified an abnormality using the SFD, I then examined the traditional analog ECG complex that corresponded to each pattern of interest revealed by the SFD.

4. Results

Comparing the SFD to the original diagnostic reports shows that the SFD missed no abnormalities except for several clinically insignificant sinus pauses up to 2.8 seconds long in one patient. Conversely, the SFD detected a total of 9 episodes of consecutive ventricular beats (from 3 to 7 beats in duration) and a total of 10 episodes of sustained ST segment depression (from 1.5 to 25 minutes in duration). All these episodes of sustained ventricular beats and ST segment depression were clinically significant, but the patients' official diagnostic reports did not mention any of them. Also, the SFD correctly revealed the artifactual nature of what the official reports had incorrectly identified as a total of 4 episodes of consecutive ventricular beats (reportedly from 3 to 6 beats in duration). The estimated typical time for the initial scanning of the standard ECG waveforms by the technician plus their subsequent review by the cardiologist was 90 minutes. In contrast, the mean time required by me to review each 24-hour ambulatory record using the SFD and to record my findings was 12 minutes, 32 seconds (range = 5 minutes, 13 seconds to 25 minutes, 30 seconds).

5. Discussion

The study shows that the SFD is more accurate than the traditional method of analyzing 24-hour ambulatory ECG data for identifying consecutive ventricular beats, ST segment displacement and artifact. In addition, the mean time required to achieve this superior performance using the SFD is only about 14% of that typically required to analyze the same ECG data using the traditional method of analysis. The better performance of the SFD occurred despite the fact that the original, traditional analysis of the data had been augmented by the use of sophisticated commercial ECG diagnostic algorithms.

The simultaneous improvement in diagnostic accuracy and efficiency is consistent with the basic nature of the SFD. The traditional way of reviewing large numbers of serially acquired ECG complexes, i.e. observing the complexes as they scroll across a

screen, is very laborious. Therefore, in reviewing ECG data recorded by ambulatory monitors, a technician typically first reviews the ECG waveforms on a screen and selects specific portions of the record to show to a physician for subsequent interpretation. Thus, no matter how skilled the physicians are at interpreting ECG signals, they are able to see only those data that less highly trained technicians have chosen to show them. Furthermore, because of the monotony involved in reviewing ECG signals that have been recorded for long periods, even very experienced technicians are likely to miss some clinically significant events. In contrast, the ability of the SFD display to compress a large amount of data in a small space makes it possible to review effectively as many as 24 hours of accumulated ECG data much more rapidly than previously, even without prior screening by a technician. Therefore, the SFD can provide full disclosure of the ECG data directly to the physician who must generate the diagnostic report. This full disclosure probably contributes to the improved diagnostic accuracy for both arrhythmias and ischemia that the SFD exhibits.

Another factor that can increase the accuracy of the SFD is the user's ability to use the computer's mouse to toggle between any portion of the SFD and the corresponding traditional ECG waveform as illustrated in Figure 3 where the familiar ECG waveform is located near the bottom of the screen. This toggling feature allows the users of the SFD to elucidate any changes in the pattern of a patient's SFD by instantly examining the more familiar analog ECG complex. It also quickly teaches the users what features of the ECG the various types of these patterns displayed by the SFD represent. Because of this immediate instructional feedback, it's likely that the user's skill and efficiency will increase with continued use of the method.

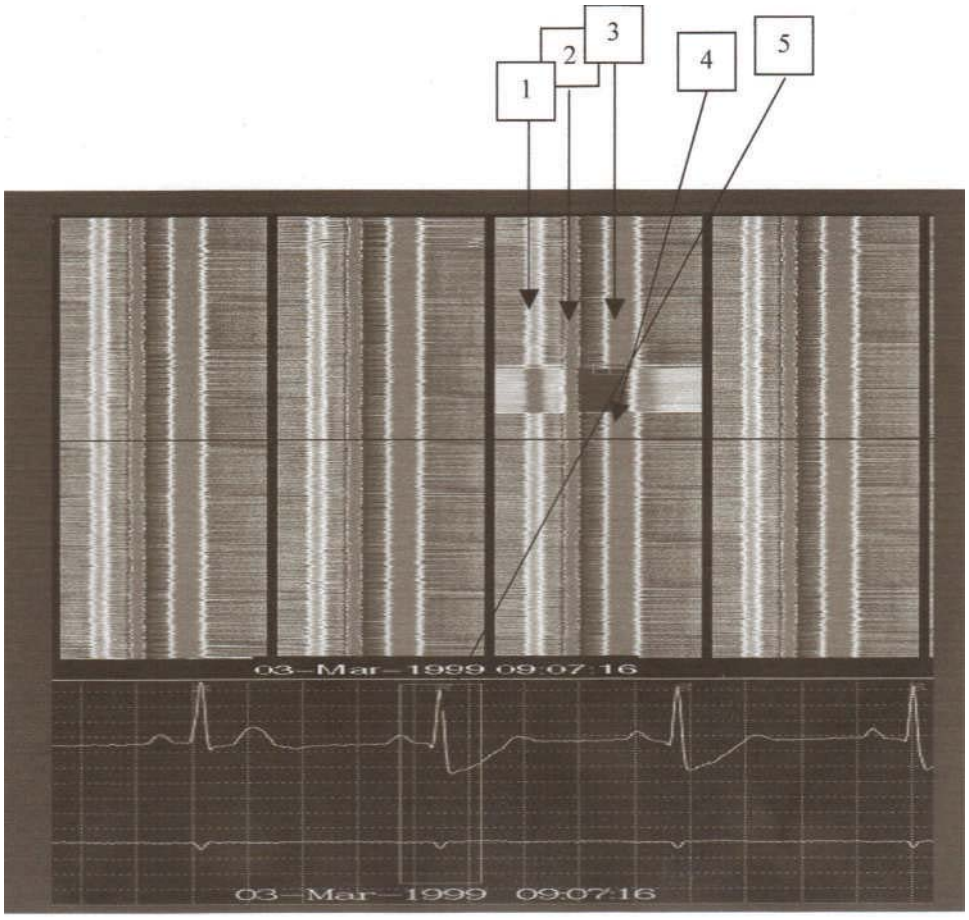
Figure 3 illustrates a screen of some of the SFD data that were used in this study. There are four columns of sequentially acquired data that are aligned vertically using the peak of the ECG R wave as a common fiducial point. Each separate column is located between a pair of thick black lines. The temporal sequence of the recordings is from top to bottom of the first column, then from top to bottom of the second column and so on. In Figure 3, Arrow 1 shows the location of the array of the P waves, Arrow 2 shows the R waves and Arrow 3 shows upright T waves. Arrow 4 shows a rectangle at the beginning of a region of the SFD that is markedly different from the preceding and most of the subsequent portions of the SFD. The SFD suggests that the ECG complexes that compose this small region have comparatively broad P waves, longer QT segments and intervals and lower T waves. Arrow 5 shows the traditional analog tracing at the point of transition to the above small region and confirms the morphological changes that the SFD suggest. The duration of the above episode lasted 15 seconds.

Figures 2 and 3 demonstrate that the ability of the SFD to compress a large number of sequentially acquired ECG data in a small space permits the reviewer to observe patterns that may not have been apparent from watching series of individual ECG complexes scroll across a screen. Furthermore, these patterns yield important quantitative information. The number of similar occurrences on one or more screens gives the frequency of an event during the period of recording. The duration of each episode of an event is proportional to the amount of vertical space that it occupies in a column. The magnitude of the change in amplitude of a portion of the ECG complex is proportional to the intensity of the change in color (or in the brightness of the pixels) that is associated with it. Finally the rapidity with which any changes take place show whether they are gradual or instantaneous. The ability to identify such patterns can have considerable diagnostic importance. For example, ST segment displacement that begins, gradually worsens and then gradually resolves over a period physiologically consistent with the known duration of myocardial ischemia would strongly suggest that the patient had had an episode of ischemia, whether symptomatic or silent. Conversely, ST displacement that occurred and ended abruptly or persisted for a very brief period would be much more consistent with transient artifact than with the diagnosis of ischemia. Figure 3 illustrates an episode associated with marked ST depression. However since this episode lasted only a few seconds and had a sudden onset and termination, it is unlikely that ischemia was the cause of the observed change in this patient's ST segments. Arrow #3 in Figure 3 shows how easily the SFD simultaneously demonstrates the abruptness, direction, severity and duration of an episode of ST displacement in a patient.

After one identifies patterns of interest using the SFD, one can then quantify the relevant features of the ECG even more accurately. This is because the system's computer has stored the digital data needed to generate the SFD and can therefore retrieve the precise computerized measurements of the analog signals associated with the identified patterns.

Besides using the SFD to detect and analyze ECG abnormalities in the initial evaluation of

Figure 3



a patient, one can also use it to assess the safety and efficacy of therapeutic interventions in the daily management of patients and in the performance of clinical trials. Other investigators have used the traditional ambulatory ECG to evaluate the treatment of arrhythmias and ischemia. The full disclosure and ease of review of accumulated ECG data that the SFD provides makes it ideal for analyzing the large numbers of ambulatory ECGs that those clinical trials can generate.

Although the present study specifically demonstrates the use of the SFD in ambulatory monitoring, it is likely that we can extrapolate our findings to the monitoring of hospitalized patients, e.g. in coronary or intensive care units. In these settings, nurses or monitor technicians often try to detect clinically important changes in the tracings of multiple patients who are being monitored simultaneously. If a given ECG complex remains on the monitor screen for only a few seconds, it is likely that some episodes of arrhythmia or ischemia will be missed. Even if all the patients' traditional ECG waveforms that had been obtained during a prolonged period were recorded, the task of subsequently reviewing them would be laborious and could diminish the ability of the professional staff to perform their other duties. Alternatively, using the SFD in conjunction with the real-time displays of monitored patients could facilitate the detection of arrhythmias and ischemia so that timely and clinically effective interventions would be more likely.

The superiority of the SFD method for reviewing ECG data compared to the more traditional method that used sophisticated commercial diagnostic ECG algorithms emphasizes an important point. The human eye and brain are extremely adept at quickly and accurately recognizing both simple and complex patterns. For example, it is common for a person to instantly recognize the face of an acquaintance even though that he might not have seen that acquaintance for a long time. This is because the person is able to easily discriminate the

acquaintance's face from the thousands of other faces that the person has seen during his life. The basic function of the SFD is to present sequentially acquired analog data in a compact and intuitive way. By doing this, the SFD permits the eye and the brain to perform the task of pattern recognition for which they are so well equipped.

6. Limitations of the Study

The number of patients evaluated in the study was small. Despite this, however, the improvements in both the detection of abnormalities and the time required for the review of the data are striking. This is especially remarkable since sophisticated diagnostic algorithms augmented the reviews that used the traditional analog displays
. Also, the portion of the present study that involves the review of the patients' recorded diagnostic reports is retrospective. A consequence of this is that the individuals who generated these reports could only provide estimates of the amount of time required to perform the traditional reviews of the ECG data. On the other hand, this aspect of the study probably makes the results of the traditional reviews more typical of day-to-day clinical practice than if the individuals had known that they were participating in a research study.

7. References

[1] Corder MP, Monaco JL, Kraf T, Levin RI. The introduction of ambulatory electrocardiographic monitoring for the diagnosis and management of myocardial ischemia. Am J Med Qual 1997:169-174.

[2] Benhorin J, Pinsker G, Moriel M, Gavish A, Tzivoni D, Stern S. Ischemic threshold during two exercise testing protocols and during ambulatory electrocardiographic monitoring. J Am Coll Cardiol 1993;22(3):671-677.

[3] Tomita F. Characteristics and clinical significance of silent myocardial ischemia during ambulatory electrocardiographic monitoring in patients with ischemic heart disease. Hokkaido Igaku Zasshi 1990;65(6):583-594.

[4] Di Marco JP, Philbrick JT. Use of ambulatory electrocardiographic (Holter) monitoring. Ann Intern Med 1990 ;113(1):53-68.,

[5] Deedwania PC, Carbajal EV. Exercise test predictors of ambulatory silent ischemia during daily life in stable angina pectoris. Am J Cardiol 1990;66(17):1151-1156.

[6] Grauer K, Leytem B. A systematic approach to Holter monitor interpretation. Am Fam Physician 1992;45(4):1641-1648

Refinement BRATUMASS' Data of Breast Phantom Processing Based on Compressive Sensing

A. Luxi Li[1], B. Weimin Ji[1], C. Yizhou Yao[2], D. Meng Yao[1]*, E. Blair Fleet[3], F. Erik D. Goodman[3], H. Huiyan Wang[4], I. John R. Deller[5]

[1] School of Info Sci and Tech, East China Normal University, Shanghai, China
[2] Central Michigan University, Mount Pleasant, MI U.S
[3]BEACON Center, Michigan State University, East Lansing, MI, U.S.
[4]School of Computer Sci and Info Egr, Zhejiang Gongshang University, Hangzhou, China
[5] ECE, Michigan State University, East Lansing, MI, U.S.
*Corresponding Author, e-mail: myao@ee.ecnu.edu.cn

Abstract - *The detection data to be analysis is obtained by Brest Tumor Microwave Sensor System (BRUTUMASS) [1]. BRATUMASS is developed to detect breast tumor for diagnosis purpose. In order to test the imaging quality of BRUTUMASS, an experiment is designed to use the BRATUMASS to detect a coin in a 3-D printed breast phantom [2]. And compressive sensing method is used to analysis this obtained data. In this paper, pseudorandom sequence sampling method-a method of compressive sensing-is used to process both simulation signals on Matlab and detection data of a breast phantom.*

Keywords: Compressive Sensing, Microwave Imaging, Breast Phantom

1 Introduction

Breast cancer is one of the major diseases which threaten, mostly, women. It usually comes from breast tumor which could later become worse and convert to breast cancer. Thus the earlier breast cancer is detected, the more likely to practice permanent cure. BRATUMASS is developed to detect breast cancer at an earlier stage, more treatable stage. It is a safe, mobile and cost-effective method compared to traditional diagnostic method such as soft X-ray mammography imaging. However, the detection data is non-linear and non-steady. Thus the analysis of the data becomes significantly important. On BRATUMASS, the frequency resolution is such important in discriminating different breast tissue. So, a strategy of "random sampling" is being researched as a means of further reducing noise in the scan data to overcome the conventional approaches to sampling signals follow Shannon theorem: the signal is uniformly sampled at or above the Nyquist rate. However, compressive sensing theory asserts that one can recover certain signals from far fewer samples than the traditional method use. Compared with conventional uniform sampling, compressive sampling reduces the number of samples without much perceptual loss. This quality is ideal in the developing of a portable cost-effective device.

2 The principle of BRATUMASS

For the breast tissues provide permittivity contrast between different tissue, backscatter would happen in edge interface of different tissue. The BRATUMASS obtain the microwave back signal via backscattered off different tissue and in every edge interface.

BRATUMASS consists of signal obtain device and data processing platform. The device consists of RF transceiver module, zero intermediate frequency (zero-IF) mixer, slot step frequency modulation signal generator, sampling module and data processing platform on PC. Fig.1 shows the partial block diagram of BRATUMASS.

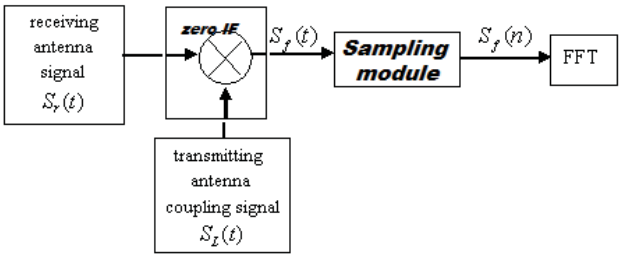


Figure 1 System diagram of BRATUMASS

On a scan point, transmitting antenna transmits slot step frequency signal shows in Fig. 2. The transmitting signal goes through two paths. One directly goes through transmitting antenna coupling network and reaches one of the two inputs of zero-IF mixer. Another transmits into the detection space and backscatters off the tissue's target edge interface that comes in its way and then reaches the second input of zero-IF mixer through receiving antenna. Because the signal transmits though coupling network reaches the mixer input almost instantly, the delay of the received signal from receiving antenna can be seen as the time that the

signal takes to travel from transmitting antenna to target tissues and then to receiving antenna. The output of zero-IF is the frequency difference of the emitting signal and back wave signal (Δf). And from Δf, the Δt can be calculated, thus the distance from every edge interface of tissues to antenna can be calculated. Fig.2.

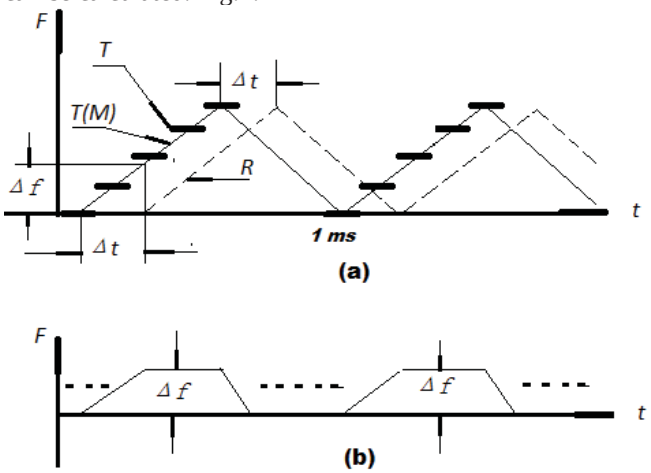


Figure 2 Transmitting and Receiving frequency pattern. (a) Transmitting and Receiving frequency pattern. (b) Output of zero-IF frequency signal.

In application, 17 scan data from 16 orientations are collected to reconstruct the image of the interior tissue and, if exists, tumor. The reconstruction images were imaged via BRATUMASS imaging platform. [3][4][5]

3 Compressive sensing

In the application of conventional uniform sampling, analog signals are usually low-pass filtered to remove most or all of the components above the Nyquist frequency (twice the maximum frequency present in the signal) in order to avoid aliasing. And if the sampling frequency is lower than the Nyquist rate of a signal, the spectrum of the sampled signal will alias and introducing distortion or error while reconstructing the original signal. In random sampling, when the sampling rate is lower than the Nyquist rate, the aliasing is possible to avoid by its randomness. Before getting to the analysis of random signal, the spectrum of random sampled signal and the antialiasing quality of random sampling is discussed.

3.1 Spectrum of random sampled signal

To random sampling sequence $\{x(t_0), x(t_1), \cdots x(t_{N-1})\}$, the sampling interval is random. Signal $x(t)$ after sampling can be expressed as:

$$x_s(t) = x(t)\sum_{n=0}^{N}\delta(t-t_n) \quad (1)$$

Practice Fourier transform on both sides of the equation, we have the relationship between the spectrum before sampling and after sampling:

$$X_s(\omega) = X(\omega) * \int_{-\infty}^{\infty}\sum_{n=0}^{N-1}\delta(t-t_n)e^{-j\omega t}dt = X(\omega) * \sum_{n=0}^{N-1}e^{-j\omega t_n} \quad (2)$$

If sample point is randomly distributed around time nT_0/N, and each point is independently distributed, then the expectation of $Xs(f)$ is:

$$E\{X_s(f)\} = \sum_{n=0}^{N-1}\int_{(n-1)\frac{T}{N}}^{n\frac{T}{N}} x(t_n)e^{-j2\pi ft_n}p_n(t_n)dt_n \quad (3)$$

Where, $P_n(t_n)$ is the probability density of the sample n. Assuming that the probability density function of a random variable τ_n is uniformly distributed in $[0,T_0/N]$, Then $P_n(t_n) = N/T_0$, we have:

$$E\{X_s(f)\} = \frac{N}{T_0}X_{T_0}(f) \quad (4)$$

The expectation of Xs(f) is unbiased estimator of Xs(f).

3.2 Spectrum calculation of random sampling data

The spectrum calculation difference of random sampled data and uniformly sampled data is the difference of time should take into account while doing integration. Assuming that $x(t)$ is a band limited signal, $Xc(f)$ is Fourier transform of $x(t)$, sampling interval is T, total sample number is N, then NT is the total sampling time. Let $x(n)$ be the uniformly sampled data, $x(t_n)$ (n = 1,2,3…) be the randomly sampled data, $X_D(f)$ be the Fourier transform of $x(t_n)$ We have:

$$X_C(f) = \int_0^{NT} x(t)\exp(-j2\pi ft)dt \quad (5)$$

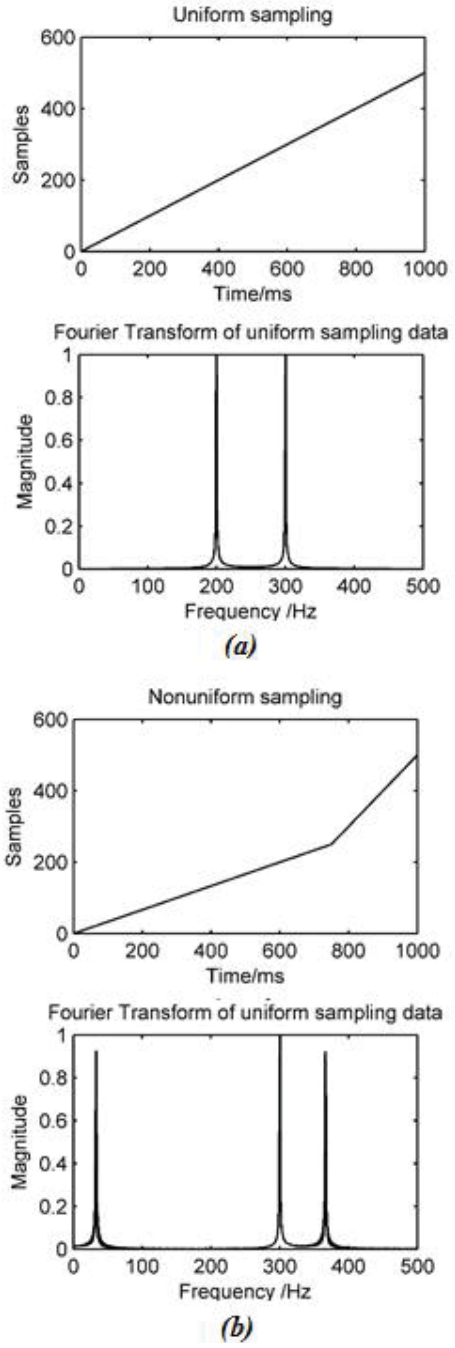
$$X_C(f) = \sum_{n=1}^{N} x(n)\exp(-j2\pi fn) \quad (6)$$

$$X_D(f) = \sum_{n=1}^{N} x(n)\exp(-j2\pi ft_n)(t_{n+1}-t_n) \quad (7)$$

3.3 Antialiasing of random sensing

In uniform sampling, the number of samples is proportional to the total time of sampling. In the practice, it will be the higher the sampling rate, the greater the slope. As shown in Fig. 3(a). Uniform sampling, a total number of 500

samples uniformly sampled in 1000ms. Then we try to do it non-uniformly. If we samples the first 250 sample points with 3ms intervals and the second 250 sample points with 1ms intervals as shown in Fig.3(b) Non-uniform sampling. Both sampling sequence were used to sample a 300Hz sinusoid. The uniform sampling is aliased because the sampling rate is lower than the Nyquist rate. The spectrum of non-uniform sampling is also aliased but the magnitude of 300Hz is greater than others. And if we do the non-uniform sampling randomly, the aliasing were avoided. As show in Fig.3(c).



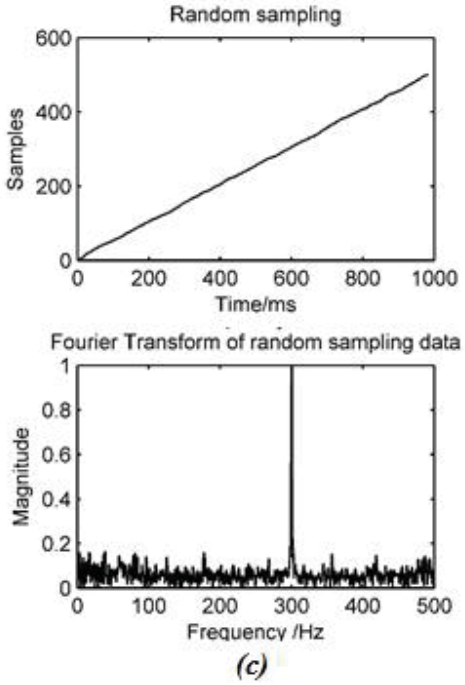


Figure 3 Aliasing of different sampling method. Signal: 300Hz. Sampling rate (mean): 500Hz. (a) Uniform sampling. (b) Non-uniform sampling. (c) Random sampling.

3.4 Resolution of random sensing

In application, as the sampling time is limited, the signal we obtained is the "view through the window". In other words, the signal we obtained is the product of the signal and a window function. In uniform sampling the physical resolution is $F_p = 1/T_p$ (T_p is the length of window function). This means signals have frequency difference smaller than T_p can't be distinguished. In this case, we set an experiment to sample the same signal with uniform sampling and random sampling. Setting signal $f(t) = sin\,\omega_1 t + sin\,\omega_2 t$, (f_1 = 300.0Hz, f_2 = 300.5Hz). The mean of random sampling rate is lower than the Nyquist rate, while T_p value is greater. We can see through the Fourier transform of both data. Fig.4. The resolution of random sampling data is higher than the uniform sampling data and the mean of sampling rate is lower (50Hz) than Nyquist sample rate.

4 About the Experiment

4.1 Preparation

The dielectric constant of breast phantom surface (ABS plastic) is smaller than that of human skin. The BRATUMASS is designed to generate resemble step microwave to penetrate through human skin. When the microwave reaches the plastic surface, strong reflection would happen. In this case, we choose to cover the plastic

surface with a slice of pigskin whose dielectric constant is close to

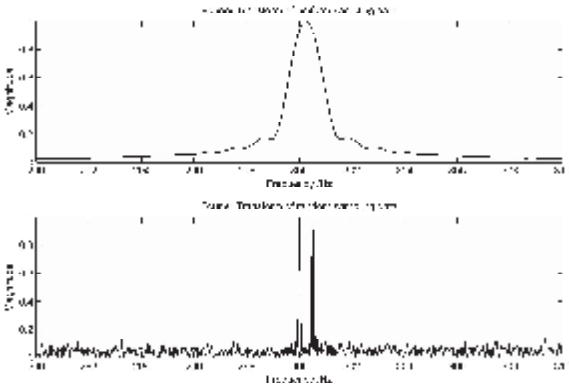


Figure 4 Signal: 300Hz, 300.5Hz; Samples: 1000; Uniform sampling rate: 1000Hz, Tp: 1s, Fails to distinguish two signals; Random sampling rate: about 50Hz(sampling intervals uniformly distributed between 0 and 0.04s), Tp: about 20s, Succeed to distinguish two signal.

human skin. A coin is used to represent the tumor inside breast phantom. (Fig. 5.). To obtain enough information, we sampled 16 scan points around the phantom (Fig.6.). One of the 16 scan points was sampled twice to confirm the conformity of data. Four groups of data (each contains 17 data series obtained from the 16 sample points) were sampled under different conditions to provide comparison.

Group A: With coin, with pigskin

Group B: With coin, without pigskin

Group C: Without coin, with pigskin

Group D: Without coin, without pigskin

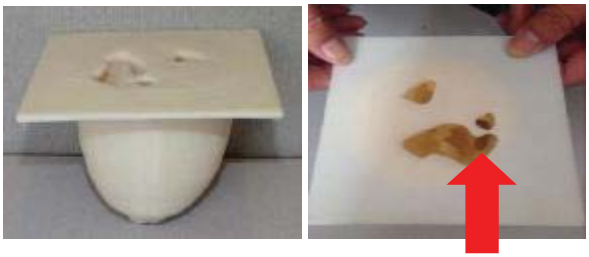


Figure 5 Location of the coin

Figure 6 Marking 16 detecting points

The liquid to fill the phantom is a solution of 90% Triton and 10% deionized water. Such solutions have been proposed previously to mimic the dielectric properties of biological tissues [6]. We can see the coin near the smallest hole. (Fig.7.).

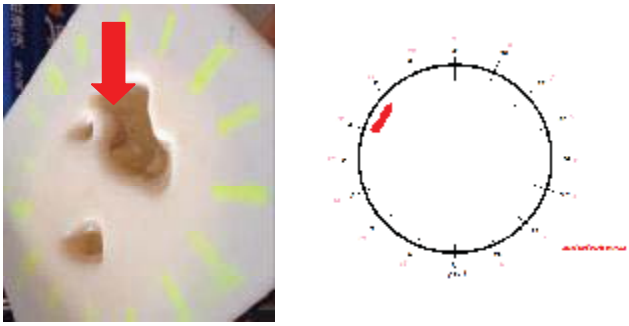


Figure 7 Coin in the solution and Supposed coin location in the inversion software platform

4.2 Signal analysis

The signal obtained from one scan point is showed in Fig.9. Sampling rate is 500Hz and 3000 samples were obtained. Its Fourier transform is showed in Fig.10. The target signal needed is below 50Hz. The greater magnitude in lower frequency indicates target backscatter at near distance (near the antenna).

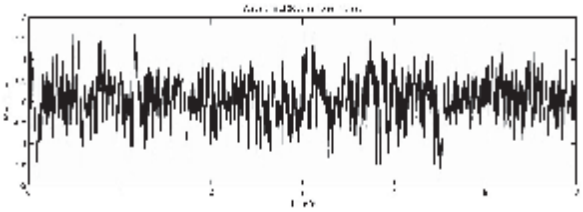


Figure 8 Waveform of 12th detected point. 3000 samples uniformly sampled.

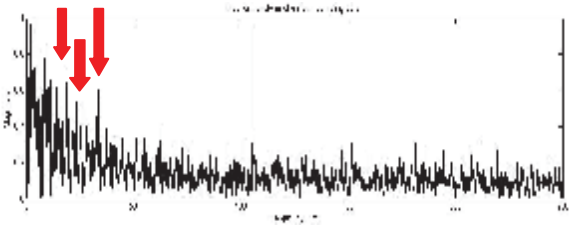


Figure 9 Fourier transform of 3000 uniformly sampled data

The data analysis below is based on the 3000 uniform sampling data. The spectrum from 0 to 50Hz is showed in Fig.11. The first spectrum is 3000 samples obtained in 6 seconds. And the second is the spectrum of the first 1500 samples in the first 3 seconds. The last one is 1500 samples randomly chosen from the 3000 samples in 6 seconds. Resolution of random sampling is clearly higher. Although the resolution of the first and last set of data should have similar resolution because of the same window function, the spectrum of random sampling is different from the spectrum of uniformly sampled 3000 data to some extent.

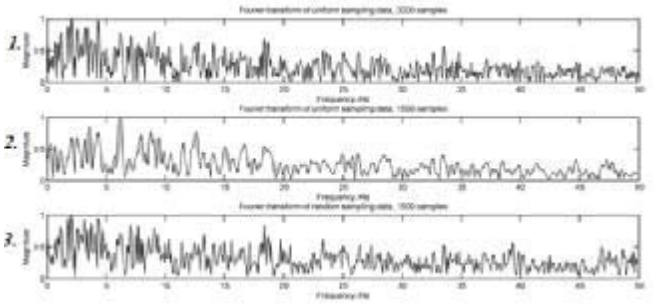


Figure 10 Spectrum of 1. Uniform 3000 samples; 2. Uniform 1500 samples; 3. Random 1500 samples

4.3 Results

Inversion of the data is successful, as is showed in Fig.12. However, the inversion of data without pigskin fails to distinguish target. Fig 12 is the inversion of data obtained through pigskin. Red pot of the picture represents high dielectric constant matter.

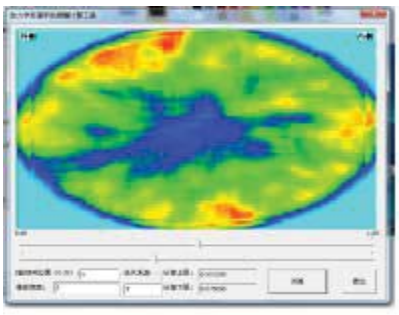


Figure 11 Inversion imaging of the Group C data

In order to improve the resolution of the inversion image, the inversion of random sampling data is still developing to deal with the effect of great magnitude signal, and to revise inversion image algorithm.

5 Conclusion

The resolution of spectrum is successfully improved using compressive sensing method. And the experiment to validate the imaging quality of microwave sensor system proves the ability of BRUTUMASS to distinguish substance of different dielectric constant and successfully locates the target coin. However, the inversion image of random sampling data still needs further research.

6 Acknowledgments

This work has been performed while Prof. M. Yao was a visiting scholar in Michigan State University, thanks to a visiting research program from Prof. Erik D. Goodman. M. Yao would also like to acknowledge the support of Shanghai Science and Technology Development Foundation under the project grant numbers 03JC14026 and 08JC1409200, as well as the support of TI Co. Ltd through TI (China) Innovation Foundation

7 References

[1] Yao, M., Tao, Z., Han, Z., Yao, Y., Fleet, B., Goodman, E.D., Wang, H., Deller, J. Breast tumor microwave sounding, imaging and system actualizing. Adv. inInformation Sciences 1(1), 1–21 (2013)

[2] Matthew J. Burfeindt, Timothy J. Colgan, R. Owen Mays, JacobD. Shea, Nader Behdad, Barry D. Van Veen, and Susan C. Hagness, "MRI-Derived 3-D-Printed Breast Phantom for Microwave Breast Imaging Validation," IEEE ANTENNAS AND WIRELESS PROPAGATION LETTERS, VOL. 11, pp. 1610-1613, 2012 .

[3] Zhi-fu Tao. R.Investigation on the methodologies of near-field microwave echo imaging integrity," The Ph.D. Thesis of East China normal university, 2011.

[4] Wang, Cui; Yao, Meng. J (2006) "*Practical Near-field Microwave Sounding Image Method for Early-stage Breast Cancer,"* Chinese Journal of Scientific Instrument. 2006(S3)

[5] Meng Yao, Zhifu Tao, Zhongling Han, "The Detection Data of Mammary Carcinoma Processing Method Based on the Wavelet Transformation," *International Journal of Online Engineering*, v 9, n SPECIALISSUE.6, p 69-72, 2013.

[6] S. Romeo, L. Etc "Dielectric characterization study of liquid-based materials for mimicking breast tissues,"*Microw. Opt. Technol. Lett.*, vol.53, no. 6, pp. 1276–1280, 2011.

A Proposed Warped Modified-B Time-Frequency Distribution Applied to Doppler Blood Flow Measurement

F. García Nocetti, J. Solano González, E. Rubio Acosta
Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas
Universidad Nacional Autónoma de México
Circuito Escolar, Ciudad Universitaria, México D. F., 04510, México
Contacting email: fabian.garcia@iimas.unam.mx

Abstract - *One of the main goals in ultrasonic Doppler blood flow measurement is the estimation of the mean velocity. The Doppler signal's instantaneous frequency has traditionally been used to estimate the mean velocity. In this work, a non-uniform discrete time frequency distribution is proposed: the warped discrete Modified-B distribution ($WTFD_{MB}$). The proposed procedure estimates the instantaneous frequency by concentrating the frequency resolution around the Doppler signal's instantaneous frequency. As a result, a better precision is obtained in the spectral estimation by using a $WTFD_{MB}$ for noisy signals when compared to other methods such as the Discrete Modified-B Time Frequency Distribution ($DTFD_{MB}$) with the instantaneous frequency calculated as the centroid of the spectrum.*

Keywords: Signal Processing, Warped Time-Frequency Distribution, Doppler Flow Measurement

1 Introduction

It is known that the blood flow mean velocity through a vessel's cross section is proportional to the instantaneous frequency of the Doppler ultrasonic signal. This work is focused on the accurate computation of the instantaneous frequency of a signal (Carotid Artery simulated signal) in the presence of noise.

A classic method to estimate the instantaneous frequency of a signal includes the computation of its spectrogram using a Short Time Fourier Transform (STFT). However, it assumes that the analysed signal is stationary and it compromises its temporal and frequency resolution. An alternative method is to use the Cohen Class Time Frequency Distributions that overcomes the stationary assumption. However, in some cases, it is desirable to increase only its frequency resolution around certain frequency of interest, for example, around the instantaneous frequency. That can be achieved if the length of the analysed discrete signal is increased but it also increases notably the computational cost. On the other hand, non-uniform discrete Fourier transforms are available such as the Warped Discrete Fourier Transform, which is able to achieve that task without having to increase the length of the analysed discrete signal.

The aim of the work presented in this paper is to incorporate a warped frequency scale (non-uniform) to the Cohen class of time-frequency distributions. We focus on the development of the warped discrete Modified-B time-frequency distribution, although the procedure can be easily extended to other distributions. In a previous work, a warped discrete Wigner-Ville time-frequency distribution has been proposed [12].

2 Time Frequency Distributions of the Cohen Class

The time frequency distributions of the Cohen class (TFD) are defined as follows [1]. Let $\phi(\theta,\tau)$ be the distribution kernel. That kernel determines the distribution and its characteristics of temporal and frequency resolution. Let also $\psi(t,\tau)$ be the Fourier transform of the distribution kernel:

$$\psi(t,\tau)=\frac{1}{2\pi}\int_{-\infty}^{\infty}\phi(\theta,\tau)e^{-it\theta}d\theta \tag{1}$$

Then, let

$$R_t(\tau)=\int_{-\infty}^{\infty}\psi(t-\mu,\tau)x\left(\mu+\tfrac{1}{2}\tau\right)x^*\left(\mu-\tfrac{1}{2}\tau\right)d\mu \tag{2}$$

be the deterministic generalised local auto-correlation function, where $x(t)$ is a complex signal. Finally, let

$$TFD(t,\omega)=\frac{1}{2\pi}\int_{-\infty}^{\infty}R_t(\tau)e^{-j\omega\tau}d\tau \tag{3}$$

be the time frequency distribution, which is defined as the Fourier transform of the local auto-correlation function, where

t is the time variable and ω is the (angular) frequency variable.

3 Modified-B Time Frequency Distribution

The Modified-B time frequency distribution (TFD_{MB}) belongs to the Cohen class [9][10][11]. Its kernel is:

$$\phi(\theta,\tau)=\frac{\Gamma(\alpha+i\pi\theta)\Gamma(\alpha-i\pi\theta)}{\Gamma^2(\alpha)} \tag{4}$$

Then, the distribution is defined by:

$$TFD_{MB}(t,f)= G_\alpha\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(\frac{1}{\cosh^{2\alpha}(t-\mu)}\right)x^*\left(\mu-\tfrac{1}{2}\tau\right)x\left(\mu+\tfrac{1}{2}\tau\right)d\mu e^{-j2\pi f\tau}d\tau \tag{5}$$

where $G_\alpha=\Gamma(2\alpha)/\left(2^{2\alpha-1}\Gamma^2(\alpha)\right)$ and f is the frequency variable,. The direct discretization of expression (5) constitutes the discrete distribution ($DTFD_{MB}$), that is:

$$DTFD_{MB}(n,k)=2G_\alpha\bullet \sum_{p=-N+1}^{N-1}W^*(-p)W(p)\sum_{m=-M}^{M}\left(\frac{1}{\cosh^{2\alpha}(m)}\right)x^*(m+n-p)x(m+n+p)\left(e^{-j\frac{2\pi kp}{L}}\right)^2 \tag{6}$$

where n is the time discrete variable, k is the frequency discrete variable, $W(n)$ is a sampling window and $x(n)$ is a discrete complex signal with support $n=-N+1,...,N-1$ and length $L=2N-1$.

Now, the discrete Modified-B time frequency distribution with periodic extension ($PTFD_{MB}$) is stated [2]. The procedure essentially consists on the following.

First the discrete distribution (6) is valued at $n=0$:

$$DTFD_{MB}(0,k)=2G_\alpha\bullet \sum_{p=-N+1}^{N-1}W^*(-p)W(p)\sum_{m=-M}^{M}\left(\frac{1}{\cosh^{2\alpha}(m)}\right)x^*(m-p)x(m+p)\left(e^{-j\frac{2\pi kp}{L}}\right)^2 \tag{7}$$

Second, the generalised local auto-correlation function is identified:

$$R_t(p)=G_\alpha\bullet W(p)W^*(-p)\sum_{m=-M}^{M}\left(\frac{1}{\cosh^{2\alpha}(m)}\right)x^*(m-p)x(m+p) \tag{8}$$

where $p=-N+1,...,N-1$. Now, the function $\overline{R}_t(p)$ which constitutes the periodic extension of $R_t(\tau)$, is constructed as follows:

$$\overline{R}_t(p)=\begin{cases}R_t(p) & 0\le p\le N-1\\ 0 & p=N\\ R_t(p-2N) & N+1\le p\le L\end{cases} \tag{9}$$

where $p=0,...,L$ and its length is $\overline{L}=L+1=2N$.

Finally, note that (7) can be written as:

$$DTFD_{MB}(0,k)=2\sum_{p=0}^{\overline{L}-1}\overline{R}_t(p)\left(e^{\frac{-j2\pi kp}{\overline{L}}}\right)^2 \tag{10}$$

At this point, the following scaling in the frequency axis is carried out. It consists on reducing by half the frequency resolution. The result is denominated a discrete Modified-B time frequency distribution with periodic extension ($PTFD_{MB}$):

$$PTFD_{MB}(0,k)=2\sum_{p=0}^{\overline{L}-1}\overline{R}_t(p)e^{\frac{-j2\pi kp}{\overline{L}}} \tag{11}$$

4 Warped Discrete Fourier Transform

The main characteristic of the warped discrete Fourier transform (WDFT) [3][4][5] is that it can concentrate the frequency resolution around a frequency of interest, since it possesses a non-uniform frequency resolution. Contrary to the conventional discrete Fourier transform (DFT) that possesses a uniform frequency resolution.

The warped discrete Fourier transform with a first order all-pass filter is defined as:

$$WDFT(k)=\sum_{p=0}^{\overline{L}-1}x(p)\left[\frac{\alpha^*+e^{\frac{-j2\pi k}{\overline{L}}}}{1+\alpha e^{\frac{-j2\pi k}{\overline{L}}}}\right]^p \tag{12}$$

where $\alpha=|\alpha|\exp(j\varphi)$ is a complex parameter that determines the warped frequency scale, $x(n)$ with $n=0,...,\overline{L}-1$ is a complex discrete signal with length $\overline{L}$, and $k=0,...,\overline{L}-1$ is an index related to the discrete frequency.

The warped frequency scale mapping is given by:

$$\Omega_W=\Omega+2\arctan\left(\frac{|\alpha|\sin(\varphi-\Omega)}{1+|\alpha|\cos(\varphi-\Omega)}\right) \tag{13}$$

where $-\pi\le\Omega\le\pi$ with $\Omega=2\pi k/\overline{L}$ is the conventional uniform frequency scale.

The magnitude of the parameter α, $|\alpha|$, is selected according to the percentage of frequency points to be

concentrated inside the spectral lobe related with the frequency of interest. The angle of the parameter α, φ, is selected according to the value of the frequency of interest.

5 Warped Discrete Time Frequency Distributions

In this work, a warped frequency scale is incorporated to the discrete time frequency distributions with periodic extension. Although this procedure is illustrated for the Modified-B distribution, its generalisation can be directly generated. The procedure calculates the warped discrete Fourier transform of the periodic extension of the generalised local auto-correlation function, instead of calculating its conventional discrete Fourier transform. Then, the warped discrete Modified-B time frequency distribution ($WTFD_{MB}$) is:

$$WTFD_{MB}(0,k) = 2\sum_{p=0}^{\overline{L}-1} \overline{R}_t(p) \left[\frac{\alpha^* + e^{\frac{-j2\pi k}{\overline{L}}}}{1+\alpha e^{\frac{-j2\pi k}{\overline{L}}}} \right]^p \tag{14}$$

where $k = 0,...,\overline{L}-1$ and the signal $\overline{R}_t(p)$ is the periodic extension of the generalised local auto-correlation function (9).

6 Frequency Estimation using the Warped TFD

The procedure to estimate the instantaneous frequency of a signal with a dominating single frequency and a narrow bandwidth is as follows [5].

First, the spectrum of the signal is calculated, using a conventional discrete Fourier transform:

$$S(k) = \left| \sum_{p=0}^{L-1} x(n) e^{\frac{-j2\pi kn}{L}} \right|^2 \tag{15}$$

Second, the value of the parameter $\alpha = |\alpha| \exp(j\varphi)$ is calculated. For this, the preliminary instantaneous discrete frequency of the signal (the centroid of the spectrum) is calculated:

$$k_i = \frac{\sum_{k=-N+1}^{N-1} k \bullet S(k)}{\sum_{k=-N+1}^{N-1} S(k)} \tag{16}$$

The angle φ of the parameter α is the instantaneous discrete frequency of the signal but expressed in radians. It is important to consider the reduccion by half of the frequency resolution.

Third, the warped discrete time frequency distribution of the signal is calculated according to (14). Then, the definitive instantaneous frequency of the signal is the phase associated to the frequency component with the maximum magnitude that is obtained.

$$\Omega_k = angle\left[\max_k \left\{ \left| WTFD_{MB}(0,k) \right| \right\}_{k=0}^{\overline{L}-1} \right] \tag{17}$$

7 Application to Doppler Flow Measurement

It is known that the mean velocity of the blood flow through the cross section of a vessel is proportional to the instantaneous frequency of a Doppler ultrasonic signal. In this work, a femoral artery signal is considered for this study. The simulation of that signal is detailed in [6][7][8]. The theoretical instantaneous frequency is shown in figure 1.

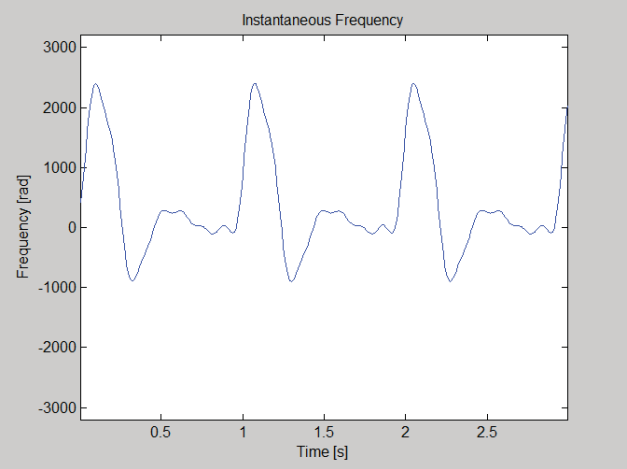


Figure 1. Theoretical instantaneous frequency of the simulated signal (Femoral artery).

The sampling frequency used is 12800 Hz. Sampling windows with 50% of overlapping and length equal to 256 are used.

For those conditions, the instantaneous frequency is calculated using both, the discrete Modified-B time frequency distribution with periodic extension, $PTFD_{MB}$ (11), and the warped discrete Modified-B time frequency distribution, $WTFD_{MB}$ (14). The procedure to calculate the instantaneous frequency has been described in the section 6. Finally, the RMS error respect to the theoretical instantaneous frequency is calculated.

8 Results

Figure 2 depicts a graph with the estimated instantaneous frequency using the warped discrete Modified-B time frequency distribution.

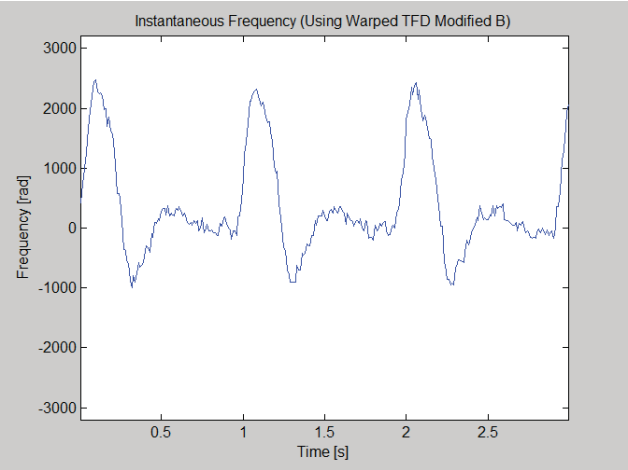


Figure 2. Estimated instantaneous frequency of the simulated signal using the warped discrete Modified-B time frequency distribution (SNR=10dB)

Table 1 shows the RMS errors in the estimation of the instantaneous frequency for the simulated signal. Different levels of normalised gaussian noise (SNR of 40, 30, 20 and 10 dB) have been added, with a concentration of frequency points around the instantaneous frequency of 40%. The precision of the warped distribution to estimate the instantaneous frequency is notoriously better in the presence of noise.

Modified-B TFD	SNR				
	Without Noise	40dB	30dB	20dB	10dB
With periodic extension $PTFD_{MB}$	34	34	34	38	118
Warped $WTFD_{MB}$	87	87	87	87	87

Table 1.- RMS error [Hz] obtained in the estimation of the instantaneous frequency of the simulated signal

9 Conclusions

This approach presented in this paper incorporates a warped frequency scale (non-uniform) to the Cohen class time frequency distributions. Particularly, the warped discrete Modified-B time frequency distribution has been developed, although this procedure can be extended easily to other distributions. The method is applied to estimate the mean velocity of the blood flow through a vessel, which is proportional to the instantaneous frequency of the ultrasound signal obtained in the process of Doppler flow measurement.

The experiments have been carried out using a Femoral artery simulated signal. The results obtained by the warped discrete time frequency distribution are compared with those obtained by the discrete Modified-B time frequency distribution with periodic extension. Results show that the distribution with periodic extension is easier to calculate since FFT-like algorithms of complexity $O\left(N \log N\right)$ are used; while the warped distribution uses algorithms which are based on the matrix multiplication whose complexity are $O\left(N^2\right)$. Nevertheless, the precision of the warped distribution to estimate the instantaneous frequency is notoriously better in the presence of noise.

Acknowledgements

The authors acknowledge project DGAPA-UNAMPAPIIT (IN101213), project Consorciado CYTED (P506PIC0295) by the financial support. Also we want to acknowledge to M. Fuentes, J. Contreras, S. Padilla and M. Vazquez for their technical support in the development of this work.

References

[1] L. Cohen, Time-Frequency Analysis (Prentice-Hall PTR, 1995)

[2] B. Boashash, P. Black, An Efficient Real-Time Implementation of the Wigner-Ville Distribution, *IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-35*(11), 1987, 1611-1618.

[3] A. Markur, S.K. Mitra, Warped Discrete Fourier Transform: Theory and Applications, *IEEE Transactions on Circuits and Systems -I:Fundamental Theory and Appications, 48*(9), 2001, 1086 –1093.

[4] S. Franz, S.K. Mitra, J.C. Schmidt, G. Doblinge, Warped Discrete Fourier Transform: a New Concept in Digital Signal Processing, *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing, 2*, 2002, 205-208.

[5] S. Franz, S. K. Mitra, G. Doblinger, Frecuency Estimation using Warped Discrete Fourier Transform, *Signal Processing, 83*, 2003, 1661-1671.

[6] J. Cardoso, G. Ruano, P. Fish, Nonstationary Broadening Reduction in Pulsed Doppler Spectrum Measurements Using Time-Frequency Estimators, *IEEE Transactions on Biomedical Engineering, 43*(12), 1996, 1176-1186.

[7] J. A. Jensen, *Estimation of Blood Velocities using Ultrasound* (Cambridge University Press, 1996).

[8] P.Fish, *Physics and Instrumentation of Diagnostic Medical Ultrasonic* (John Wiley Sons, 2000).
[9] B. Boashash, V. Sucic, Resolution Measure Criteria for the Objective Assessment of the Performance of Quadratic Time-Frequency Distributions, *IEEE Transactions on Signal Processing, 51*(5), 2003, 1253-1263
[10] Z. Hussain, B. Boashash, Adaptive Instantaneous Frequency Estimation of Multicomponent FM Signals Using Quadratic Time–Frequency Distributions, *IEEE Transactions on Signal Processing, 50*(8), 2002, 1866-1876
[11] B. Barkat, B. Boashash, A High-Resolution Quadratic Time–Frequency Distribution for Multicomponent Signals Analysis, *IEEE Transactions on Signal Processing, 49*(10), 2001, 2232-2239
[12] E. Rubio, J. Solano, F. Torres, F. García-Nocetti, A Proposed Warped Wigner-Ville Time Frequency Distribution Applied to Doppler Blood Flow Measurement, *Proceedings of Biomedical Engineering (BioMED),* 2006.

An Exploration of the Relationship between Resilience and Quality of Life

Daxin Zhu[1], Xiaodong Wang[2*], and Jun Tian[3*]
[1] Quanzhou Normal University, Quanzhou, Fujian, China
[2] Fujian University of Technology, Fuzhou, Fujian, China
[3]Fujian Medical University, Fuzhou, Fujian, China

Abstract - *AIM: To study the relationship between resilience and quality of life (QOL) in patients with digestive cancer. Methods: The resilience of patients was measured prior to treatment, and their psychological distress, fatigue status, and treatment side effects were assessed 3 weeks after. Their QOL was measured after their treatment ended. A relationship model of these variables was constructed using path analysis. Results: Resilience explained 33.2% of the variance in psychological distress, 16.1% of the variance in fatigue, and 1.23% of the variance in side effects. The relationship between resilience and QOL was statistically significant ($t = 4.499$, $P < 0.001$) when psychological distress, fatigue, and side effects were absent from the regression model, whereas the adjusted regression coefficient of resilience was not statistically significant ($t = 1.562$, $P > 0.05$) when these variables were added. Psychological distress together with fatigue and side effects could explain 52.40% of the variance in QOL ($P < 0.05$). Physiological distress accounted for 28.94% of the total effect on QOL, fatigue accounted for 33.72%, side effects accounted for 22.53%, and resilience accounted for 14.80%. Conclusion: Resilience is not an independent predictor of QOL in patients with digestive cancer, but it is a main factor influencing psychological distress and side effects.*

Keywords: Resilience; Psychological distress; Fatigue; Quality of life; Path analysis; Digestive cancer

1 Introduction

Cancer is a disease that severely damages human physical and mental health. Its diagnosis significantly affects a patient's emotional and psychological status [1], with the patient's quality of life (QOL) often being affected considerably after surgery and chemotherapy/radiotherapy [2–4]. However, many studies have found that cancer patients with similar diseases and treatment status have significantly different QOLs [5, 6]. Psychologists believe that resilience is the main factor that causes patients with similar situations to have different perceptions of their QOL [7, 8].

* Corresponding author. This work was supported in part by the Natural Science Foundation of Fujian (Grant No.2013J01247), Fujian Provincial Key Laboratory of Data-Intensive Computing and Fujian University Laboratory of Intelligent Computing and Information Processing.

Resilience is an individual's capacity to maintain his or her psychological and physical well-being in the face of adversity [8]. In recent years, the role of resilience in the process of cancer treatment has been given increasing attention [9–14]. Studies have found that resilience can powerfully predict patients' fatigue in the treatment [12], good resilience can help patients reduce treatment-induced damage to bodily functions and shorten the time of their recovery thereof [13], and patients with good resilience are able to treat their disease correctly and maintain relatively good mental and psychological states, thereby resulting in a better QOL [11, 12].

Although much research has shown that a relationship exists between QOL and resilience in cancer patients [11, 14–17], limited information is available on the nature of this relationship and the degree of the influence of resilience on QOL. Exploring whether resilience is an independent predictor of QOL and estimating the degree of its impact on QOL can make us understand the role of resilience in improving the QOL of cancer patients, as well as provide clinical staff with information on psychological intervention and psychological care programs for cancer patients.

In this study, we used path analysis to detect the relationships of resilience, psychological distress, fatigue, and treatment side effects with QOL. We drew a path map to show the paths of the influences of resilience, psychological distress, fatigue, and side effects on QOL and quantitatively estimated their direct and indirect effects on it. Our results may help explain whether strategies to improve resilience are important in promoting the QOL of cancer patients.

2 Materiails and Methods

Participants

We selected patients with digestive tumors from Fujian Province for this study, because digestive cancer ranks as the leading cause of death in this area. They were recruited from five province-level hospitals in Fuzhou City during 2008–2011. The study sample was limited to patients whose tumors were located in the esophagus, stomach, or colorectum. The following eligibility criteria were used: (1) age between 18 and 70 years; (2) non-illiteracy; (3) absence of mental or psychological disease; and (4) with known diagnosis of cancer.

All participants provided their written informed consent. The study was approved by the relevant institutional review boards for human research from Fujian Medical University.

Measurements

The RS-14, which was proposed by Wagnild [18], was used to measure the resilience of the participants. It is a 14-item questionnaire, and the score for each item ranges from 1 (not true) to 7 (true). Patients score the items based on their personal circumstances. The total score of the scale ranges from 14 to 98, with a high total score indicating good resilience. The RS-14 has been used for measuring an individual's degree of resilience in a wide variety of age groups [19], and its reliability and validity have been confirmed by many researchers [20]. In the present study, the Chinese version of this tool was found to have a reliability of 0.93.

The Hospital Anxiety and Depression Scale, which is a 14-item (7 for the anxiety subscale, 7 for the depression subscale) questionnaire, was used to evaluate the psychological distress of the participants [21]. The score for each item ranges from 0 to 3. Patients score the items based on their current situation. The total score of the scale ranges from 0 to 42, with a high total score indicating severe psychological distress. The Chinese version of this scale has been confirmed to be suitable for Chinese patients [22]. In the current study, this version had a reliability of 0.92.

The 20-item Multidimensional Fatigue Inventory Scale, developed by a Dutch research group [23], was used for measuring the fatigue of the participants. The score for each item ranges from 1 (true) to 5 (not true). Patients score the items based on their current situation. A high total score indicates severe fatigue. The Chinese version of this instrument has been confirmed to be suitable for Chinese patients [24].

Treatment side effects in the participants were examined from seven aspects: gastrointestinal system, respiratory system, liver and kidney, heart, hair, skin, and nervous system. The severity of side effects in each of these aspects had five ordinal scales: not at all, mild, moderate, a bit of severe, and severe, which were scored 1–5, respectively. The total score was the sum of the scores of the seven categories, with a high total score indicating severe side effects.

The European Organization for Research and Treatment of Cancer Core Questionnaire (Version 3.0) determines the QOL of cancer patients [25]. It is a 30-item questionnaire that includes 28 items scored from 1 to 4 and 2 items scored from 1 to 7. The Chinese version of this questionnaire has been confirmed to be suitable for Chinese cancer patients [25]. In the current study, the sum of the scores for Items 1–5 and Items 8–19 describes the physical aspect of QOL (QOL-Physical), the sum of the scores for Items 20–25 describes the mental aspect of QOL (QOL-Mental), and the sum of the scores for Items 6, 7, 26, and 27 describes the social aspect of QOL (QOL-Social). All these scores were transformed into values in the range of 0–100. A high total score indicates good QOL.

Procedure

Data on each patient were collected in three periods: Before the first treatment cycle began (first period), the patient's resilience was measured by trained graduate students from Fujian Medical University; in the third week of treatment (second period), the patient's psychological distress, fatigue, and side effects were measured by trained nurses in the hospitals; and at the end of the first treatment cycle(third period), the patient's QOL was measured by trained graduate students from Fujian Medical University.

Path analysis model

Low resilience affects mental health [26]. Poor mental health can increase side effects and fatigue [27] and, together with fatigue and side effects, influence the QOL of an individual [28, 29]. Therefore, we assumed that the models shown below describe the relationships between resilience (x), psychological distress (y1), fatigue (y2), side effects (y3), QOL-Social (z1), QOL-Mental (z2), and QOL-Physical (z3):

$$\begin{cases} y_1 = a_1 + \alpha_1 x + e_1 \\ y_2 = a_2 + \alpha_2 x + e_2 \\ y_3 = a_3 + \alpha_3 x + e_3 \end{cases} \quad (1)$$

$$\begin{cases} z_1 = b_1 + \beta_{11} y_1 + \beta_{21} y_2 + \beta_{31} y_3 + \varepsilon_1 \\ z_2 = b_2 + \beta_{12} y_1 + \beta_{22} y_2 + \beta_{32} y_3 + \varepsilon_2 \\ z_3 = b_3 + \beta_{13} y_1 + \beta_{23} y_2 + \beta_{33} y_3 + \varepsilon_3 \end{cases} \quad (2)$$

Models (1) and (2) are plotted as a path map. In the figure, r12, r23, and r13 represent correlation coefficients between psychological distress (y1), fatigue (y2), and side effects (y3).

Method of estimating effects on QOL

The path coefficients were estimated using path analysis. The degree of the effects of resilience on the three domains of QOL is equal to the sum of the products of the path coefficients in the path map as follows:

$$\begin{cases} \text{Effect}(x \rightarrow z_1) = \alpha_1 \times \beta_{11} + \alpha_2 \times \beta_{21} + \alpha_3 \times \beta_{31} \\ \text{Effect}(x \rightarrow z_2) \\ = \alpha_1 \times \beta_{12} + \alpha_2 \times \beta_{22} + \alpha_3 \times \beta_{32} \\ \text{Effect}(x \rightarrow z_3) \\ = \alpha_1 \times \beta_{13} + \alpha_2 \times \beta_{23} + \alpha_3 \times \beta_{33} \end{cases} \quad (3)$$

The coefficient βij in Model (2) expresses the direct effect of yi on zj (i, j = 1,2,3). The indirect effects of psychological distress (y1), fatigue (y2), and side effects (y3) on QOL-Social (z1), QOL-Mental (z2), and QOL-Physical (z3) can be calculated with the following formula:

$$\begin{cases} \text{Indirect Effect}(y_1 \rightarrow z_i) = r_{12} \times \beta_{2i} + r_{13} \times \beta_{3i} \\ \text{Indirect Effect}(y_2 \rightarrow z_i) = r_{12} \times \beta_{1i} + r_{23} \times \beta_{3i} \\ \text{Indirect Effect}(y_3 \rightarrow z_i) = r_{23} \times \beta_{2i} + r_{13} \times \beta_{1i} \\ i = 1,2,3 \end{cases}$$

Statistical analysis was performed using SAS (Version 9.0) for Windows (SAS Institute, Inc., Cary, NC).Methodology

3 Research Results

In total, 970 participants, including 699 (72.06%) males and 271 (27.94%) females at an average age of 56.38 years (SD = 12.91), were included in this study. The percentages of participants with primary school, middle school, high school, and college education levels were 27.61%, 30.97%, 24.87%, and 16.54%, respectively. Among the 970 participants, 338 (34.84%) had esophageal cancer, 374 (38.56%) had gastric cancer, and 258 (26.60%) had colon cancer; in addition, 122 (12.56%), 343 (35.36%), 316 (32.58%), and 189 (19.48%) were in stages I–IV of their respective diseases. Moreover, 750 (77.32%) patients underwent surgery combined with chemotherapy, 85 (8.76%) underwent surgery combined with radiotherapy, and 135 (13.92%) underwent surgery combined with chemotherapy and radiotherapy. The average time of the first treatment cycle was 4 weeks.

Two assumptions are depicted in the model as follows. One assumption was that psychological distress, fatigue, and treatment side effects were the main factors influencing QOL. To verify the correctness of this assumption, we analyzed the relationships between QOL as the dependent variable and psychological distress, fatigue, and side effects as the independent variables using multiple linear regression. The results revealed an adjusted R2 value of 0.524 for the model, indicating that psychological distress, fatigue, and side effects collectively could explain 52.4% of the variance in QOL and thus confirming that our assumption was appropriate. The other assumption in the model was that resilience had no direct effect on QOL. To verify the correctness of this assumption, we set resilience as the independent variable and QOL as the dependent variable in the regression model. The results showed that the regression coefficient of resilience, adjusted for age, sex, disease stage, psychological distress, fatigue, and side effects, was not statistically significant ($t = 1.562$, $P = 0.119$), indicating that resilience had no direct effect on QOL. Based on the above-described analysis, we concluded that the model was appropriate.

After adjusting for age, sex, and disease stage, the partial correlation coefficient between psychological distress and fatigue was $r12 = 0.440$ ($P < 0.001$), that between psychological distress and side effects was $r13 = 0.246$ ($P < 0.001$), and that between fatigue and side effects was $r23 = 0.178$ ($P < 0.001$).

The standardized coefficient for each path in the path map was estimated, and the standardized coefficients adjusted for age, sex, and disease stage are shown. The square of the standardized coefficient of resilience revealed that it could explain 33.2% of the variance in psychological distress and 16.1% of that in fatigue. These results suggest that resilience is an important factor that affects both psychological distress and fatigue.

The standardized coefficients were written into the path map. The direct and indirect effects of resilience, psychological distress, fatigue, and side effects on QOL were calculated according to Equations (3) and (4). This table shows that psychological distress and fatigue had greater effects on the three dimensions of QOL. The direct effects of fatigue were the largest for QOL-Social and QOL-Physical, the direct effect of psychological distress was the largest for QOL-Mental, and side effects mainly influenced QOL-Physical.

By summing the direct effects on the three domains of QOL, we obtained the direct effects of psychological distress, fatigue, and side effects on QOL. By summing the indirect effects on the three domains of QOL, we obtained the indirect effects of psychological distress, fatigue, side effects, and resilience on QOL. The proportions of direct and indirect effects on QOL for psychological distress, fatigue, side effects, and resilience are shown in the fifth and sixth columns. Fatigue, psychological distress, and side effects accounted for 48.32%, 29.71%, and 21.97%, respectively, of the total direct effect on QOL. Of the total effect on QOL, fatigue accounted for 33.72%, psychological distress accounted for 28.94%, side effects accounted for 22.53%, and resilience accounted for 14.80%. These results suggest that psychological distress and fatigue produced in the course of treatment are important factors influencing the QOL of patients. Although resilience has a lower proportion of the total effect on QOL, it has significant effects on fatigue and psychological distress.

4 Analysis & DIscussion

Resilience refers to an individual's capacity to maintain his or her psychological and physical well-being in the face of adversity. Resilience can be viewed as a defense mechanism that enables one to thrive amid distress. Therefore, improving resilience may be an important target for disease treatment and prophylaxis [26]. Patients with cancer can show high levels of functioning in physical domains of QOL but not in others, suggesting that an individual's capacity to adjust and cope will influence his or her QOL. Individual differences in resilience cause patients to have different coping styles and adjustment capacities [5]. Therefore, it is necessary to introduce the concept of resilience into studies of the QOL of cancer patients.

QOL is an indicator of a patient's social, psychological, and physiological status and well-being [1]. In theory, resilience affects the psychological aspect of QOL [12, 16] and thus should have a direct effect on QOL. However, in this study, the regression coefficient of resilience (adjusted for age, sex, and disease stage) was statistically significant ($t = 4.499$, $P < 0.001$) when psychological distress, fatigue, and treatment side effects were absent from the regression model; the reverse was true ($t = 1.562$, $P > 0.05$) when these variables were added. These results suggest that the effect of resilience on QOL may be passed on by psychological distress, fatigue, and side effects and is therefore indirect. Further studies are necessary to confirm this conclusion.

In this study, we analyzed the patients in the following order: resilience psychological distress and fatigue and side effects QOL; that is, resilience was plotted on the left part of the path map, which means that if patients with low resilience can be identified early and are given good social support as well as psychological care, then their psychological distress will likely decrease. This would prompt them to actively respond to treatment-induced fatigue and side effects, thereby improving their QOL.

In summary, the data obtained by our epidemiological survey showed that although resilience is not an independent predictor of QOL in patients with digestive cancer and accounted for only 14.80% of the total effect on QOL, it is a main influencing factor of psychological distress and side effects. In addition, psychological distress and fatigue are important factors that affect QOL, indicating that the role of resilience in improving QOL cannot be ignored. In studying the QOL of patients with cancer, we should focus on strategies that improve their resilience.

5 Conclusion & Suggestions

Background

Resilience is an individual's capacity to maintain his or her psychological and physical well-being in the face of adversity. Cancer is a disease causing severe psychological distress of the patients. Exploring association between resilience and QOL can make us understand the role of resilience in improving the QOL of cancer patients, as well as provide clinical staff with information on psychological intervention and psychological care programs for cancer patients.

Research frontiers

In recent years, the role of resilience in the process of cancer treatment has been given increasing attention. Studies have found that resilience can powerfully predict patients' fatigue in the treatment, and good resilience can help patients reduce treatment-induced damage to bodily functions and shorten the time of their recovery thereof. The researchers suggested that patients with good resilience are able to treat their disease correctly and maintain relatively good mental and psychological states, thereby resulting in a better QOL.

Innovations and breakthroughs

Although much research has shown that a relationship exists between QOL and resilience in cancer patients, limited information is available on the nature of this relationship and the degree of the influence of resilience on QOL. This study detected the relationships of resilience, psychological distress, fatigue, and treatment side effects with QOL.

Applications

Our results indicate that the role of resilience in improving QOL cannot be ignored. In studying the QOL of patients with cancer, we should focus on strategies that improve their resilience.

6 References

[1] Bottomley A. The cancer patient and quality of life. Oncologist 2002; 7:120-5.

[2] Efficace F, Bottomley A, van Andel G. Health related quality of life in prostate carcinoma patients: a systematic review of randomized controlled trials. Cancer 2003; 97:377-88.

[3] Andersen BL. Quality of life for women with gynecologic cancer. Curr Opin Obstet Gynecol 1995; 7:69-76.

[4] Arora NK, Gustafson DH, Hawkins RP, McTavish F, Cella DF, Pingree S, Mendenhall JH, Mahvi DM. Impact of surgery and chemotherapy on the quality of life of younger women with breast carcinoma: a prospective study. Cancer 2001; 92:1288-1298.

[5] Joanne L, Christine E. Exploring links between the concepts of quality of life and resilience. Pediatric Rehabilitation 2001; 4(4):209-216.

[6] Epping-Jordan JE, Compas BE, Osowiecki DM, et al. Psychological adjustment in breast cancer processes of emotional distress. Health Psychology 1999; 18(4):315-326.

[7] Yi JP, Vitaliano PP, Smith RE, et al. The role of resilience on psychological adjustment and physical health in patients with diabetes. British Journal of Health Psychology 2008; 13:311-325.

[8] Richardson GE. The metatheory of resilience and resiliency. Journal of Clinic Psychology 2002; 58:307-321.

[9] Pearman T. Quality of life and psychosocial adjustment in gynecologic cancer survivors. Health and Quality of Life Outcomes 2003, 1:33.

[10] Bull AA, Meyerowitz BE, Hart S, et al. Quality of life in women with recurrent breast cancer. Breast Cancer Research Treat 1999; 54:47-57.

[11] Wenzel LB, Donnelly JP, Fowler JM, et al. Resilience, reflection, and residual stress in ovarian cancer survivorship: A gynecologic oncology group study. Psycho-Oncology 2002; 11:142-153.

[12] Strauss B, Brix C, Fischer S, et al. The influence of resilience on fatigue in cancer patients undergoing radiation therapy. Journal of cancer research and clinical oncology 2007; 133:511-518.

[13] Hou Wk, Law CC, Yin J, Fu YT. Resource loss, resource gain, and psychological resilience and dysfunction following cancer diagnosis: A growth mixture modeling approach. Healthy Psychology 2010; 29(5):484-495.

[14] Costanzo ES, Ryff CD, Singer BH. Psychosocial adjustment among cancer survivors: Findings from a national survey of health and well-being. Health Psychology 2009; 28(2):147–156.

[15] Yng HC, Thornton LM, Shapiro CL, Andersen BL. Surviving Recurrence: Psychological and Quality-of-life Recovery. Cancer 2008; 112 (5): 1178-1187.

[16] Bull AA, Meyerowitz BE, Hart S, et al. Quality of life in women with recurrent breast cancer. Breast Cancer Res Treat 1999; 54:47-57.

[17] Antoni MH, Goodkin K. Host moderator variables in the promotion of cervical neoplasia-personality facets. J Psychosom Res 1988; 32:327-338.

[18] Wagnild GM. The Resilience Scale user's guide for the US English version of the Resilience Scale and the 14-Item Reselience Scale (RS-14). Montana: The Resilience Center, 2009.

[19] Ahern NR, Kiehl EM, Sole ML, Byers J. A review of instruments measuring resilience. Issues in Comprehensive Pediatric Nursing 2006; 29(2):103–125.

[20] Nishi1 D, Uehara R, Kondo M, Matsuoka Y. Reliability and validity of the Japanese version of the Resilience Scale and its short version. BMC Research Notes 2010; 3:310.

[21] Zigmond AS, Snaith PR. The hospital anxiety and depression scale. Acta Psychiatrica Scandinavica 1983; 67: 361-370.

[22] Leung CM, Ho S, Kan CS, Hung CH, Chen CN. Evaluation of the Chinese version of the Hospital Anxiety and Depression Scale. Across-cultural perspective. Int J psychsom 1993; 40: 29-34.

[23] Smets EM, Garssen B, Bonke B, De Haes JC. The Multidimensional Fatigue Inventory (MFI) psychometric qualities of an instrument to assess fatigue. J Psychosom Res 1995; 39: 315-325.

[24] Tian J, Hong JS. Validation of the Chinese version of Multidimensional Fatigue Inventory-20 in Chinese patients with cancer. Supportive Care in Cancer 2012, 20(10):2379-83.

[25] Zhen LC, Tian HR, Xie PZ. Medical Quality of survival Evaluation. Beijing: Junshi Yixue Kexue Press, 2000.

[26] Davydov DM, Stewart R, Ritchie K, Chaudieu I. Resilience and mental health. Clinical Psychology Review 2010; 30:479 – 495.

[27] Tian J, Chen ZC, Hang LF. Effects of nutritional and psychological status of gastrointestinal cancer patients on tolerance of the cancer treatments. World Journal of Gastroenterology 2007; 13(30): 4136-4140.

[28] Tian J, Chen ZC, Hang LF. The Effects of psychological status of the patients with digestive system cancers on prognosis of the disease. Cancer Nursing 2009; 32(3):230-235.

[29] Tian J, Chen ZC, Hang LF. Effects of nutritional and psychological status of the patients with Advanced Stomach cancer on Physical Performance Status. Supportive Care in Cancer 2009 ;17(10):1263-1268.

SESSION

POSTER PAPERS

Chair(s)

TBA

Advances in Bioinformatics and Computational Biology: Don't take them too seriously anyway.

Emanuel Diamant
VIDIA-mant, Kiriat Ono, Israel

Abstract - *In the last few decades or so, we witness a paradigm shift in our nature studies – from a data-processing based computational approach to an information-processing based cognitive approach. The process is restricted and often misguided by the lack of a clear understanding about what information is and how it should be treated in research applications (in general) and in biological studies (in particular). The paper intend to provide some remedies for this bizarre situation.*

Keywords: Information, Physical information, Semantic information, Bioinformatics.

1 Introduction

Striking advances in high-throughput sequencing technologies have resulted in a tremendous increase in the amounts of data related to various biological screening experiments. Consequently, that gave rise to an urgent need of new techniques and algorithms for analyzing, modeling and interpreting these huge amounts of data.

To reach this goal, Computational Biology and Bioinformatics techniques and tools are being devised, developed and introduced into research practice.

What is the difference between the two? Wikipedia does not see any difference at all [1]. NIH working definition, [2], distinguishes only a slight disparity between them:

"Computational biology uses mathematical and computational approaches" (to reach its goals), while "Bioinformatics applies principles of information sciences and technologies" (for the same purposes).

Perhaps the most evident difference lies in their historical background. Computational biology starts when the "brain as a computer" metaphor becomes generally accepted as the dominant research paradigm. Therefore, almost all scientific fields have become "computational" – Computational neuroscience, Computational genomics, Computational chemistry, Computational ecology, Computational linguistics, Computational intelligence, and so on. It was acknowledged then that the surrounding world is represented by data that is sensed by our sensors and thus processing of this data (making computation on it) was accepted as the prime duty of the research community.

At the same time, it was acknowledged that human interaction with the external world can be seen as a communication process by which sensory data is delivered to the conscious mind. For such a case, Shannon's "Mathematical Theory of Communication", [3], and the Information Theory embedded in it have been developed and become the dominant research paradigm of the second half of the past century. Obviously, this was the ground on which Bioinformatics has emerged and has gained its recognition as a separate research field.

However, Shannon's information is restricted only to data communication issues. Message meaning (semantics) – a crucially important part of a communication process – is totally omitted from its considerations. That explains the visible similarity between Computational Biology and Bioinformatics – both are first of all busy with data processing, at the same time, both are deficient in dealing with information issues (due to the lack of understanding about the essence of information).

The intention of this paper is to attempt to clarify the existing confusion.

2 So, what is information?

A proper definition of the term "information" does not exist. Therefore, I would like to propose my own one. It is an extended version of the Kolmogorov's mid-60s definition [4], which can be now expressed in the following way:

"Information is a linguistic description of structures observable in a given data set".

A digital image would serve us as a testbed for definition analysis. An image is a two-dimensional set of data elements called pixels. In an image, pixels are distributed not randomly, but due to the similarity in their physical properties, they are naturally grouped into some clusters or clumps. I propose to call these clusters **primary or physical data structures**.

In the eyes of an external observer, the primary data structures are further arranged into more larger and complex assemblies (usually called "visual objects"), which I propose to call **secondary data structures**. These secondary structures reflect human observer's view on the primary data structures composition, and therefore they could be called **meaningful or semantic data structures**. While formation of primary data structures is guided by objective (natural, physical) properties of the data, ensuing formation of secondary structures is a subjective process guided by human habits and customs.

As it was already said, **Description of structures observable in a data set should be called "Information".** In this regard, two types of information must be distinguished – **Physical Information and Semantic Information**. They are both language-based descriptions; however, physical information can be described with a variety of languages (recall that ma-

thematics is also a language), while semantic information can be described only by means of the natural human language. (More details on the subject can be find in [5]).

Every information description is a top-down evolving coarse-to-fine hierarchy of descriptions representing various levels of description complexity (various levels of description details). Physical information hierarchy is located at the lowest level of the semantic hierarchy. The process of sensor data interpretation is reified as a process of physical information extraction from the input data, followed by an attempt to associate the physical information at the input with physical information already retained at the lowest level of a semantic hierarchy. If such association is reached, the input physical information becomes related (via the physical information retained in the system) with a relevant linguistic term, with a word that places the physical information in the context of a phrase, which provides the semantic interpretation of it. In such a way, the input physical information becomes named with an appropriate linguistic label and framed into a suitable linguistic phrase (and further – in a story, a tale, a narrative), which provides the desired meaning for the input physical information.

3 New wine in old wineskins

In the light of the above elucidation, the mutual interrelations between Computational Biology and Bioinformatics can be now explained and put into action: Essentially, Computational Biology is an attempt to mimic physical information descriptions while Bioinformatics is an attempt to mimic semantic information descriptions. Now, all further advances in their development have to take into account the integration-dissociation peculiarities and task division strategy following from the new information definition.

Let me put it again: semantic perception of the sensed data begins with physical information extraction from it. It must be emphasized that only physical information is being processed further in the semantic information-processing stream. All physical traits of the input data are lost at this stage. In the end, we understand the essence of an image ignoring its illumination conditions or color palette. The same is with speech perception – we understand the meaning of a phrase independent of its volume or gender voice differences.

The extracted physical information is associated then with the physical information retained at the lowest level of the semantic hierarchy. In such a way, it finds its place in a linguistic expression, which determines its meaning, its semantics. (Analogous to "comprehension from usage" or "understanding from action" forms of semantics disambiguating).

This physical data structures naming is in a close resemblance to the ontology-based annotation process. Ontologies are the most recent form of knowledge representation and are widely used in biomedical science enabling to turn data into knowledge. Despite of the resemblance, semantic information hierarchies and ontologies are strikingly different. From my definition of semantic information follows that 1) knowledge is memorized (retained in the system) information (and nothing else!), 2) semantic information is an observer's property, and 3) semantic information has nothing to do with data! That is, data is semantics devoid. So, the purpose of ontologies "to describe the semantics of data", is misinterpreted. Computational biology tools developers have to pay more attention to this peculiarity.

4 Conclusions

One can hardly overestimate the importance of physical and semantic information segregation. For the first time, data-based information and its semantic (language-based) interpretation are detached and now can be treated correctly and essentially.

For the first time, information is represented as a linguistic description, as a string of words, a piece of text. It does not matter that in biotic applications these texts are written in the four-letter nucleotide alphabet. The important thing is that now information is **materialized**, and as such can be stored, retrieved, changed, transmitted and (generally speaking) processed as any other material object.

In this regard, the paradigm shift from data-based computational approach to information-based cognitive approach receives its proper theoretical underpinning, which will certainly promote its further development and utilization.

One of the obvious problems that arises in such a transition is as follows: We are accustomed to use computers in our everyday life. Computer is a data processing devise. Semantic information comes about as a text string. Therefore, semantic information processing must be treated as text strings processing. But that is not what our computers are supposed to do. There is an urgent need to invent a new generation of computers that will be capable to process natural language texts (which are the expression of semantic information).

5 References

[1] Computational biology**,** From Wikipedia, the free encyclopedia
http://en.wikipedia.org/wiki/Computational_biology
[2] NIH working definition of bioinformatics and computational biology,
http://www.bisti.nih.gov/docs/compubiodef.pdf
[3] Shannon, C., Weaver, W. The Mathematical Theory of Communication, University of Illinois Press, 1949.
http://raley.english.ucsb.edu/wp-content/Engl800/Shannon-Weaver.pdf
[4] Kolmogorov, A. Three approaches to the quantitative definition of information, Problems of Information and Transmission, Vol. 1, No. 1, pp. 1-7, 1965.
http://alexander.shen.free.fr/library/Kolmogorov65_Three Approaches-to-Information.pdf
[5] Diamant, E. Brain, Vision, Robotics, and Artificial Intelligence. http://www.vidia-mant.info

Feedback and Coupled Feedback Loop Identification within KEGG and Reactome Pathway Database

Yin-Jie Liu[1], Chen-Lung Liu[1], Tun-Wen Pai[1*], and Wen-Shyong Tzou[2], and Chin-Hwa Hu[2§]
[1]Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung 20224, Taiwan, R.O.C., [*]twp@mail.ntou.edu.tw
[2]Department of Bioscience and Biotechnology, National Taiwan Ocean University, Keelung 20224, Taiwan, R.O.C., [§]chhu@mail.ntou.edu.tw

Abstract – *Each gene regulatory network (GRN) involves a set of DNA components interacting with each other directly or indirectly to control gene expression levels. Several sub-networks appear in a GRN holding feedback loop or coupled feedback loop mechanisms, which possess returning control mechanisms and aim to intervene or enhance certain functional conditions. Many cases have been reported and experimentally verified the existence of feedback loop relationships in transcription regulatory networks. However, most feedback loops were neither directly shown nor easy to be recognized from public pathway databases. In this study, we integrated two well-known pathway databases of KEGG and Reactome by parsing biological pathway maps into a common standard format, and the developed system could identify hidden feedback and coupled feedback loop information automatically through cross-database and cross-map analyses. We integrated and verified all possible cross-regulatory pathways from these two databases by examining each gene appeared in all collected pathway maps. Over 1,500 biological pathway maps were parsed and examined comprehensively, and an on-line web based system was designed for biologists to query any specified gene for its possible feedback relationships. In addition, paralogous gene information was considered for discovering potential coupled feedback loop relationships among various pathways. The built system for identifying hidden cross-linked relationships could facilitate biologists a fast and easy way to discover how many unveiled dynamic patterns of gene regulation maintain the balance of a specific biological function.*

Keywords: feedback loop, coupled feedback loop, biological pathway, gene regulatory network, orthologous, paralogous

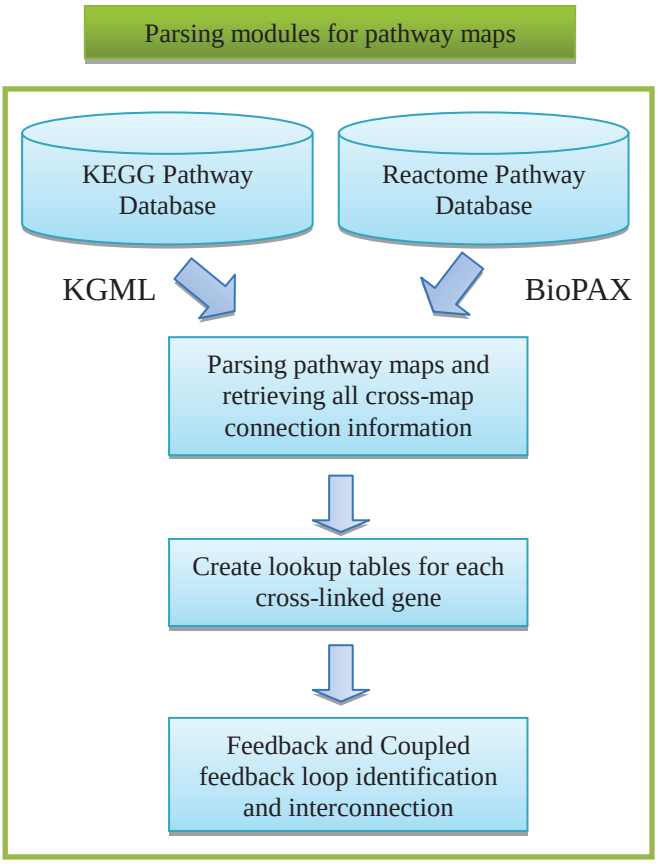


Figure 1:. System Flowchart of the proposed feedback loop and coupled feedback loop identification system.

In this study, we have designed a system for identifying all feedback loops and coupled feedback loops between the KEGG and Reactome pathway databases. The system flowchart is depicted in Figure 1. Based on the developed system, we have successfully detected 15,875 feedback loops by integrating these two pathway databases, of which 15,788 cross-map pairs were retrieved within KEGG, 24 cross-map pairs found in Reactome, and 63 cross-database feedback loops between KEGG and Reactome. All these 15,875 feedback loops cannot be found directly by observing any single map in these two databases. In the other words, integration of these two pathway databases and cross-map identification could reveal novel gene regulation relationships for biological relevant research. Especially, insufficient pathway network components and lack of feedback control relationship might lead to wrong interpretations for related biological experiments. To the best of our knowledge, this is the first work to integrate two different pathway databases and verify all feedback loop relationships among all pathway maps. The comprehensive analysis and reconstruction of feedback loop relationships could provide and enhance the completeness of pathway databases, and this can be extended to an even larger scale with an efficient mechanism. According to the cross-database evaluation, exclusive properties of pathway components for both KEGG and Reactome databases exist in most gene regulating connections.

When both KEGG and Reactome databases were integrated simultaneously, the system could detect 63 feedback loops through cross-database approaches. These detected 63 feedback loop pairs could be carefully and individually identified. All these detected pairs hold a common feature that all of them possessed compensated conditions. For example, the entry of identified gene pair (EP300, IFNB) possesses a hidden cross-link relationship from two databases. The type of this feedback loop relationship is represented as " EP300 → IFNB(Reactome) & EP300 ← IFNB(KEGG) with a relationship of Negative (R-K)". The (R-K) stands for Reactome-KEGG which means the gene EP300 has at least one path to regulate gene IFNB observed from Reactome and there is a reverse path for IFNB regulates EP300 observed from KEGG. These two reverse paths can only be found by considering these two databases simultaneously. Featuring only one pathway database might lead to insufficient supporting evidences and wrong conclusions. Through integrating two or more pathway databases simultaneously, users could upgrade to discover comprehensive regulation relationships between any two gene pair.

In Table 1, we listed another 4 cross-database coupled feedback loops with direct regulation relationships by integrating both KEGG and Reactome databases. In fact, these cross-map coupled results were obtained by analyzing the feedback loops appeared within both datasets individually. In the other words, there are a total of 8 feedback loops with direct regulation relationship in KEGG, and 10 feedback loops with direct regulation in Reactome, respectively. We combined these 18 feedback loops with direct regulation to validate and identify the cross-database coupled feedback loops. There are 4 observed coupled feedback loops belonging to Positive-Negative (PN) type.

Table 1. Coupled feedback loops by integrating KEGG and Reactome databases.

Coupled Feedback Loops(CFLs)	Types of CFL
PGC1-NR1D1 & NR1D1-ARNTL	PN
PGC1-NR1D1 & NR1D1-CLOCK	PN
NR1D1-ARNTL & NR1D1-CLOCK	PN
PIK3C-PTK2 & PIK3R-PTK2	PN

As the previously shown results, there were a total of 15,875 feedback loops by integrating two well-known databases, of which 63 feedback loops must be identified through cross-database approaches. These identified feedback loop information could not be intuitively observed through any existing bioinformatics tools. Our developed system could facilitate biologists an efficient and effective way to discover invisible relationships from the current pathway maps and to perform advanced researches on gene regulation networks. For the future work, integrating KEGG and Reactome pathway databases won't be satisfied for all biologists, since there exist several well-known pathway resources worldwide. How to integrate extra pathway databases and keep updating the newest information will be the next big challenge for this research field.

References:

[1] S. Banerjee and I. Bose, "Functional characteristics of a double positive feedback loop coupled with autorepression," Phys Biol, vol. 5, p. 046008, 2008.
[2] J. R. Kim, Y. Yoon, and K. H. Cho, "Coupled feedback loops form dynamic motifs of cellular networks," Biophys J, vol. 94, pp. 359-65, Jan 15 2008.
[3] H. Ogata, S. Goto, K. Sato, W. Fujibuchi, H. Bono, and M. Kanehisa, "KEGG: Kyoto Encyclopedia of Genes and Genomes," Nucleic Acids Res, vol. 27, pp. 29-34, Jan 1 1999.
[4] D. Croft, G. O'Kelly, G. Wu, R. Haw, M. Gillespie, L. Matthews, M. Caudy, P. Garapati, G. Gopinath, B. Jassal, S. Jupe, I. Kalatskaya, S. Mahajan, B. May, N. Ndegwa, E. Schmidt, V. Shamovsky, C. Yung, E. Birney, H. Hermjakob, P. D'Eustachio, and L. Stein, "Reactome: a database of reactions, pathways and biological processes," Nucleic Acids Res, vol. 39, pp. D691-7, Jan 2011.

Comparative analysis of two bacterial binding lectins

Sim-Kun Ng[1], Yi-En Chen [1], Sheng-Hao Chou [2], Jiun-Hau Fu[1], Tse-Kai Fu[1], Tung-Kung Wu[2], and Margaret Dah-Tsyr Chang[1*]

[1]Institute of Molecular and Cellular Biology & Department of Medical Science, National Tsing Hua University, Hsinchu, Taiwan, Republic of China, [2]Department of Biological Science & Technology, National Chiao Tung University, Hsinchu, Taiwan, Republic of China

Abstract - *Tachypleus plasma lectin 2 (TPL2) derived from Taiwanese Tachypleus tridentatus recognizes spceific bacteria and shows a 76% sequence identity with another bacteria binding lectin, Tachylectin-3 (TL-3), derived from Japanese T. Tridentatus. Neither secondary nor tertiary structure of these two bacteria binding lectins has ever been solved yet. In this study TPL2 and TL-3 structures were predicted to possess two separate domains by Phyre[2] database. Several ligand binding sites containing hydrophilic aromatic residues were predicted by feature-incorporated alignment (FIA) and molecular docking, and selective ligand binding residues in TPL2 were verified by in vitro assays. Interestingly, TPL2 and TL-3 showed differential ligand binding preference and activities, which might be correlated with minor differences in their structural and functional features.*

Keywords: Bacteria binding lectin, structure prediction, ligand binding site, hydrophilic aromatic residue

1 Introduction

Lectins are a group of proteins that recognize specific sugar moieties and are widely distributed in various organisms. Lectins perform many different biological functions such as cell surface receptors, mediators of signaling, and recognition molecules against pathogen. In invertebrate such as horseshoe crab, lectin plays important roles in the immune system. In Japanese horseshoe crab *Tachypleus tridentatus*, 6 types of lectins have been identified, namely Tachylectin-1 (TL-1) to -4 from hemocytes and TL-5A and -5B from plasma. They have been investigated to recognize specific pathogens through pathogen-associated molecular patterns (PAMPs).

Tachypleus plasma lectin 2 (TPL2) was isolated and characterized as novel hemolymph protein from Taiwanese *T. tridentatus* [1]. TPL2 binds to 3 species of bacteria, including *Streptococcus pneumoniaee R36A, Vibrio parahaemolyticus* and *Escherichia coli Bos-12*. Recently, TPL2 is also found to bind to selective medically important pathogens isolated from clinical specimens, such as *Pseudomonas aeruginosa*, *Klebsiella pneumonia*, and *Streptococcus pneumoniae* serotypes through molecular interaction with a specific sugar moiety, rhamnose (Rha), in PAMPs on pathogen surface [2]. In comparison with Japanese horseshoe crab lectins TPL2 shares low sequence identity with most TLs except TL-3 [3], and both show ligand specificity toward lipopolysaccharides (LPSs), particularly *O*-antigen. However, neither secondary nor tertiary structures are solved yet.

In this study differential features in primary sequences, putative secondary and tertiary structures, and ligand binding sites of TPL2 and TL-3 were analyzed using integrated *in silico* and *in vitro* methodologies. Functional characterization of putative ligand binding residues indicated that selective hydrophilic aromatic residues played crucial roles in differential ligand binding activities of these two horseshoe crab lectins.

2 Results

Here primary sequence of TPL2 was aligned with TL-3 using Clustal W2 (http://www.ebi.ac.uk/Tools/msa/clustalw2/) as shown in **Figure 1**. The sequence identity between TPL2 (AAF74774.1) and TL3 (BAA75214.1) is 76%. Besides, secondary structures of these lectins were predicted using Protein Homology/analogY Recognition Engine V 2.0 (Phyre[2], http://www.sbg.bio.ic.ac.uk/phyre2/html) [5]. Quite similar secondary structural elements, 4 α-helices and 9 β-sheets in TPL2, and 5 α-helices and 6 β-sheets in TL-3 were observed. In addition to *N*-terminal and *C*-terminal ends, major difference was located in loop α1-β5 in TPL2 and loop α1-α2 in TL-3 where only 60% sequence identity was observed. As aromatic residues surrounded by polar residues within two amino acids in neighboring sequences, namely "hydrophilic aromatic residues" (HARs), are reported to involve in glycan ligand binding [4], sequence analysis revealed that 5 HARs in TPL2 (Y46, W49 in loop α1-β5, and F101, F103, W105 in loop α3-α4) and TL-3 (Y42, W45 in loop α1-α2, and F97, F99, W101 in loop α3-α4) were quite conserved and possibly served as ligand binding sites (**Figure 1**, gray shadow). The 5 HARs of TPL2 were further site-directed mutated into Ala to investigate functional roles of selective HARs. As expected, bacteria binding ELISA showed that TPL2(Y46A) and TPL2(F103A) binding activities to *Pseudomonas aeruginosa* were dramatically reduced (data not shown), strongly indicating that these two residues served as ligand binding sites in good agreement with *in silico* prediction.

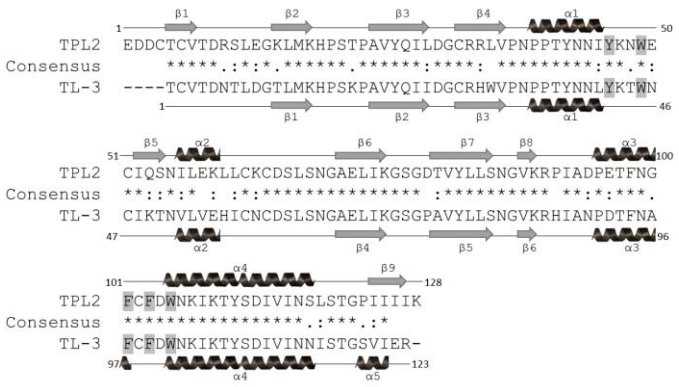


FIGURE 1: Primary and secondary structure alignment between TPL2 and TL-3. Amino acid sequences of TPL2 and TL-3 were aligned using Clustal W2. Fully conserved, highly similar, and weakly similar amino acids were respectively indicated by asterisk (*), colon (:), and dot (.). Secondary structures predicted by Phyre[2] database were shown as ribbons and arrows. Hydrophilic aromatic residues (HARs) in loop regions were labeled with gray shadow.

Furthermore, putative 3D structures of TPL2 and TL-3 were also predicted by Phyre[2]. Both modeled structures of TPL2 and TL-3 were predicted to form two separated and equal-size domains, D1 and D2 (**Figure 2**). $TPL2_{D1}$ and $TL\text{-}3_{D1}$ adopted similar structural fold such that each domain comprised 3 β-sheets capped on each side by a short helix (**Figure 2A** and **2E**). However, $TPL2_{D1}$ had one more β-sheet between α1 and α2, and the α2 in $TPL2_{D1}$ was longer than that of $TL\text{-}3_{D1}$. Coincidentally, $TPL2_{D2}$ and $TL\text{-}3_{D2}$ also adopted similar structural fold (**Figure 2B** and **2F**), but $TL\text{-}3_{D2}$ possessed a long loop between α3 and α4. The long loops in $TL\text{-}3_{D1}$ and $TL\text{-}3_{D2}$ might accommodate appropriate orientation of HARs and governed ligand binding specificity. To further compare ligand binding profiles of two glycans, Rha, a TPL2 binding ligand, and blood group A-pentasaccharide, a TL-3 binding ligand, were docked *in silico* by AutoDock Vina program and docking poses with the lowest binding energy were displayed (**Figure 2**). Here binding energy of Rha to $TPL2_{D1}$ and $TPL2_{D2}$ was estimated to be -4.47 and -4.74 kcal/mol, respectively (**Figure 2A** and **2B**). Likewise, binding energy of Rha to $TL\text{-}3_{D1}$ and $TL\text{-}3_{D2}$ was in the same order of magnitude as respectively -3.9 and -5.0 kcal/mol (**Figure 2E** and **2F**), strongly suggesting that TL-3 might also bind Rha. As for A-pentasaccharide, the binding energy to $TPL2_{D1}$, $TPL2_{D2}$, $TL\text{-}3_{D1}$, and $TL\text{-}3_{D2}$ was respectively -0.79, -0.78, +0.31 and -1.91 kcal/mol (**Figure 2C**, **2D**, **2E**, and **2F**). Binding energy of A-pentasaccharide to $TL\text{-}3_{D2}$ appeared to be the lowest in four structures, hence TL-3 might bind to A-pentasaccharide through $TL\text{-}3_{D2}$. High binding energy implied weak interaction between A-pentasaccharide and lectin, which was clearly validated with our *in vitro* bacteria binding ELISA showing that up to 100 μM A-pentasaccharide could not inhibit binding between TPL2 and *P. aeruginosa* [2].

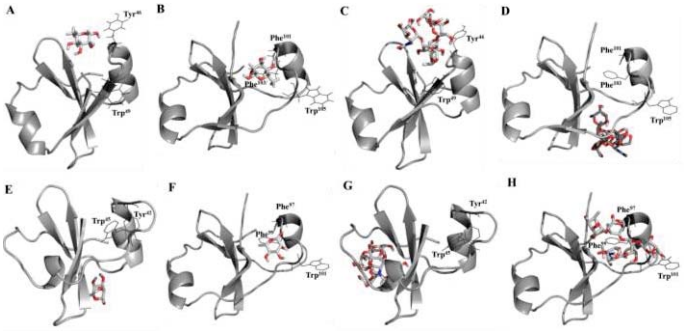


FIGURE 2: Molecular docking of Rha and A-pentasaccharide in TPL-2 and TL-3. Rha and A-pentasaccharide were docked in predicted structures by AutoDock Vina program and docking poses with the lowest binding energy were displayed. The predicted structures were shown in cartoon and the ligands were presented as stick. (A) $TPL2_{D1}$-Rha. (B) $TPL2_{D2}$-Rha. (C) $TPL2_{D1}$-A-pentasaccharide. (D) $TPL2_{D2}$-A-pentasaccharide. (E) $TL\text{-}3_{D1}$-Rha. (F) $TL\text{-}3_{D2}$-Rha. (G) $TL\text{-}3_{D1}$-A-pentasaccharide. (H) $TL\text{-}3_{D1}$-A-pentasaccharide.

3 Conclusion

Integrated of *in silico* prediction and *in vitro* assays have demonstrated that minor differences in primary sequences of evolutionarily related TL-3 and TPL2 contribute to different structural and functional features, which in turn give rise to differential application potentials of these bacteria binding lectins. This work was sponsored by Ministry of Science and Technology grant 103-2627-M-007 -006 to M.D.-T. Chang.

4 References

[1] S. T. Chiou, Y. W. Chen, S. C. Chen, C. F. Chao, and T. Y. Liu, "Isolation and characterization of proteins that bind to galactose, lipopolysaccharide of Escherichia coli, and protein A of *Staphylococcus aureus* from the hemolymph of *Tachypleus tridentatus*," *J Biol Chem*, vol. 275, pp. 1630-4, Jan 21 2000.

[2] S. K. Ng, Y. T. Huang, Y. C. Lee, E. L. Low, C. H. Chiu, S. L. Chen, *et al.*, "A recombinant horseshoe crab plasma lectin recognizes specific pathogen-associated molecular patterns of bacteria through rhamnose," *PLoS One*, vol. 9, p. e115296, 2014.

[3] K. Inamori, T. Saito, D. Iwaki, T. Nagira, S. Iwanaga, F. Arisaka, *et al.*, "A newly identified horseshoe crab lectin with specificity for blood group A antigen recognizes specific *O*-antigens of bacterial lipopolysaccharides," *J Biol Chem*, vol. 274, pp. 3272-8, Feb 5 1999.

[4] W. Y. Chou, W. I. Chou, T. W. Pai, S. C. Lin, T. Y. Jiang, C. Y. Tang, *et al.*, "Feature-incorporated alignment based ligand-binding residue prediction for carbohydrate-binding modules," *Bioinformatics*, vol. 26, pp. 1022-8, Apr 15 2010.

[5] L. A. Kelley and M. J. Sternberg, "Protein structure prediction on the Web: a case study using the Phyre server," *Nat Protoc*, vol. 4, pp. 363-71, 2009.

SESSION

BIOINFORMATICS, COMPUTATIONAL BIOLOGY, AND RELATED ISSUES + LATE BREAKING PAPERS AND POSITION PAPERS

Chair(s)

TBA

Improved and Novel Cluster Analysis for Bioinformatics, Computational Biology and All Other Data

Ruming Li [1], **Xiu-Qing Li** [2*], **and Guixue Wang** [3*]
[1, 2] Molecular Genetics Laboratory, Potato Research Centre, Agriculture and Agri-Food Canada
850 Lincoln Road, P.O. Box 20280, Fredericton, New Brunswick, E3B 4Z7, Canada
[3]Key Laboratory of Biorheological Science and Technology (Chongqing University), Ministry of Education;
Bioengineering College of Chongqing University, Chongqing, 400044, China

Abstract - *Cluster analysis has been widely used in bio-informatics or biology to classify objects (items, DNA bands, markers, genes, individuals, species, taxa, etc.). Theoretically, there are numerous clustering methods available but only those that are well-established and proven are commonly used in practice. For their improved applicability, those methods that exploit the most data information and yield the better cluster properties should be focused and new algorithms are expected. The other issues and problems that are relevant to the performance of cluster analysis also need to be addressed. The well-prepared data can be significant to ameliorate the clustering results. They include the proper measure conversion from similarities to dissimilarities or distances and the necessary data standardizing transformation. Apart from z-score standardization, average- or mean-based scaling was introduced to achieve comparability among variables with the least value manipulation. A solution was given to improve updating the distance matrix with a centroid-related equation. A novel method called the percent similarity cluster analysis (PSCA) was devised to analyze the DNA band or marker data from electrophoresis and all other data.*

Keywords: cluster analysis, clustering methods, clustering improvement, special clustering, data similarity and transformation, data scaling and standardization,.

1 Introduction

Cluster analysis is a classification technique that is used to divide a set of objects (aka, entities, cases, points, observations, samples, or items) into separate subsets or clusters (aka, classes, groups or partitions). In the object space, each object has a multi-dimensional space that consists of a set of variables (aka, attributes, characters, or dimensions). The aim is to partition objects into mutually exclusive clusters such that the members in a cluster are sufficiently similar to each other and sufficiently dissimilar to non-members in other clusters [1,2]. Clustering can group data into a previously unknown or preset number of homogeneous clusters [3,4]. The resulting clusters can be visualized in a dendrogram (a tree diagram) that displays the fusions or divisions made at each successive step of cluster analysis.

There has been a rapid growth in the literature so far that addresses cluster analysis with somewhat different concepts and arguments since 1960. A broad range of sciences have been involved in adopting these clustering techniques, with varied assumptions, settings, and outcome interpretations. And a growing number of software programs for performing cluster analysis and the formation of cliques of cluster analysis users also make its application diverse. In response to some issues of cluster analysis, it is expected that there should be a practical approach that integrates and optimizes the existent clustering methods based upon their well-elucidated mathematic theory. Since cluster analysis is carried out as related to inter-disciplinary realms, additional algorithms are needed to take care of data types that come from discipline-specific sources such as genomics in biology. Taking electrophoresis-aided banding technology for example, it has become the currently critical tool for us to ascertain significant information in biology as well as in other sciences. With this technology, a great volume of bands have spawned from PCR, DNA/RNA test, proteomics analysis and so on that provide a massive yet unique source of data [5,6]. Although DNA microarray-based gene expression data can be treated as quantitative and analyzed with a classic clustering method, electrophoresis-based band or marker data need to be otherwise tackled [7]. Unfortunately the traditional clustering algorithms are not adequate for all the questions in bioinformatics and computational biology at present. One of the questions, for instance, is the multiple "tied" objects that can be merged at once, rather than only two are joined at a time. There are certain inconsistencies among clustering algorithms and issues on some ambiguous notions that lead to different results. Other questions are that binary data processing by simple matching is unsuitable for band or marker data that can be multi-state, and that clustering a range of similar bands or markers by a similarity criterion is also unavailable in percent matching. This criterion in terms of percent similarity among bands or markers is due to the need of selectively classifying them. Particularly when the dimensionality of a data vector becomes very huge or the exactly percent matching is impractical to compute, clustering by a relaxed similarity criterion is necessary. When the input data type is quantitative and usually its magnitudes among the variables have unequal influence on the distance between objects, the conventional data standardization used is statistically debatable.

This study discussed all of these questions with major clustering methods and proposed some improved and new algorithms for classic cluster analysis from a practical perspective. In addition, we introduced the novel clustering method that is needed to deal with DNA band or marker data as well as other categorical data. The notation used throughout the paper will remain effective thereafter.

2 Data preparation and methods

The input data type is the first thing to be considered for a cluster analysis to be pragmatically applicable. There is a big data space in the real world that provides quite a few sources of analytical data; typically they fall into two major types: quantitative and categorical. In bioinformatics, for example, gene expression data can be analyzed in a quantitative mode, and some categorical (nominal or ordinal) variables such as binary data are also amenable to numerical analysis. A data set in coordinate form constitutes a data (or pattern) matrix such as DNA microarray-based genes (objects) and experimental conditions (variables) in gene expression profiles [8,9]. Clustering such a matrix may find gene expression patterns and functionally related genes, thereby suggesting the function of currently unknown genes. To perform such a cluster analysis, the similarity or proximity between gene expression profiles has to be properly measured. Usually this is measured in terms of Euclidean distance, as it provides the shortest length between two points in metric space. This distance is regarded as the most natural and also commonly used measure as compared to others. Since almost all clustering methods operate on Euclidean distance, it is assumed in this study.

For binary data, the absence-presence usually scores 0-1. For ternary band or marker data, it may use "w" to label the "weak" band in a lane from electrophoretic tests and has the trio of states 0, 1, and w. For quaternary band or marker data, it may use one more state label "u" to indicate the "unidentified" band and so forth. The distance or dissimilarity between objects (testing samples) *i* and *j* for these nominal data is measured by mismatch score or percent disagreement that is defined as

$$d(i, j) = 1 - \frac{n_{00} + n_{11}}{n} \quad \text{for binary (0-1) data}$$

$$d(i, j) = 1 - \frac{n_{00} + n_{11} + n_{ww}}{n} \quad \text{for ternary (0-1-w) data}$$

$$d(i, j) = 1 - \frac{n_{00} + n_{11} + n_{ww} + n_{uu}}{n} \quad \text{for quaternary (0-1-w-u) data}$$

where $0 \le d(i, j) \le 1$, n_{00}, n_{11}, n_{ww} or n_{uu} is the number of respective 0-0, 1-1, w-w, and u-u matches, and n is the number of all attributes (dimensions). Notice that the fractional term in each of the expressions is the simple matching coefficient and is complementary to $d(i, j)$; that is, its higher value corresponds to shorter distance or closer relationship. Practically, mismatch scores are better used for clusterings to safely retain the precision of results than the decimal $d(i, j)$, which will be addressed in the next section.

The hierarchical (agglomerative) cluster analysis is a major clustering method applicable to the analytical data. In the hierarchical clustering with data matrix input, the first step is to calculate the distance matrix for all possible pairs of objects. Usually all the elements in this matrix are the squared Euclidean distances between objects but also can be the $d(i, j)$ values computed from the above nominal data.

2.1 Similarity measure conversion

In addition to data matrix (raw data set), distance or dissimilarity and correlation-based similarity matrices also can be imported as special data types for cluster analysis. First of all, we need to clarify the notion of distance- and similarity-type data. That is, the former is the measure of a straight line in Euclidean space, whereas the latter is the measure of proximity or relationship between objects. The similarity-type data can be correlation measures (Pearson, Spearman rank, Kendall τ, etc.), or any other scores that measure the association or pattern resemblance between objects. In general, distances are the best bet to detect differences and correlations are often better to find similarities [10]. Those highly correlated values can be considered to be very similar to each other; as such, increasing similarities translates into decreasing distances or dissimilarities [3,11]. Prior to cluster analysis, all correlation-based similarities must be converted to dissimilarities (distance-type similarity or correlation-based distance). Just as distance is essentially a measure of dissimilarity, dissimilarity matrix here is a synonym for distance matrix in the sense that the minimum rather than the maximum is to be searched for in the matrix for clustering. There are a variety of ways to convert similarity measures, such as taking reciprocals or sub-tracting from the upper bound that is 1 as constrained by a cophenetic matrix [12]. A correlation-based distance or dissimilarity *d* usually is defined in terms of the correlation (or similarity) coefficient *r* (or *s*) as

$$d = 1 - r \quad \text{or} \quad d = 1 - s. \qquad (1)$$

It should be indicated that *r* or *s* be consistently positive. A negative *r* or *s* fails to give any information on distance measures due to its reverse direction of association even if all coefficients are negative. Thus, *r* or *s* takes on a value that ranges from 0 to 1 and cannot range from 0 to -1.

What is the right upper bound that converts from similarity measures to distances while retaining as much of the original information as possible? The use of 1 as the upper bound may not robustly achieve the best correspondence between a similarity matrix and its cophenetic matrix. Table 1 illustrates some extreme similarity coefficients that are closer to 1 and the resulting correspondence is poor due to the loss of signi-ficant decimal digits for very small values. If a computing program operates on three or more decimal digits, the extreme *r* or *s* (.999) will be converted to *d* (.001), resulting in the loss of more significant decimal digits and so on. What is worse, the poor precision of those very small values in the cophenetic matrix makes inaccurate the drawing of a dendrogram. Statistically, the correlation coefficient *r* is the amount that explains the association of one variable with another and is meaningful only in this respect. However, the expression *1-r* is not as interpretable and uniquely significant as the *r* is. From this perspective, there is no constraint that 1 is the only upper bound for conversion purposes. In essence, the criterion to achieve the best correspondence is the upper bound that results in the minimum loss of significant decimal digits for converted values.

From a practical point of view, if r_{max} or s_{max} is the maximum r or s in a correlation or similarity matrix and p is a percentage, then d in expression (1) is re-defined by

$$d = (1 + p)\, r_{\max} - r \qquad \text{or} \qquad d = (1 + p)\, s_{\max} - s \qquad (2)$$

where p equals 10%, which is proposed but can be other desired percentage. This expression not only ensures sufficiently significant decimal digits for converted values but it also eliminates data-dependent inconsistency on a proportional basis.

Note that, as with expression (1), expression (2) does not satisfy the triangle inequality as well. Indeed, none of the correlation-based distances follows this constraint; this is the general property of the correlation coefficient. Unlike the Euclidean distance that is a true metric and does satisfy the triangle inequality, the correlation-based distance is sometimes thought of as semi-metric. Nevertheless, the similarity measure conversion is not such constrained because it merely turns the coefficients into other corresponding values in their mirror data space. It merely re-quantifies, in other way, the differentiation between distances using the same scale as the correlation coefficients in order to depict them in a dendrogram. What is more, the use of expression (2) leads to an improved and practically acceptable cluster analysis. As shown in Table 1, for instance, the extreme r or s (.99) will be converted to d (.10) by (2) rather than to d (.01) by (1). The significant digits of .10 begin from the first decimal place and hence retain two significant digits, whereas that of .01 begins from the second decimal place and retains only one significant digit. Thus, the former contains more of the original information about r or s than does the latter. With the upper bound, expression (2) also can handle the conversion of similarity matrices whose elements are not coefficients ($s \geq 1$) such as match scores, proximity counts or similarity ranks.

Table 1. A comparison of values in the correlation or similarity matrix and its cophenetic matrix converted by expression (1) and expression (2).

O	Original values (r or s)				d: converted by (1)				d: converted by (2)			
	1	2	3	4	1	2	3	4	1	2	3	4
1	.00				.00				.00			
2	**.99**	.00			**.01**	.00			**.10**	.00		
3	.97	.86	.00		.03	.14	.00		.12	.23	.00	
4	.94	**.99**	.95	.00	.06	**.01**	.05	.00	.15	**.10**	.14	.00

2.2 Data standardizing transformation

The advantage of correlation measures is that they are generally not influenced by differences in scale between objects. On the other hand, distance measures are significantly affected by differences in scale across variables. From a data matrix, the distance between objects is determined by the sum over all differences of paired variables. If these variables are on different scales, their varying sizes would contribute differently to the distance. To balance the relative importance among the different variables and make multi-dimensional variation comparable, these variables should be transformed such that they are on a common scale. This scaling is necessary because information from each variable needs to be fairly, unbiasedly reflected in determining the distances with no artifacts. That is, without considering the principal components, all variables should contribute equally to the distance between objects. There can be a few transformation schemes: mean, median, maximum, range, and variance scaling that equalize overall variables to achieve unit-measure homogeneity.

Since cluster analysis generally is a nonparametric, descriptive method and has no distributional assumptions unless they are confirmed. In reality, it is difficult to find all the variables following one distribution, especially if the dimensionality is very huge. Therefore, any forced variance scaling (i.e., z-score standardization) based on the multivariate normality is not recommended. From this standpoint, the criterion to judge a sound transformation is how well it retains information of original data. It is something like saying how well it maintains as much "fidelity" of the raw data as it could. Of the five schemes, the mean, median, or maximum scaling imposes the least manipulation on raw data and therefore preserves the most information about it. Dividing a variable by its maximum appears to best scale it to 0-1 range but taking outlier into account would make this scaling less desirable, as the extremely large maximum could scale values to very small ones. Dividing a variable by its median is well-known for tackling outliers but that is better used in that case and is not always the case for all variables. In general, the arithmetic mean is the best data representative (with centrality) because its calculation exploits all values of data and hence reflects all information on a variable. Suppose x_{ij} is the observation and X_{ij} is the scaled one for the ith object and jth variable. Let $\bar{x}_{.j}$ be the arithmetic mean, $\tilde{x}_{.j}$ be the median, and $x_{\max\text{-}j}$ be the maximum for the jth variable. These three transformations are defined below by the mean, median, and maximum scaling, respectively:

$$X_{ij} = \frac{x_{ij}}{\bar{x}_{.j}}, \qquad X_{ij} = \frac{x_{ij}}{\tilde{x}_{.j}}, \qquad X_{ij} = \frac{x_{ij}}{x_{\max-j}}.$$

The difference between the resultant unit mean and unit maximum would be considered to be different consistencies over variables. That is, although the mean scaling does not standardize data to 0-1 range as the maximum scaling does, it still scales them to the consistent differences across variables. This is because all variables use invariably their means to do the scaling. The transformed data are not required virtually to fall within the 0-1 range as long as they are equitable and comparable among variables on a unit-measure basis [13].

The soundness of the mean scaling can be shown by the property that the difference of any two mean-scaled values for variable j between objects i and k is equal to the difference of their non-scaled values divided by the common mean of variable j. Namely,

$$X_{ij} - X_{kj} = \frac{x_{ij}}{\bar{x}_{.j}} - \frac{x_{kj}}{\bar{x}_{.j}} = \frac{x_{ij} - x_{kj}}{\bar{x}_{.j}}.$$

This ensures that the contribution of each variable to the distance between objects is adjusted proportionally by its mean, regardless of the differences in scale. In Table 2, the equality (1:1) across variables is achieved at the high end for maximum -scaling and at the middle for mean-scaling. The former has the better inter-variable comparability only on one side near the

high end, while the latter has the better comparability on two sides near the middle. Mean- and maximum-scaling have a shiftable relationship, as their equality points can be shifted. Nevertheless, taking equality at the middle makes transformed values more widely and stably comparable than at one end. That is, with the representativeness (centrality) of the average, the mean scaling is robust to varying transformation, while maximum-scaling is prone to extreme transformation. By this nature, mean-scaled values retain the more characteristics of raw data and have the more equitable influence on the distance. For gene expression data that fail in the test of multivariate normality due to the more attributes in the tissue types, time series, and/or treatment conditions, the mean scaling is more sound than the forced z-score standardization.

Table 2. A demonstration of the terminal equality (one-side comparability across variables) obtained from maximum- scaling and the central equality (two-side comparability across variables) obtained from mean-scaling.

O	Raw data matrix		By maximum scaling		By mean scaling	
	Variable 1	Variable 2	Variable 1	Variable 2	Variable 1	Variable 2
1	0.0	5.0	0.00	0.56	0.00	0.71
2	1.0	6.0	0.25	0.67	0.50	0.86
3	2.0	7.0	0.50	0.78	**1.00**	**1.00**
4	3.0	8.0	0.75	0.89	1.50	1.14
5	4.0	9.0	**1.00**	**1.00**	2.00	1.29

2.3 Handling tied data and mergers

The first issue on ties in cluster analysis is the tied objects that could be encountered at level 1 of clustering. Handling this scenario is to merge all tied objects and then treat them as one object. That is, use any of the tied objects as a representative and include it in the participating objects, and the other tied objects are excluded from the remaining cluster analysis. The reason for this elimination is that the tied objects provide nothing more of distinctive data vectors and they are just simple redundancies and should count once in data processing; that is, one copy alone from duplicates suffices. Since the objects excluded fail to contain new meaningful information, they do not make a difference in the subsequent clusterings. The weights they may provide only do not improve but bias the cluster analysis in most cases unless weighted contribu-tions are under study.

The second issue is that the tied mergers may appear at each clustering level. The treatment in this scenario is to accept all tied mergers at each level [14] even if possibly merging all objects/subclusters into a single cluster at very few levels. The reason for this is that the principal aim of cluster analysis is obtaining all proper clusterings; logically, the earlier level they are partitioned at, the more reasonable clusters they are revealed as. If the analyst feels uncomfortable with this way, the first one of tied mergers ordered or sorted in the distance matrix would be taken.

3 Clustering results

3.1 Quantitative hierarchical cluster analysis

For a sound clustering methodology that could exploit as much information as being contained in the data, we focused on the average linkage (UPGMA), centroid method (UPGMC), and Ward's minimum-variance method [11,15-17]. Their underlying computational algorithms and pragmatic iterative implementations were provided.

In the average linkage, let the capital D stand for a squared Euclidean distance, the lowercase d for a Euclidean distance (the square root of D) between objects/clusters, and the combinatorial pq for the merger of clusters p and q. Suppose i is any other cluster, n is the number of objects in a cluster, k is the kth object in the merger, and l is the lth object in cluster i. It has been verified that only the following two combinatorial forms of computation of D are equivalent to each other:

$$D(pq, i) = \frac{1}{n_{pq} n_i} \sum_{k=1}^{n_{pq}} \sum_{l=1}^{n_i} D(k,l) \quad (1)$$

$$D(pq, i) = \frac{n_p}{n_{pq}} D(p,i) + \frac{n_q}{n_{pq}} D(q,i) \quad (2)$$

where formula (1) uses all paired objects between pq and i, and the update equation (2) uses the previous-step D for separate p or q with i and hence gains implementation efficiency. They both can take care of either data or distance/similarity matrix input but equation (2) is actually adopted. This method has no issue or ambiguity on its mathematic derivation and practical application, although there is a weighted variant (WPGMA).

In the centroid method, let $\overline{X}$ be a cluster centroid (or mean vector). It has been verified that only the following two combinatorial forms of computation of D are equivalent to each other:

$$D(pq, i) = \| \overline{X}_{pq} - \overline{X}_i \|^2 \quad (3)$$

$$D(pq, i) = \frac{n_p}{n_{pq} + n_i} D(p,i) + \frac{n_q}{n_{pq} + n_i} D(q,i) - \frac{n_p n_q}{(n_{pq} + n_i)^2} D(p,q) \quad (4)$$

where formula (3) directly employs the centroids to update the distance between the merger and any other cluster, and the update equation (4) is used for efficient implementation. The former can only be used with data matrix input, whereas the latter can deal with both data and distance/similarity matrix input and is adopted in practice.

There are ambiguous ways of calculating a cluster centroid, each resulting in a likely different quantities of distance. In Table 3, the $d1_{pq}$ is calculated by taking the mean over two previous centroids, whereas the $d2_{pq}$ takes the mean over all objects in a merger, for each step. The difference between the $d1_{pq}$ (= 6.0) and $d2_{pq}$ (= 6.3) illustrates this discordance. Only the way that employs all objects in a merger to get $d2_{pq}$ is considered to be rational for a true centroid (a real barycenter). Another validation of this as the right way is that it has been proved by the equivalence of two-form computations for $d(pq,$

i), while using the other way cannot yield such equivalence. This other way is regarded as weighted and therefore should be explicitly called the median (WPGMC) method [18].

Table 3. The stepwise cluster centroids (*CC*) and distances (*d*) calculated by the two previous-step centroids and by all the objects in a merger.

#	Partitions of Clusters	$CC1_p$	$CC1_q$	$d1_{pq}$	$CC2_p$	$CC2_q$	$d2_{pq}$
5	(1) (2) (3) (4) (5)	—	—	—	—	—	—
4	**(1, 2)** (3) (4) (5)	[1.0, 1.0]	[1.0, 2.0]	1.0	[1.0, 1.0]	[1.0, 2.0]	1.0
3	(1, 2) (3) **(4, 5)**	[8.0, 2.0]	[8.0, 0.0]	2.0	[8.0, 2.0]	[8.0, 0.0]	2.0
2	(1, 2) **(3, 4, 5)**	[6.0, 3.0]	[8.0, 1.0]	2.8	[6.0, 3.0]	[8.0, 1.0]	2.8
1	**(1, 2, 3, 4, 5)**	[1.0, 1.5]	[7.0, 2.0]	**6.0**	[1.0, 1.5]	[7.3, 1.7]	**6.3**

In Ward's method, suppose *i* is the *i*th cluster for *l* clusters, *j* is the *j*th variable for *m* variables, and *k* is the *k*th object for *n* objects in a cluster. The error sum of squares (*ESS*) within the *i*th cluster has been given by

$$ESS_i = \sum_{j=1}^{m}\sum_{k=1}^{n}(X_{ijk}-\bar{X}_{ij.})^2 . \quad (5)$$

From (5), letting ESS_{pq} be *ESS* within the merger cluster, the minimum increase in *ESS* (ΔESS_{pq}) has the following relationships with others:

$$\Delta ESS_{pq} = ESS_{pq} - ESS_p - ESS_q = \frac{n_p n_q}{n_p + n_q} ||\bar{X}_p - \bar{X}_q||^2 \quad (6)$$

upon which the metric criterion of distance between clusters is based, as the clustering proceeds.

Let $\Delta ESS_{pq.i}$ be the minimum increase in *ESS* as clusters *pq* and *i* merge. It has been verified that the following two combinatorial forms of computing ΔESS are equivalent to each other only if D' is a previous-step ΔESS:

$$\Delta ESS1_{pq.i} = \frac{n_{pq} n_i}{n_{pq} + n_i} ||\bar{X}_{pq} - \bar{X}_i||^2 = \frac{n_{pq} n_i}{n_{pq} + n_i} D(pq, i) \quad (7)$$

$$\Delta ESS2_{pq.i} =$$

$$\frac{n_p + n_i}{n_{pq} + n_i} D'(p,i) + \frac{n_q + n_i}{n_{pq} + n_i} D'(q,i) - \frac{n_i}{n_{pq} + n_i} D'(p,q) . \quad (8)$$

Let ESS_T be the total error sum of squares over all clusters, ESS'_T be the previous-step ESS_T for a cumulative operation, and *ESS(pq, i)* be ESS_T as clusters *pq* and *i* merge. It has been verified that only the following three combinatorial forms of computation of ESS_T derived from Ward's method are equivalent to one another:

$$ESS(pq, i) = ESS_T = \sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n}(X_{ijk}-\bar{X}_{ij.})^2$$

when *p* and *q* merge (9)

$$ESS(pq, i) = ESS_T = ESS'_T + \Delta ESS1_{pq.i} \quad (10)$$

$$ESS(pq, i) = ESS_T = ESS'_T + \Delta ESS2_{pq.i} \quad (11)$$

where formula (9) for *ESS* originates from the primary Ward's algorithm, the direct update equation (10) employs centroids that is related to ESS_T from (7), and the update equation (11) is used for efficient implementation from (8). The variant (10) can only be used with data matrix input, whereas the variant (11) can deal with both data and distance/similarity matrix input and is adopted in practice.

To implement the Ward's algorithm, the simple way is to convert all squared Euclidean distances (*D* values) in the distance matrix to the minimum increment ΔESS_{pq} at the first step of iteration. First, this conversion is required to use the update equation (8) in which $D'(p, i)$, $D'(q, i)$, and $D'(p, q)$ must be the previous-step $\Delta ESSpi$, $\Delta ESSqi$, and $\Delta ESSpq$, respectively. Second, all elements in the distance matrix must be of the type ΔESS rather than the type *D* in order to be comparable and be searched for the minimum increase in *ESS*. For this reason, such elements must be either all ΔESS values or all *D* values and may not be of discordant type. To use the ΔESS distance matrix, its element simply takes half of the *D* value, as $n_p n_q/(n_p+n_q)$ becomes ½ when any two objects merge initially (n_p=1, n_q=1). However, this way needs to convert all *D* values in the distance matrix and is thought to have the extra cost of computation.

To cope with this problem and keep using the *D* distance matrix without having to use the ΔESS distance matrix, the solution is to utilize the update equation (4) in the centroid method afore-mentioned. Since calculating a cluster centroid is related to ΔESS in Ward's method, updating the distance matrix can be done by equation (4), rather than by equation (8).

From equation (7), we have

$$\Delta ESS_{pq.i} = \frac{n_{pq} n_i}{n_{pq} + n_i} D(pq, i) \quad (12)$$

where $\Delta ESS_{pq.i}$ is directly proportional to *D*(*pq*, *i*) and hence searching for the minimum ΔESS amounts to searching for the minimum *D* in the respective distance matrix.

Equalize the squared Euclidean distance *D(pq, i)* on the right part of the expression with formula (3), which is in turn equal to equation (4), in the centroid method. Then substituting (4) into (12) yields

$$\Delta ESS_{pq.i} =$$

$$\frac{n_{pq} n_i}{n_{pq} + n_i}\left[\frac{n_p}{n_{pq} + n_i} D(p,i) + \frac{n_q}{n_{pq} + n_i} D(q,i) - \frac{n_p n_q}{(n_{pq} + n_i)^2} D(p,q)\right]$$

(13)

This equation (13) provides a solution to using equation (4) instead of classic (8) and preserves all the desired properties that it remains capable of updating a distance matrix, that it retains direct use of squared Euclidean distances without having to convert them otherwise, and that it can handle both data and distance/similarity matrix input.

There is an inconsistency with the criterion of distance between clusters given by ESS_T, by ESS_{pq} plus the previous-step value, or by ΔESS. Because ESS_{pq} contains ESS_p and ESS_q, it is not a net increase in *ESS* due to the fusion of clusters *p* and *q*. From the stepwise increment of *ESS* (Table 4), one can see that the distance criterion should be given by ΔESS_{pq}–based ESS_{T2} rather than ESS_{pq}–based ESS_{T1}. Another validation of ESS_T as the distance criterion is that it has been proved by the

equivalence of three-form computations for ESS_T, while using ESS_{pq} cannot yield such equivalence. It is unreasonable for ΔESS *per se* to be the distance criterion in that the primary Ward's algorithm takes care of ESS_T, not ΔESS produced by the merger alone. ESS_T measures overall distance, while ΔESS measures inter-centroid distance as it is closely related to D values as revealed by formula (7). The former gives non-centroid distance by accumulation (monotonic increase), while the latter functions more likely as a centroid method. To distinguish Ward's method (with monotonicity) from a centroid-like method (without monotonicity), only ESS_T is treated as the real Ward's criterion of distance between clusters.

Table 4. The stepwise error sums of squares (ESSs) for the merger ESS_{pq}, the incremental ΔESS_{pq}, and the respective total ESS_Ts.

#	Partitions with Minimum ESS_T	ESS_{pq}	ΔESS_{pq}	ESS_{T1}	ESS_{T2}	ESS_{T3}
6	(1) (2) (5) (7) (9) (10)	0.00	0.00	0.00	0.00	0.00
5	**(1, 2)** (5) (7) (9) (10)	0.50	0.50	0.50	0.50	1.00
4	(1, 2) (5) (7) **(9, 10)**	0.50	0.50	1.00	1.00	1.00
3	(1, 2) **(5, 7)** (9, 10)	2.00	2.00	3.00	3.00	4.00
2	(1, 2) **(5, 7, 9, 10)**	14.75	12.25	**17.75**	**15.25**	**24.50**
1	**(1, 2, 5, 7, 9, 10)**	67.33	52.08	67.33	67.33	104.17

Whatever forms these formula or update equations may take, they are the sums of squared deviations from mean vectors. Thus it makes sense that they should be restored to original metrics (i.e., square roots) after being subject to squaring. This not only provides a real measure in Euclidean space but also makes it consistent and comparable with those obtained by other clustering linkages. Moreover, the usage of Euclidean distance improves cluster representation as well in a dendrogram by scaling down the longer distances between clusters and scaling up the shorter ones (Fig. 2). This is because the distance could be inflated by squaring a greater-than-one node value in a dendrogram as shown in Figure 1. Particularly, the square root of decimal figures gives the larger values and hence removes the distortion caused by squaring the decimal distances. Therefore, the practical criterion of distance or similarity between clusterings should be given by Euclidean distance rather than by squared one. Namely,

$$d(pq,i) = \sqrt{ESS(pq,i)} = \sqrt{ESS_T}\,.$$

3.2 Categorical hierarchical cluster analysis

Categorical variables such as the foregoing nominal attributes (dichotomous or multi-state) can be subjected to numerical analysis if they turn into a dissimilarity matrix by mismatch score or percent disagreement. The clustering methods above are applied to such data as well. Here we introduced a new hierarchical clustering method that allows for analysis of DNA band, marker, or other nominal data. With such clusterings, data need not turn into a dissimilarity matrix. This method is called the percent similarity cluster analysis (PSCA) and used primarily in such categorical settings.

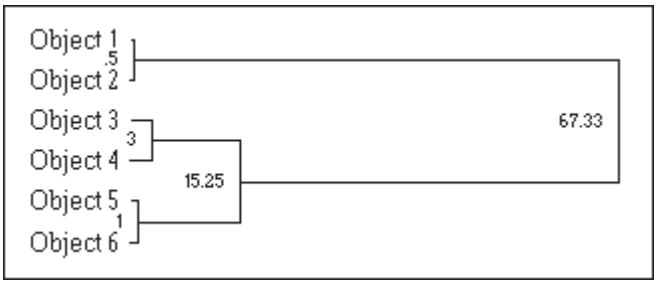


Fig. 1. The tree branch distance could be elongated by the squared greater-than-one node value (67.33) and shortened by the squared decimal node value (0.5). This causes the distorted graphic representation of clusters in the dendrogram.

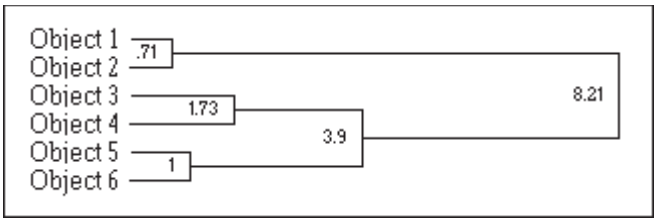


Fig. 2. The square root of the decimal distance gives a larger node value (0.71) and the square root of the greater-than-one distance gives a smaller node value (8.21) from Fig. 1.

This removes the distortion of the tree diagram caused by squaring operation on the distances as compared to Figure 1.

DNA band or marker data from electrophoresis consist of biological items (objects) each of which has a band vector whose elements (band values taking 0, 1, w, or u) are nominal attributes in a data matrix. They are analyzed through comparison of each pair of corresponding band values between items and those items showing a certain or higher percent band similarity are clustered. It is utilized to discover those banding patterns where identical, similar, and distinct bands are identified. Further, they are used for a comparative study of molecular identities such as the biological variety appraisal by DNA fingerprint clustering.

The underlying algorithm for PSCA is: From the first item, make all possible comparisons among items until get the first cluster in which members meet a mutual similarity criterion. Then from the second item (if clustered go to the next), make all possible comparisons among the remaining items until get the second cluster with the same mutual similarity criterion met. Iterate it until get the last-run cluster at the first level of clustering. Afterward, all the analogous comparisons for the next level are made against the relaxed similarity criterion and based on either clusters or singletons. The similarity criteria are from 100% down to10%, relaxed by 10 (the default but can be

specified otherwise); so there are up to 10 (or other) levels of hierarchy. At level 1, each item is a cluster center for com-parison. After each level, the number of cluster centers may be reduced due to the merger of clusters. For a merger, any element of its cluster center that does not match the cor-responding band value of any member is marked. Clustering proceeds with comparison of the elements of cluster centers, which become less matchable after each fusion, until the last level (0% similarity) is reached. The tied items will not get weighted, as they fail to offer new information on the identity of an item. Figure 3 illustrates a local portion of complex results of PSCA using the 10-band-per-item data from electrophoresis of 8 items with RT-PCR in our DNA fingerprinting.

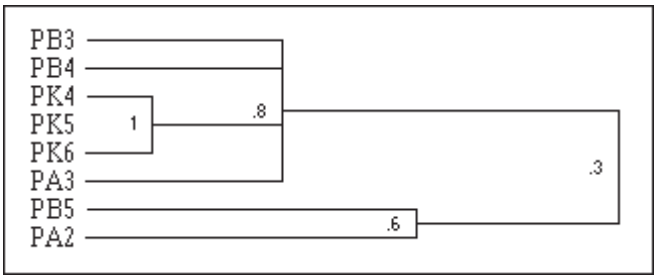


Fig. 3. An illustration of part of the complex results from the percent similarity cluster analysis (PSCA) in the dendrogram. PK4, PK5 and PK6 are identical. PB3 through PA3 are 80% similar. PB5 is 60% similar to PA2. Entirely, all the items are 30% similar to one another.

4 Applicability and discussion

In the PSCA method, it is remarkable that the clustering result differs from the one by a classic clustering method that merges traditionally paired objects/clusters based on the minimum dissimilarity criterion. With PSCA, it merges multiple items or subclusters and only those that are consistently similar to one another are clustered based on the criterion that is alterable by level. In Figure 4, for instance, the bands for item 1 have a 90% similarity with the bands for items 2, 3, 4, and 5. However the continuing pairwise comparisons from item 2 through item 5 turn out that item 3 is not consistently similar to all other items. So it will be excluded from this group of similar items and only items 1, 2, 4, and 5 are clustered. That is, only those items that have shown criterion-met mutual similarities are deemed to form a cluster. This clustering is applied to discovering banding patterns and interpreting their relations before we get the insight by other proofs. The labels "w" and "u" in the multi-state band or marker data are thus clustered for the convenience of later parallel pattern recognition, image processing, and data mining.

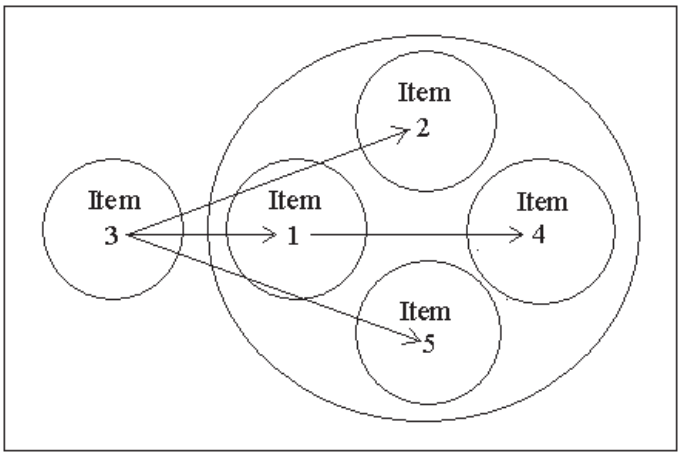


Fig. 4. A configuration of 5 items showing similarity linkages where items 1, 2, 4, and 5 exhibit closeness to one another on a similar band basis after all paired comparisons are made. Item 3 is closer to item 1 alone and is far from the distant items due to dissimilarity, which translates that there is no consistent similarity or homogeneity among all items. Thereby, item 3 will be excluded and only items 1, 2, 4, and 5 are clustered.

Cluster analysis is an unsupervised learning technique that could lead to different results via numerous approaches. There-fore, the choice of the right methodology is critical to the researcher. For hierarchical clustering, only the average, centroid, and Ward's linkage are focused in that they safely result in no undesirable cluster properties generated by the use of little or one-sided data information. Taking for example the chaining effect produced by single linkage, it is often criticized because objects being similar at one end of a cluster may be markedly dissimilar at the opposite ends. Likewise complete linkage also has no control of the resulting cluster shape and doesn't employ information from all member objects and tends to produce chaining clusters in the tree. Since clustering is a kind of collective or group behavior, not an individual behavior, a single object is not informative enough to reflect the entire cluster structure. Therefore, a between-cluster clusterability determined by a single object is not sound and reasonable, which makes its results difficult to interpret. These two methods usually perform well in cases when the objects actually form naturally distinct clumps in the data space. Instead of relying on extreme values as in single or complete linkage, one uses the average, centroid, and minimum variance to link clusters, which not only gains robustness but also warrants reliability. These linkages take the entire cluster structure into account and employ information from all member objects to determine a between-cluster clusterability; so its results are reasonable and interpretable. Generally such methods are less sensitive to noisy data, and outliers are not given any special favour in the cluster decision. Their resulting clusters have compact, spherical shapes where all members of a cluster tend to be tightly bound together. It is nevertheless advisable to acquire cluster information that is not based upon a particular algorithm and that should be objective-dependent. Indeed, there is no such thing as a single correct or desirable classification. To be safe in

cluster analysis, there is no better practice than one that removes outliers and has missing values retrieved.

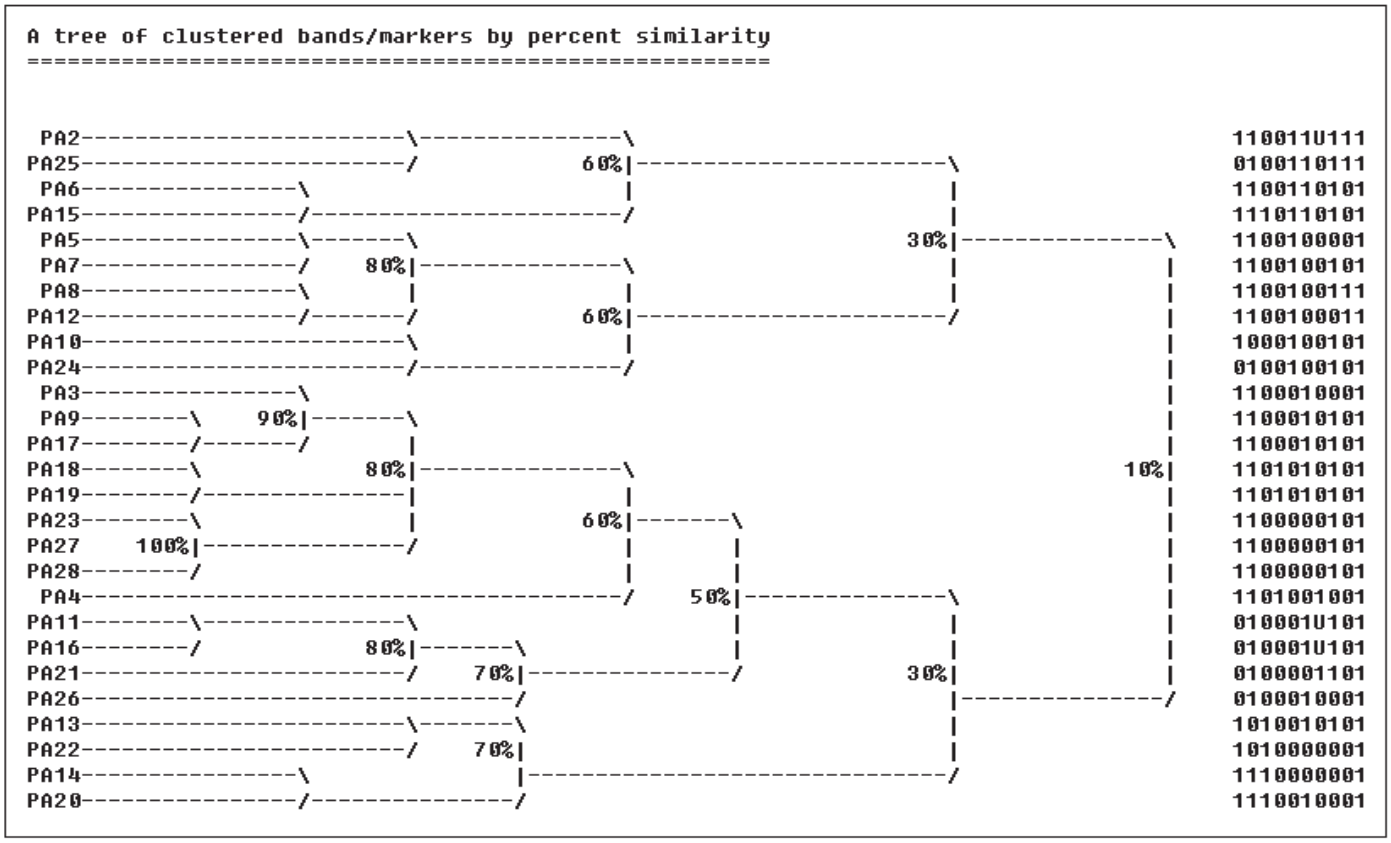


Fig. 5. A demonstration of text-based special clusterings in addition to graph-based special clusterings, to the right of which is a listing of DNA band data that is used to check with the corresponding entries of biological items/varieties (taxa) on the left side. With this textual tree view, each item on the left side corresponds to its 10 bands on the right side for a convenience of studies and manipulation. The letter "U" in the band data listing stands for an "unidentified" band. For a 0-1 band data system, another letter that may appear is "W" to label the "weak" band in a lane from electrophoretic testing. A band data range can be extended to any other data such as amino acid sequences, etc.

The average linkage (UPGMA) is practicable and superior to the weighted variant (WPGMA) in that all objects receive equal weights in the computation, which conforms to the equal contribution of each object to the distance. The centroid linkage (UPGMC) is as preferable to the median linkage (WPGMC) as UPGMA is to WPGMA. The median linkage does not really imply an outlier-proof algorithm as it suggests literally, and is better used when unequal cluster sizes (the numbers of objects) are treated as biased.

Since centroid clustering is not monotonic it may produce reversals of the levels in the dendrogram. This reversal is regarded as a violation of the ultrametric property but it is true only with respect to graphic representation of clusters and is sensible to numeric operation on centroids. That is, the length of a branch of a tree corresponds to the inter-centroid distance, a shorter length or a reversal can be generated by the shorter distance from level to level. It is possible for the inter-centroid distance to be non-monotonic as a result of subtracting operation between such centroids even if they are monotonically increasing. Therefore, the result from centroid clustering is still interpretable, regardless of reversals.

The PSCA method can be used to gather information on mutual similarities or interrelations of DNA bands or marker data and the like, and to discover banding patterns in data structures and layouts from electrophoresis. The results are most applicable to identifying the most likely genotype of an unknown organism via DNA fingerprints. It can be employed as well in bioinformatics studies and to discriminate between items (e.g., samples and specimens) that produce bands or markers. Moreover, the patterns it reveals are useful in exploring a potential biological relatedness or affinity among the items (e.g., species) [19].

All improved algorithms, solutions and novel methods for cluster analysis covered in this paper have been implemented in our earlier software BioCluster for Windows and the latest ParCluster2.0 (to be separately published). They are available

upon request (rli@alumni.lsu.edu) or from some web sites. In ParCluster v.2.0, special clusterings in addition to the PSCA method also are available (Fig. 5).

5 References

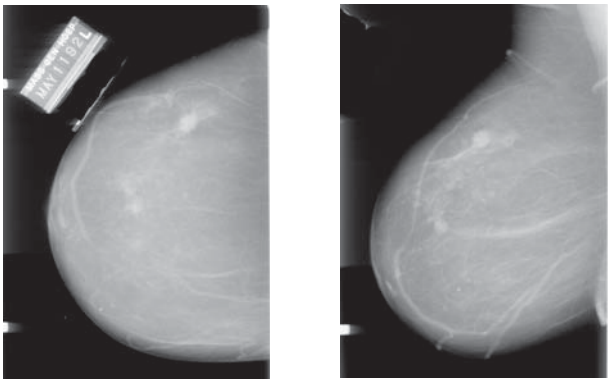
[1] Everitt, B. S. Cluster Analysis. 2nd ed. Heinemann Educ. Books, London, 1980.
[2] Li, R., X. Li, and G. Wang. Global optimal and minimal solutions to K-means cluster analysis. Proceedings of The 19th International Conference on Image Processing, Computer Vision, & Pattern Recognition (In press), 2015.
[3] Anderberg, M. R. Cluster Analysis for Applications. Academic Press: New York, 1973.
[4] Hartigan, J. A. Clustering Algorithms. John Wiley & Sons, New York, 1975.
[5] Figueroa, A., Borneman, J. and Jiang, T. Clustering binary fingerprint vectors with missing values for DNA array data analysis. *J. of Comput. Biol.*, 11(5): 887-901, 2004.
[6] Vickery, M. C. L., Harold, N. and Bej, A. K. Cluster analysis of AP-PCR generated DNA fingerprints of *Vibrio vulnificus* isolates from patients fatally infected after consumption of raw oysters. *Letters in Applied Micro-biology*, 30(3): 258–262, 2000.
[7] Belacel, N., Wang, Q. and Cuperlovic-Culf, M. Clustering methods for microarray gene expression data. OMICS: *A Journal of Integrative Biology*, 10(4): 507-531, 2006.
[8] Ben-Dor, A., Shamir, R. and Yakhini, Z. Clustering gene expression patterns. *J. of Comput. Biol.*, 6(3-4): 281-297, 1999.
[9] Ramoni, M. F., Sebastiani, P. and Kohane, I. S. Cluster analysis of gene expression dynamics. *Proceedings of the National Academy of Sciences of the United States of America*, 99(14): 9121-9126, 2002.
[10] Massart, D. L. and Kaufman, L. The interpretation of analytical chemical data by the use of cluster analysis. John Wiley & Sons, New York, 1983.
[11] Sneath, P. H. A. and Sokal, R. R. Numerical taxonomy: the principles and practice of numerical classification. W. H. Freeman: San Francisco. p. 573, 1973.
[12] Legendre, P. and Legendre, L. Numerical Ecology. 2nd English ed. Elsevier Science, Amsterdam, 1998.
[13] Stoddard, A. M. Standardization of measures prior to cluster analysis. *Biometrics*, 35(4): 765-773, 1979.
[14] Jardine, N. and Sibson, R. Mathematical Taxonomy. Wiley, New York, 1971.
[15] Lance, G. N. and Williams, W. T. A general theory of classificatory sorting strategies. I. Hierarchical systems. *Comput. J.*, 9: 373–380, 1967.
[16] Ward, J. H., Jr. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301): 236-244, 1963.
[17] Wishart, D. An algorithm for hierarchical classifications. *Biometrics*, 25(1): 165-170, 1969.
[18] Gower, J. C. A comparison of some methods of cluster analysis. *Biometrics*, 23(4): 623-637, 1967.
[19] Johnson, B. L. and Thein, M. M. Assessment of evolutionary affinities in Gossypium by protein electrophoresis. *Amer. J. Bot.*, 57(9): 1081-1092, 1970.

Classification of mass and non-mass mammography based on Tsallis entropy

Rafaela S. Alcântara[1] **and Perfilino E. Ferreira Junior**[1]
[1]Science Computer Department, Federal University of Bahia, Salvador, Bahia, Brazil

Abstract— *Breast cancer is a neoplasia that affects a high number of women in the world every year and it is the second on ranking of the diseases that more affects women. Mammography is the most important exam to aid in early detection and diagnosis of breast cancer. Computational methods have been developed to assist the radiologist in diagnosis and to improve the perception of the results. Feature extraction of mammography images has been allowed good results in classification of these mass. This paper presents a new approach of texture descriptors using multilevel decomposition and extracting Tsallis entropy from this new images and the best result for a vector of 24 features was 84.21% of accuracy.*

Keywords: breast cancer, mammography, classification, features, multilevel decomposition



(a) Cranio-Caudal (b) Medio-Lateral Oblique

Fig. 1: Mammography views extracted from DDSM database [6], [7]

1. Introduction

Nowadays breast cancer is the second most common cancer in the world. In 2012 about 1.67 millions new cases were registered in all the world and a rate of mortality of 522 thousands of deaths [1]. Breast cancer is a heterogeneous neoplasia therefore it can be developed through many factors as the heredity and lifestyle factors [2].

The number of breast cancer cases increased more than 960 thousands between 1980 and 2010. On the other hand, the rate of mortality has been increased in 175 thousands during the same period [3]. These statistics show that disease detection, diagnosis and treatment has been improved among these years. Early detection of breast cancer is a important step for a better treatment and screening mammography exam nowadays is considered the most effective method to provide the most accurate diagnosis for the patient [4] and the only one that decreases the rate of mortality [5].

The screening mammography is a gray-level radiography image used to assist the radiologist to get a better vision from the women breast. The exam can be done using two different views (Figure 1). Cranio-caudal view is obtained by a vertical vision from woman breast while medio-lateral oblique view is obtained by a predefined angle view from the breast.

These two views mammography image provide a plenty of information about breast tissue. Computer-aided detection (CADe) and computer-aided diagnosis (CADx) are systems developed to extract and analyze theses informations and provide a classification for the image or the region of interest (ROI). CADe systems provides information about the location of every suspicious mass and the CADx systems analysis theses masses from every location and provide the diagnosis. This work presents a CADx system based on statistical feature extraction of gray-level co-occurrence matrix (GLCM).

Texture descriptors from GLCM matrix are constantly used for breast cancer classification. In [8] a set of spatial diversity indexes were developed to compose a feature vector for mass and non-mass classification. The authors used the Support Vector Machine (SVM) for classification step and their method provided an accuracy of 99.7%. The diversity index calculated by Tsallis entropy has been used to increase threshold techniques for microcalcification detection [9], [10].

In this paper, a new CADx system was developed to compare a set of feature vectors based on Tsallis entropy extracted from GLCM matrix of all six multilevel decomposition images. Figure 2 presents our work-flow details.

1.1 Methodology

This section provides a brief resume of all concepts used in this paper as gray-level co-occurrence matrix (GLCM), multilevel decomposition and Tsallis entropy measure.

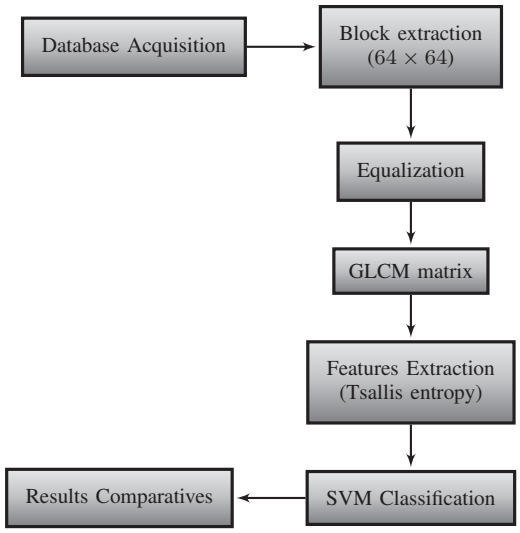


Fig. 2: Steps of the proposed approach

1.2 Gray-level co-occurrence matrix (GLCM)

Lets assume that $f(x, y)$ is a function representing the gray-level of one pixel in the position (x, y)

$$f : L_x \times L_y \rightarrow G \tag{1}$$

where L_x and L_y represents the horizontal and vertical dimensions of image f, respectively and G represents the number of gray tone levels of f. Gray-level co-occurrence matrix is defined as the angular relationship represented by P_{ij} between two neighbors pixels with gray-level i and j with a specific distance d and direction θ. Figure 3 presents an example of an image and the respectively GLCM matrix.

0	0	1	2
1	1	2	0
0	1	2	2
2	0	1	1

(a) Gray-levels of an image

	0	**1**	**2**
0	2	3	2
1	3	4	3
2	2	3	2

(b) GLCM of image (a)

Fig. 3: Co-occurrence matrix calculation example

Haralick et al. [11] proposed four values of distance (1,2,3 and 4) and direction and four variation for direction θ ($0°, 45°, 90°$ and $135°$). From these sixteen GLCM matrices, [11] extracted fourteen features to represent intensity variation.

1.3 Multilevel decomposition

This paper provides a multilevel decomposition based on uniform quantization algorithm. All images extracted from the database are 8-bits gray-level images (256 gray-level) and were decomposed to other five different gray-level image: $2^3, 2^4, 2^5, 2^6, 2^7, 2^8$. Thus, GLCM matrix calculation for these six quantization levels images could be calculated varying the value of G set.

1.4 Diversity feature extractions

Feature extracted from GLCM matrix are used as texture descriptor for gray-level images based on algebraic and statistical measures.

From biological concepts it is possible to abstract a plenty of mathematical measures to classify mammography images. Diversity indexes provides relative information about specie variety in a certain ecosystem. Theses biological entropies have been used to compose these sets of feature vectors [8].

1.4.1 Tsallis entropy

As a generalization of Boltzmann-Gibbs statistics used for thermodynamic, Tsallis entropy provides a diversity index that can be used in many applications [12]. Equation (2) was adapted for our experiment with GLCM matrix and is expressed as follows:

$$S = \frac{1 - \sum_{i=0}^{G} \sum_{j=0}^{G} (p_{ij})^q}{q - 1} \tag{2}$$

where p_{ij} represents a normalized relative probability of one pixel intensity i and another intensity j are in a given direction θ and distance d from GLCM matrix.

From empirical experiments we selected $q = 0.3$ which provided the best values of accuracy rate.

1.5 Classification

The classification process consists on selection of instances of two or more classes of objects. In this paper, the objects are ROIs from mammogram images which are coded using a set of feature vectors based on Tsallis entropy. Such feature vectors contain some of the descriptors of specified patterns to be identified.

An auxiliary library called libSVM [13] was used in this work to provide pattern classification based on SVM concepts. The Kernel function chosen in this stage was the radial basis function (RBF) given by:

$$K(x, y) = e^{-\gamma ||x-y||^2} \tag{3}$$

where $\gamma > 0$ is a user-defined parameter. Such parameter normalize units in the gaps of SVM feature spaces.

These parameters were selected based on a script provided by libSVM library [13]. The *grid.py* script was executed to find the best values of γ for each experiment.

2. Experiments and results

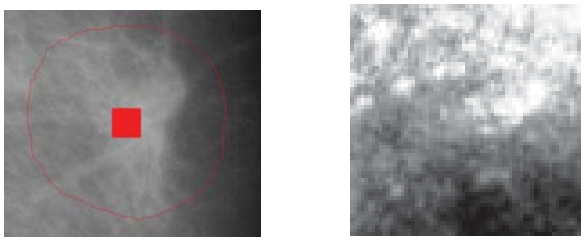
The main work proposal consists on GLCM features extraction for classification of breast tissue in mass and non-mass category. Digital Database Screening Mammography (DDSM) [6], [7] was used as main image database source.

This database have been supported by the Breast Cancer Research Program of the U.S. Army Medical Research and Material Command, the Massachusetts General Hospital, the University of South Florida and Sandia National Laboratories and contains 2620 cases, separated by normal, benign and cancer cases.

For this paper 297 mass images and 297 non-mass images were randomly selected for all the experiments. DDSM cases are separated from anomalies diagnosis and each one contains at least five different files: four mammography images (left and right CC view and MLO view) and one text file composed by exams information (scanning date, film type and scanning density). For mass cases (malignant and benign) another text file is required to specify informations about anomalies presents on the images such as pathology, number of anomalies, anomalies shapes and borders and the chain code.

Chain code information provides a set of numbers about anomaly contour and localization to help on ROI extraction in pre-processing step. From the first two numbers in chain code sequence, the first (x, y) pixel coordinates are obtained and used as start pixel for our extraction step algorithm. After reading all numbers in the sequence, all pixels inside this contour are compared to select the four pixels coordinates (maximals and minimals points) and then calculate the center pixel coordinate.

Therefore, we can expand bounding box limits from center pixel. This technique provide us to select the most anomalies pixels in chain code ROI. After this step, ROI image extracted was equalized to enhanced gray-level tonalities (Figure 4). For empirical experiments, we select the ROI size 64×64, which provided the best results for accuracy rate.



(a) ROI extraction (b) ROI equalization

Fig. 4: Mammography ROIs

2.1 First experiment: Fixing a direction

This experiment evaluate accuracy rate from only Tsallis entropy. Calculation of GLCM matrix from multilevel decomposition of each equalized image on randomly selected database.

For each gray-level G we calculate a GLCM matrix for a fixing direction θ and varying d, where $d \in \{1, 2, 3, 4\}$. Then, Tsallis entropy was calculated from each one of these GLCM matrix. Thus, 24 features were extracted from this experiment.

Table 1 presents the results of accuracy rate based on Tsallis entropy from six GLCM matrix with a fixed direction. The best result was provided by direction $\theta = 0°$ with a accuracy rate of 84.21%.

Table 1: Tsallis entropy with a fixed direction θ

Angle	Train/Test	Acc.	Sens.	Spec.	VPP	VPP
0°	100/494	84.21%	78.54%	89.88%	88.58%	80.73%
45°	100/494	78.54%	72.87%	84.21%	82.19%	75.64%
90°	100/494	77.94%	72.06%	83.81%	81.65%	75.00%
135°	100/494	79.35%	75.71%	83.00 %	81.66%	77.36%

2.2 Second experiment: Fixing a distance

Following the same concepts presented on the above experiment, the main propose of this second experiment was evaluate accuracy rate by fixing the distance d of GLCM calculation and varying the direction θ, where $\theta \in \{0°, 45°, 90°, 135°\}$

This experiment provided the results of only Tsallis entropy. Table 2 presents this results with the best accuracy rate of 81.58% provided by the fixed $d = 2$.

Table 2: Tsallis entropy with a fixed distande d

Distance	Train/Test	Acc.	Sens.	Spec.	VPP	VPP
1	100/494	80.77%	73.68%	87.85%	85.85%	76.95%
2	100/494	81.58%	73.28%	89.88%	87.86%	77.08%
3	100/494	68.42%	68.02%	68.83%	68.57%	68.27%
4	100/494	77.13%	67.61%	86.64 %	83.50%	72.79%

3. Conclusions

In this paper, we developed two experiments based on GLCM and Tsallis diversity entropy concepts. From the first experiment, calculating GLCM using fixed directions θ, we can observe that the best result was 84.21% with a $\theta = 0$. Based on the same analysis, the second experiment using the calculation of GLCM using fixed distances provided the best results was 81.58% from a fixed distance $d = 2$.

The main proposed of this paper was evaluate the influence of Tsallis entropy on classification of mammogram images. As the results showed, this descriptor can be considered a powerful feature for this problem, based on the fact that we used only a set of 24 features. As future work, the objective consists on the analysis of combining possibilities of this descriptor to increase the accuracy rate.

References

[1] J. e. a. Ferlay. (2013) Cancer incidence and mortality worldwide.

[2] P. Economopoulou, G. Dimitriadis, and A. Psyrri, "Beyond brca: New hereditary breast cancer susceptibility genes," *Cancer Treatment Reviews*, vol. 41, pp. 1 – 8, 2015.

[3] K. D. Choundhry, N. and N. Sharma, "Breast cancer: A paradigm shift," *Apollo Medicine*, vol. 9, pp. 133–137, 2012.

[4] A. Howell, A. Anderson, R. Clarke, S. Duffy, D. Evans, M. Garcia-Closas, A. Gescher, T. Key, J. Saxton, and M. Harvie, "Risk determination and prevention of breast cancer," *Breast Cancer Research*, vol. 16, 2014.

[5] J. S. Drukteinis, B. P. Mooney, C. I. Flowers, and R. A. Gatenby, "Beyond mammography: New frontiers in breast cancer screening," *The American Journal of Medicine*, vol. 126, no. 6, pp. 472 – 479, 2013.

[6] B. K. K. D. Heath, M., "The digital database for screening mammography." Medical Physics Publishing, 2001, pp. 212–218.

[7] D. K. Michael Heath, Kevin Bowyer, "Current status of the digital database for screening mammography," *Digital Mammography*, 1998.

[8] G. B. Junior, S. V. da Rocha, M. Gattass, A. C. Silva, and A. C. de Paiva, "A mass classification using spatial diversity approaches in mammography images for false positive reduction," *Expert Systems with Applications*, vol. 40, no. 18, pp. 7534 – 7543, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417413005137

[9] Mohanalin, P. K. Kalra, and N. Kumar, "An automatic method to enhance microcalcifications using normalized tsallis entropy," *Signal Processing*, vol. 90, no. 3, pp. 952 – 958, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165168409004034

[10] Mohanalin, Beenamol, P. K. Kalra, and N. Kumar, "A novel automatic microcalcification detection technique using tsallis entropy a type ii fuzzy index," *Computers and Mathematics with Applications*, vol. 60, no. 8, pp. 2426 – 2432, 2010.

[11] R. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. SMC-3, Nov 1973.

[12] P. Rodrigues and G. Giraldi, "Computing the q-index for tsallis nonextensive image segmentation," in *Computer Graphics and Image Processing (SIBGRAPI), 2009 XXII Brazilian Symposium on*, Oct 2009, pp. 232–237.

[13] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

SESSION

LATE BREAKING PAPERS: BIOINFORMATICS AND COMPUTATIONAL BIOLOGY

Chair(s)

TBA

A NOVEL ALGORITHM FOR FINDING LONGTERM INFLUENCE AND SENSITIVITY OF GENES IN PROBABILISTIC GENETIC REGULATORY NETWORKS

QUOC-NAM TRAN*
THE UNIVERSITY OF SOUTH DAKOTA, USA

ABSTRACT. Investigation of the gene's long-term behavior such as the attractors of the system, which were hypothesized to characterize cellular phenotype, becomes more important in understanding the interactions between different genes and how the genes collectively behave. Unfortunately, current computational methods for analysis the long-term behavior of genes in a Probabilistic Boolean Genetic Regulatory Networks (PBN), require the construction of the network's transition probability matrix, which has a huge size of $2^n \times 2^n$ where n is the number of genes on the PBN. That said, even a PBN with a small set of 20 genes would require 8,796 GB of memory surpassing the size of any computer system.

We present an algebraic method for direct computation of the long-term influence, sensitivity and impact factor of genes in a PBN. Our novel method only requires $O(n^2)$ memory space - a significant improvement in term of space as well as time complexity. We are able to analyze the long-term behavior of genes in PBNs with 500 genes within 15 minutes on a desktop computer. We compare our novel method with previous known methods and report experimental results to illustrate our theoretical arguments. Our method enable the calculation of likelihood of the occurrence of certain events of interest. Thus allow quantitative statements to be made about the system's behavior, expressed as probabilities or expectations of biological systems.

KEYWORDS: Genetic regulatory networks, probabilistic Boolean networks, longterm influence of genes, sensitivity of genes, impact factor of genes.

1. INTRODUCTION

Boolean networks are well-studied discrete models of biological networks such as gene regulatory networks where DNA segments in a cell interact with each other indirectly through their RNA and protein expression products or with other substances in the cell, thereby governing the rates at which genes in the network are transcribed into mRNA. A Boolean network consists of a set of Boolean variables whose state is determined by other variables in the network. They are a particular case of discrete dynamical networks, where time and states are discrete. A Boolean network can be considered as a directed graph where the nodes represent the expression status of genes and directed edges represent the actions of genes on other genes. Each node $x_i \in \{0,1\}$, $i = 1 \dots n$, is a Boolean variable whose state value at time $t+1$ is completely determined by the state values of nodes $x_{j_1}, x_{j_2} \dots x_{j_l}$ for some $1 \leq l \leq n$ at time t by means of a Boolean function $f_i : \{0,1\}^l \to \{0,1\}$ when there are edges from x_{j_k} to x_i for all $k = 1 \dots l$. Thus, one can write $x_i(t+1) = f_i(x_{j_1}(t), x_{j_2}(t) \dots x_{j_l}(t))$, $i = 1 \dots n$.

Probabilistic Boolean Genetic Regulatory Networks (PBN) are probabilistic or stochastic generalizations of Boolean networks. In these models, the deterministic dynamics are replaced by probabilistic dynamics, which can be framed within the mature and well-established theory of Markov chains, for which many analytical and numerical tools have been developed. The value of node x_i at time $t+1$ is now specified by possibly different Boolean functions and state transition probabilities

$$x_i(t+1) = \begin{cases} f_{i_1}(x_{j_1}(t), x_{j_2}(t) \dots x_{j_l}(t)) & \text{with probability } p_i^1 \\ \cdots & \\ f_{i_m}(x_{j_1}(t), x_{j_2}(t) \dots x_{j_l}(t)) & \text{with probability } p_i^m \end{cases} \tag{1.1}$$

*Supported in part by NSF through award CCF-1450146.

where $p_i^k \in [0,1]$, $1 \leq k \leq m$, and $\sum_{k=1}^{m} p_i^k = 1$.

This computational tool has been used in system biology to study biological systems from a holistic perspective to provide a comprehensive, system level understanding of cellular behavior. PBN modeling can be used for the design and analysis of intervention strategies for moving the networks out of undesirable states such as those associated with diseases into the more desirable ones. PBNs have also been used to study the analysis and control of biological networks in order to find a method for a suitable medication which can be used for drug discovery and cancer treatment [3, 5, 9, 11, 12, 16, 17, 19, 22, 23, 29].

Boolean networks are special cases of PBNs in which state transition probabilities are either 1 or 0. The probabilistic nature of this PBN model affords flexibility and power in terms of making inferences from data, which necessarily contain uncertainty, as well as in terms of understanding the dynamical behavior of biological networks, particularly in relation to their structure. PBN is a discrete-time Markov chain in that the behavior at each point in time can be described by a discrete probabilistic choice over several possible outcomes.

Unfortunately, modeling of gene regulatory networks often leads to dynamic models with huge state space surpassing the size of any computer systems by orders of magnitude. One of the key aspects in the analysis of PBN is the investigation of their long-term behavior such as the attractors of the system, which were hypothesized to characterize cellular phenotype [6, 14, 15].

Markov chain Monte Carlo (MCMC) has been proposed for analyzing long-term behavior distribution by running the Markov chain for a sufficient long time until convergence into the stationary distribution and observing the proportion of time the process spent in the parts of the state space that represent the information of interest such as the joint stationary distribution of several specific genes [3, 5, 17, 22]. Due to the difficulties with the assessment of the convergence rate to the longterm distribution, approximation such as the two-state Markov chain can be used to empirically determine when to stop the simulation and output estimates. Unfortunately, the actual inference step of the two-state Markov chain is challenging and to our knowledge the method has not been widely applied for the analysis of large PBNs.

One of the biggest obstacles for the existing methods to analyze the long-term behavior genes of a PBN is the requirement to compute the state transition diagram, which has 2^n nodes for a given PBN with n states. In this paper, we propose a new method in which the state transition diagram is not needed. We utilize algebraic computation for the direct computation of the long-term influence and sensitivity of genes in a PBN. Our novel method only requires $O(n^2)$ memory space - a significant improvement in term of space as well as time complexity. We are able to analyze the long-term behavior of genes in PBNs with 500 genes within 15 minutes on a desktop computer. We then compare our novel method with previous known methods and report experimental results to illustrate our theoretical arguments. Our method enable the calculation of likelihood of the occurrence of certain events of interest. Thus allow quantitative statements to be made about the system's behavior, expressed as probabilities or expectations of biological systems.

2. Influence and Sensitivity Factors of Genes in PBNs

In a PBN, some genes may be more sensitive or have more impact than others in determining the value of a target gene. Finding the genes those has the most potent effect is an important task in studying the PBN. For example, if gene x_1 has the following predictors

$$x_1(t+1) = \begin{cases} f_{1_1}(x_1(t), x_2(t), x_3(t)) = x_2 & \text{prob. } 0.7 \\ f_{1_2}(x_1(t), x_2(t), x_3(t)) = x_2 + x_1 \cdot x_3 & \text{prob. } 0.3 \end{cases} \tag{2.1}$$

then x_2 is a more important variable in influencing gene x_1. Similarly, some genes may be more stable with other genes have little effect on it.

There are many examples of such biased regulation of gene from biologists. The cell cycle regulator gene p21, which is a potent cyclin-dependent kinase inhibitor can be transcriptionally activated by a series of genes p53, smad4, AP2, BRCA1, etc. Among those genes, p53 has the most potent effect [8].

In this paper, we assume that a PBN has been built from experimental data. Many research on building a PBN can be found from recent works on building methods such as the coefficient of determination [10, 21, 27, 28]. We will concentrate on analyzing the long-term influence and sensitivity of genes in a given PBN and will present a computational method for direct computation of the longterm influence and sensitivity of genes in a PBN.

Following the work of [23], we use the notion of partial derivatives of Boolean functions in defining the influence of a gene. It is noticed that the expressiveness of Boolean algebras is significantly extended by the Boolean differential calculus [1, 24]. The additionally defined differentials of Boolean variables, differentials and further differential operators of Boolean functions as well as several derivative operations of Boolean functions allow to model changes of function values together with changes of the values of variables and many other properties of Boolean functions.

The partial derivatives of Boolean function $f(x)$ with respect to a variable x_j is defined as

$$\frac{\partial f(x)}{\partial x_j} = f(x^{(j,0)}) \oplus f(x^{(j,1)})$$

where $\oplus$ is the *xor* operator and $x^{(j,k)} = (x_1, \ldots, x_{j-1}, k, x_{k+1}, \ldots, x_n)$ for $k = 0, 1$. The partial derivative of a Boolean function with respect to the j-th variable indicates whether or not the function differs along the j-th dimension. The partial derivative is 0 if toggling the value of variable x_j does not change the value of the function and it is 1 otherwise.

The influence of variable x_j on the function $f(x)$ is further defined as the expectation of the partial derivative with respect to the distribution $D(x)$:

$$I_j(f) = E_D\left[\frac{\partial f(x)}{\partial x_j}\right] = Pr\left[\frac{\partial f(x)}{\partial x_j} = 1\right] = Pr\{f(x) \neq f(x^{(j)})\}$$

where $Pr\{f(x) \neq f(x^{(j)})\}$ is the probability that a toggle of the j-th variable changes the value of the function $f(x)$ [13].

2.1. **Influence Factor of Genes.** Let $F_i = \{f_1^{(i)}, \ldots f_{l(i)}^{(i)}\}$ be the set of predictors for gene x_i. The influence of variable x_j on variable x_i is defined as

$$I_j(x_i) = \sum_{k=1}^{l(i)} I_j(f_k^{(i)}) \cdot p_k^{(i)}.$$

We now construct a matrix A of influences as $A_{i,j} = I_j(x_i)$. A graph of influence can be constructed where vertices are genes. There is an edge from node j to node i if gene j should transfer its influence to gene i. For example, in Figure 2.2 gene 1 has three outgoing edges, so it will transfer its influences to gene 1, 2 and 3. In general, if a node has k outgoing edges, it will pass on its importance to each of the nodes that it links to. Hence, we will normalize each column of matrix A so that $\sum_{i=1}^{n} A_{ij} = 1$, for $j = 1..n$. In other words, we defined a column-stochastic matrix with all of its entries are nonnegative and the entries in each column sum to one.

Conversely, gene 3 has four incoming edges, so it will be influenced by gene 1 with probability 0.50, gene 2 with probability 0.93, gene 3 with probability 0.51 and gene 4 with probability 0.05 at time $t+1$. However, gene 1 was influenced by gene 2 with probability 1.0 at the previous time t. That said besides the direct influence with probability 0.93 from gene 2 to gene 3 at time $t+1$, gene 2 also indirectly influences gene 3 at the previous time through gene 1. In general, if a node has k incoming edges, it will be influenced by each of the nodes that it is linked to. From our calculation, if a gene is influenced at higher rate then it is more sensitive to changes from other genes.

Suppose that initially the *influence factor* or sensitivity is uniformly distributed among the 4 genes, each getting 1⁄4. In fact, we can use any sensitivity value for the genes that we can determine at the initial time. Denote by v the initial influence or sensitivity vector, having all entries equal to 1⁄4. Each incoming link increases the influence factor of the gene, so at time 1, we update the influence factor of each gene by adding to the current value the influence of the incoming links. This is the same as multiplying the matrix A with v. At time 1, the new influence

vector is $v_1 = A \cdot v$. We can iterate the process, thus at time 2, the updated influence vector is $v_2 = A(A \cdot v) = A^2 \cdot v$. We notice that the sequences of iterates v, $A \cdot v$, ..., $A^k \cdot v$ tends to the equilibrium value. We call this the influence vector of our PBN. Clearly, our method requires only $O(n^2)$ space for matrix A and there is no need to build the transition graph of size $O(2^n)$. Notice that Step 3 in Algorithm 1 can be replaced by $A \leftarrow A^2$ as an alternative way to calculate the influence vector of genes.

Input: a PBN
Output: influence vector of genes in the PBN
Step 1: Construct a matrix of influences A, and an initial influence or sensitivity vector, having all entries equal to $1/n$.
Step 2: Normalize A to make it a column-stochastic matrix
Step 3: While $\|A \cdot v - v\| > \varepsilon$ do $v \leftarrow A \cdot v$.
Step 4: Return v

Algorithm 1: geneInfluence

Lemma 1. *Algorithm geneInfluence always terminates and gives an influence vector of genes in the PBN.*

Proof. Since A is a column-stochastic square matrix, A and its transpose A^T share the same characteristic polynomial and hence have the same eigenvalues. It is easy to see that $A^T \cdot e = e$, so that 1 is an eigenvalue for A^T and hence for A where e denote an n dimensional column vector with all entries equal to 1. □

2.2. **Impact Factor of Genes.** In the same manner, we can obtain a matrix for how a gene impacts other predictors by defining $B = A^T$. In other words, $B_{i,j} = A_{j,i}$. For example, in Figure 2.2 gene 1 has three outgoing edges, so it will pass on its influences to gene 1, 2 and 3. Hence, in corresponding graph for matrix B, which we call the Impact Graph of the PBN, the outgoing edges were reversed to become incoming edges for gene 1. In general, if a node has k incoming edges, it impacted the nodes that it is linked from.

Suppose that initially the importance or *impact factor* is uniformly distributed among the 4 nodes, each getting 1⁄4. In fact, we can use any impact value for the genes at the initial time. Denote by w the initial impact vector, having all entries equal to 1⁄4. Each incoming link increases the impact factor of the gene, so at time -1, we update the impact factor of each gene by adding to the current value the impact of the incoming links. This is the same as multiplying the matrix B with w. At time -1, the new influence vector is $w_1 = B \cdot w$. We can iterate the process, thus at time -2, the updated influence vector is $w_2 = B(B \cdot w) = B^2 \cdot w$. We notice that the sequences of iterates w, $B \cdot w$, ..., $B^k \cdot w$ tends to the equilibrium value. We call this the impact vector of our PBN.

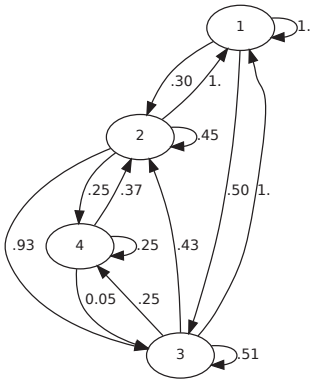


Figure 2.3: Influence Graph

2.3. **Boolean Algebra.** In this section, we will lay down an algebraic framework for the calculation of the partial derivatives of Boolean function. In practice, for a PBN of n genes the predictor functions has only k variables, where k is the in-degree of the networks. In any case, the following algebraic approach will replace the need for building the truth tables for the calculation of the partial derivatives.

Boolean algebras, which were introduced by Boole in the 1850's to codify the laws of thought, have become a popular topic of research since then. The discovery in 1930's of the duality between Boolean algebras and Boolean spaces by Stone [25, 26, 4] was a major breakthrough of the field. Stone also proved that Boolean algebras and Boolean rings are the same in the sense that one can convert from one algebraic structure to the other. In spite of its long history and elegant algebraic properties, the Boolean ring representation has rarely been used in the computational context.

Definition 2. A ring $\mathbf{K} = \langle K, +, \cdot, 0, 1 \rangle$ is Boolean if $\mathbf{K}$ satisfies $x^2 \approx x, \forall x \in K$.

Lemma 3. *If* **K** *is a Boolean ring, then* **K** *is commutative and* $x+x \approx 0$ [4].

Every Boolean algebra $(K, \wedge, \vee)$ gives rise to a ring $(K, +, \cdot)$ by defining $a+b = (a \wedge \neg b) \vee (b \wedge \neg a)$ (this operation is called XOR in the case of logic) and $a \cdot b = a \wedge b$. The zero element of this ring coincides with the 0 of the Boolean algebra; the multiplicative identity element of the ring is the 1 of the Boolean algebra. Conversely, if a Boolean ring **K** is given, we can turn it into a Boolean algebra by defining $x \vee y = x+y+x \cdot y$ and $x \wedge y = x \cdot y$. Since these two sets of operations are inverses of each other, we can say that every Boolean ring arises from a Boolean algebra, and vice versa. Furthermore, a map $f : A \to B$ is a homomorphism of Boolean algebras if and only if it is a homomorphism of Boolean rings. The categories of Boolean rings and Boolean algebras are equivalent. By using these translations, there exists a Boolean polynomial for each Boolean formula and vice versa.

Since congruences on rings are associated with ideals, it follows that the same must hold for Boolean algebras. An ideal of the Boolean algebra **K** is a subset I such that $\forall x, y \in I$ we have $x \vee y \in I$ and $\forall a \in K$ we have $a \wedge x \in I$. This notion of ideal coincides with the notion of ring ideal in the Boolean ring **K**. An ideal I of R is called prime if $I \neq K$ and if $a \wedge b \in I$ always implies $a \in I$ or $b \in I$. An ideal I of K is called maximal if $I \neq K$ and if the only ideal properly containing I is K itself. These notions coincide with ring theoretic ones of prime ideal and maximal ideal in the Boolean ring **K**.

1	1	1	0
0.30	0.45	0.43	0.37
0.50	0.93	0.51	0.05
0	0.25	0.25	0.25

Table 2.3 : Influence Matrix

Despite its extremely simplicity, the Boolean ring representation has not been used extensively both in logical reasoning and in computation. The main reason, which has been shared by other researchers, is that the XOR operator used in Boolean rings is nilpotent and hence negation does not appear in the normal forms. This makes Boolean ring formulas hard to read for human because one cannot tell which predicate symbol is negated and which one is not. Especially, when a formula is long, it is almost impossible to make a natural interpretation of its meaning. For the calculation of the partial derivatives of Boolean functions, we can simply convert our problem into finding the solutions for a Boolean polynomial.

We now provide an example PBN and show how to build an influence graph and how to calculate the influence factor as well as the impact factor of the PBN.

Example 4. Given a PBN consisting of four genes $V = \{v_1, v_2, v_3, v_4\}$ and a set of predictors

$$v_1(t+1) = v_1 + v_2 + v_3 \qquad \text{prob. } 1.00$$

$$v_2(t+1) = \begin{cases} v_1 \cdot v_2 \cdot v_3 + v_1, & \text{prob. } 0.246 \\ v_2 \cdot v_3 \cdot v_4 + v_3 + v_4 + 1, & \text{prob. } 0.378 \\ v_1 \cdot v_2 \cdot v_4 + v_1 \cdot v_3 \cdot v_4 + v_2 \cdot v_3 \cdot v_4 + v_1 \cdot v_2 + v_2 \cdot v_3 + v_2 \cdot v_4 + v_3 \cdot v_4, & \text{prob. } 0.361 \\ v_1 \cdot v_2 \cdot v_4 + v_2 \cdot v_4 + v_2 & \text{prob. } 0.014 \end{cases}$$

$$v_3(t+1) = \begin{cases} v_1 \cdot v_3 + v_2 + 1, & \text{prob. } 0.932 \\ v_1 \cdot v_3 \cdot v_4 + v_3 \cdot v_4 + v_1 + v_3 + v_4 & \text{prob. } 0.068 \end{cases}$$

$$v_4(t+1) = v_2 \cdot v_3 \cdot v_4 + v_2 \cdot v_3 \qquad \text{prob. } 1.00$$

Notice that the probabilities have been rounded off. The actual numbers are [[1.], [.2461853940, .3784104050, .3609895426, 0.1441465845e-1], [.9319607618, 0.6803923820e-1], [1.]] for v_1, v_2, v_3 and v_4 , respectively. The influence matrix of this PBN before normalized is showed in Table 2.3 and the influence graph of the PBN is showed in Figure 2.2.

Suppose that initially the *influence factor* or sensitivity is uniformly distributed among the 4 genes, each getting 1⁄4. The longterm sensitivity or influence vector converged to [0.450595824829880, 0.207132994504042, 0.264470757717693, 0.0778004230596858]. That said gene 4 is the least sensitive gene or in other word the most stable gene in the long run. Gene 1 is the most sensitive gene in the PBN.

Similarly, The longterm impact factor or impact vector converged to [0.210910910659655, 0.357142856843150, 0.292460317400817, 0.139485914740911]. That said gene 2 is the most impacting gene in the PBN.

Notice that the most stable gene of the PBN is not necessarily the most impacting gene of the networks. We have many examples showing that they are in fact not correlated.

3. A Biological Case Study

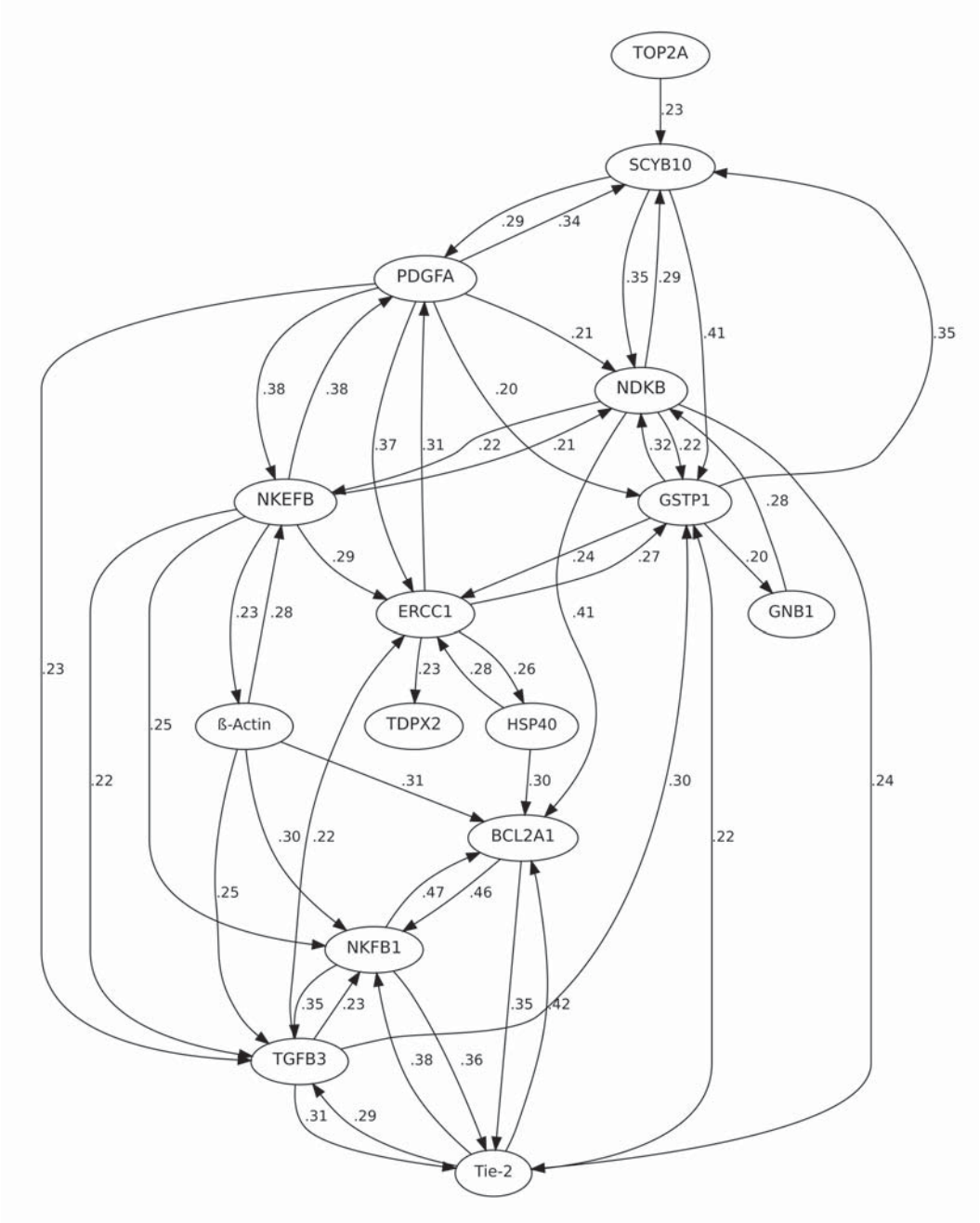


Figure 3.1. Influence Graph for Gliomas

To illustrate the efficiency and the accuracy of our novel algorithm, we use the data from a human glioma gene expression data set [7]. Gliomas are the most common form of primary malignancies of the central nervous system (CNS) mainly affecting adults. These tumors have a histological resemblance to different types of glial

0	0.29	0	0	0	0	0	0.17	0	0	0	0	0	0.31	0.43
0.22	0	0	0	0	0	0	0	0	0	0.13	0.14	0.22	0.30	0
0	0.21	0	0.48	0	0.22	0	0	0	0	0.21	0.18	0	0	0
0	0	0.24	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0.21	0	0	0	0	0	0	0	0	0	0	0	0
0.17	0.28	0.25	0	0	0	0	0.16	0	0.39	0.12	0	0	0	0
0	0	0	0	0	0.18	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0.29	1.00	0	0	0.33	0.12	0.13	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0.32	0	0.21	1.00	0	0.20	0	0	0	0
0	0	0.29	0	0	0	0	0	0	0.28	0	0.24	0	0	0
0	0	0	0	0	0	0	0.16	0	0	0.22	0	0.25	0	0
0	0	0	0	0	0	0	0	0	0	0	0.15	0	0	0
0.29	0.22	0	0	0	0	0	0	0	0	0	0.16	0.26	0	0.57
0.32	0	0	0.52	0	0	0	0.30	0	0	0	0	0.27	0.40	0

TABLE 1. Influence Matrix for Gliomas

cells and are categorized into astrocytomas, oligodendrogliomas, oligoastrocytomas, and ependymomas, based on the predominant cell type(s) in the respective tumor.

A PBN network of 597 genes was inferred using the coefficient of determination as in [5, 23]. A small subnet of 15 genes is shown in Figure 3.1 with the weights on the edges representing the influences of the genes. The influence matrix of this PBN after normalized is showed in Table 1.

With the complexity of predictor functions, some gene influences are pretty small. To deal with many small influences we use the same idea of damping factor in Google's PageRank [2] where the influence matrix is replace by $(1-p)\cdot A + p\cdot T$ where T is the $n\times n$ "teleportation" matrix, i.e, the matrix each of whose entries is $1/n$. We use $p = 0.10$ in our calculation.

Suppose that initially the *influence factor* or sensitivity is uniformly distributed among the genes. The longterm sensitivity or influence vector converged and after normalizing we have [0.987752376216898, 0.613897953484682, 0.509838752830795, 0.163205931648281, 0.150098397974440, 0.800165646879760, 0.181809171192399, 0.656777507172725, 0.0496073065684904, 0.519537291701068, 0.369183660877093, 0.238311660580107, 0.0814175288392782, 1., 0.981858486296723] for genes Tie-2, TGFB3, ERCC1, HSP40, TDPX2, GSTP1, GNB1, NDKB, TOP2A, SCYB10, PDGFA, NKEFB, β-Actin, NKFB1 and BCL2A1, respectively.

Some obvious expectation such as gene TOP2A should be stable and should not be sensitive to the influence of other genes can be verified from its lowest influence factor. Our results also present some knowledge that is not known before.

3.1. **Stable genes.** From the converged influence vector it shows that genes HSP40, TDPX2, GNB1, NKEFB and β-Actin are stable or not be sensitive to the influence of other genes in a long-run. Table 2 shows the similar pattern of these genes when the influence vector converges to its fixed-point.

3.2. **Sensitive genes.** On the other hand, we found that genes Tie-2, GSTP1, NFKB1 and BCL2A1 are very sensitive to the influence of other genes in a long-run. This finding is in-line with what we have learned from biologists as [18] found that Tie2 activation was related to the up-regulation of integrin beta1 levels and the formation of focal adhesions. These results, together with the reported fact that malignant gliomas express high levels of Ang1, suggest the existence of an autocrine loop in malignant gliomas and that a Tie2-dependent pathway modulates cell-to-extracellular matrix adhesion, providing new insights into the highly infiltrative phenotype of human gliomas. The abnormal function of tyrosine kinase receptors is a hallmark of malignant gliomas. Tie2

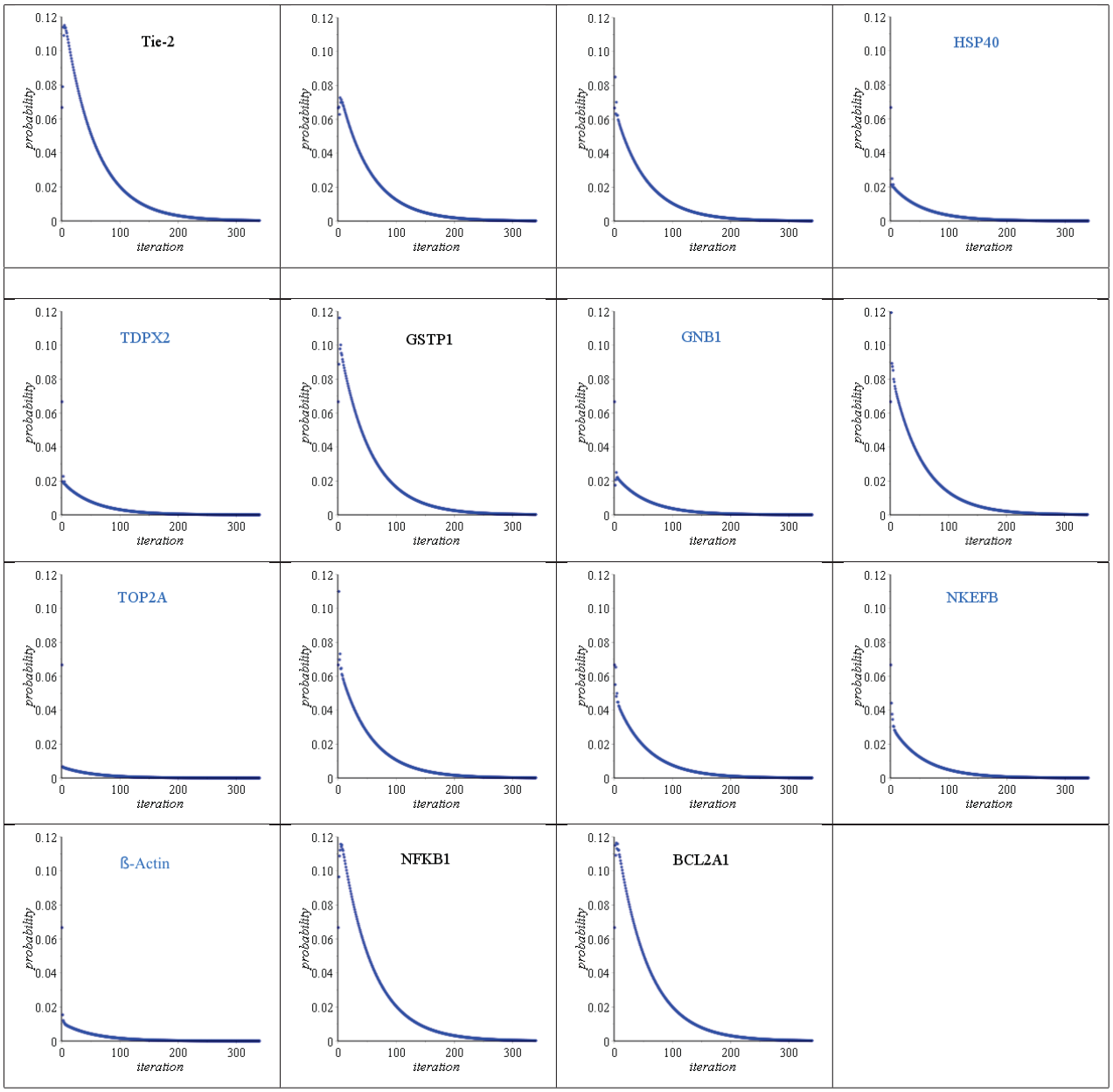


TABLE 2. Convergence of influence factors

receptor tyrosine kinase is a specific endothelial cell receptor whose function is positively regulated by angiopoietin 1 (Ang1). Recently, Tie2 has also been found in the nonvascular compartment of several tumors, including leukemia as well as breast, gastric, and thyroid cancers. Furthermore, analysis of a tissue array consisting of 116 human glioma samples showed that Tie2 expression in the neoplastic glial cells was significantly associated with progression from a lower to higher grade. Importantly, Ang1 stimulation of Tie2+ glioma cells resulted in increased adherence of the cells to collagen I and IV, suggesting that Tie2 regulates glioma cell adhesion to the extracellular matrix. Conversely, the down-regulation of Tie2 levels by small interference RNA or the addition of soluble Tie2 abrogated the Ang1-mediated effect on cell adhesion.

3.3. **Hi-impact genes.** Similarly, suppose that initially the *impact factor* is uniformly distributed among the genes. The longterm impact vector converged and after normalizing we have [0.230510562021945, 0.2817332740-14609, 0.554286767759576, 0.162393143989202, 0.0389022223245352, 0.496877449208407, 0.135282472604-153, 0.513376834870215, 0.128454945353256, 0.512887379689246, 0.875556528547034, 0.999999999947150, 0.436019685843127, 0.200929165773528, 0.150127497384119] for genes Tie-2, TGFB3, ERCC1, HSP40, TDP-X2, GSTP1, GNB1, NDKB, TOP2A, SCYB10, PDGFA, NKEFB, β-Actin, NKFB1 and BCL2A1, respectively.

From the converged impact vector it shows that genes PDGFA and NKEFB are the highest impacting genes in a long-run. This finding is in-line with what we have learned from biologists as [20] showed the family of platelet-derived growth factors (PDGFs) plays a number of critical roles in normal embryonic development, cellular differentiation, and response to tissue damage. Not surprisingly, as it is a multi-faceted regulatory system, numerous pathological conditions are associated with aberrant activity of the PDGFs and their receptors. As it has been shown, human gliomas, especially glioblastoma, express all PDGF ligands and both the two cell surface receptors, PDGFR-α and -β. The cellular distribution of these proteins in tumors indicates that glial tumor cells are stimulated via PDGF/PDGFR-α autocrine and paracrine loops, while tumor vessels are stimulated via the PDGFR-β.

4. Conclusion

We presented an algebraic method for direct computation of the long-term influence and sensitivity of genes in a PBN. Our novel method only requires $O(n^2)$ memory space in contrast to other known methods in the literature which require the construction of the network's transition probability matrix with a huge size of $2^n \times 2^n$ where n is the number of genes on the PBN. We are able to analyze the long-term behavior of genes in PBNs with 500 genes within 15 minutes on a desktop computer.

Our biological case study, using PBN from a human glioma gene expression data set, showed that the long-term influence and sensitivity of genes we found in this gliomas PBN are in-line with what we have learned from biologists. Furthermore, our results also present some knowledge that is not known before. We hope that this new finding will help to direct more wet bench research on long-term influence and sensitivity of genes.

References

[1] D. Bochmann and C. Posthoff. *Binäre Dynamische Systeme*. R. Oldenbourg Verlag, München, 1981.

[2] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *COMPUTER NETWORKS AND ISDN SYSTEMS*, pages 107–117. Elsevier Science Publishers B. V., 1998.

[3] M. Brun, E. R. Dougherty, and I. Shmulevich. Steady-state probabilities for attractors in probabilistic boolean networks. *Signal Processing*, 85(10):1993–2013, 2005.

[4] S. N. Burris and H. P. Sankappanavar. *A Course in Universal Algebra*. Springer-Verlag, 1981.

[5] E. R. Dougherty and I. Shmulevich. Mappings between probabilistic boolean networks. *Signal Processing*, 83(4):799–809, 2003.

[6] E. Dubrova, M. Teslenko, and A. Martinelli. Kauffman networks: analysis and applications. In *ICCAD '05: Proceedings of the 2005 IEEE/ACM International conference on Computer-aided design*, pages 479–484, Washington, DC, USA, 2005. IEEE.

[7] G. Fuller, C. Rhee, K. Hess, L. Caskey, R. Wang, J. Bruner, W. Yung, and W. Zhang. Reactivation of insulin-like growth factor binding protein 2 expression in glioblastoma multiforme: a revelation by parallel gene expression profiling. *Cancer Res.*, 59(17):4228–4232, 1999.

[8] A. Gartel and A. Tyner. The role of the cyclin-dependent kinase inhibitor p21 in apoptosis. *Mol Cancer Res.*, 1(8):639–649, 2002.

[9] C. Gershenson, S. A. Kauffman, and I. Shmulevich. The role of redundancy in the robustness of random boolean networks. *CoRR*, abs/nlin/0511018, 2005.

[10] M. Grimaldi, R. Visintainer, and G. Jurman. RegnANN: Reverse engineering gene networks using artificial neural networks.

[11] R. F. Hashimoto, S. Kim, I. Shmulevich, W. Z. 0011, M. L. Bittner, and E. R. Dougherty. Growing genetic regulatory networks from seed genes. *Bioinformatics*, 20(8):1241–1247, 2004.

[12] M. Hayashida, T. Tamura, T. Akutsu, S. Zhang, and W.-K. Ching. Algorithms and complexity analyses for control of singleton attractors in boolean networks. *EURASIP J. Bioinformatics and Systems Biology*, 2008, 2008.

[13] J. Kahn, G. Kalai, and N. Linial. The influence of variables on boolean functions. In *FOCS: IEEE Symposium on Foundations of Computer Science (FOCS)*, 1988.

[14] S. A. Kauffman. Metabolic stability and epigenisis in randomly constructed genetic nets. *J. Theoretical Biology*, 22:437–467, 1969.

[15] S. A. Kauffman. *The Origins of Order: Self-Organization and Selection in Evolution.* Oxford University Press, 1993.

[16] H. Lähdesmäki, S. Hautaniemi, I. Shmulevich, and O. Yli-Harja. Relationships between probabilistic boolean networks and dynamic bayesian networks as models of gene regulatory networks. *Signal Processing*, 86(4):814–834, 2006.

[17] H. Lähdesmäki, I. Shmulevich, and O. Yli-Harja. On learning gene regulatory networks under the boolean network model. *Machine Learning*, 52(1-2):147–167, 2003.

[18] O. Lee, J. Xu, J. Fueyo, G. Fuller, K. Aldape, M. Alonso, Y. Piao, T. Liu, F. Lang, B. Bekele, and C. Gomez-Manzano. Expression of the receptor tyrosine kinase tie2 in neoplastic glial cells is associated with integrin beta1-dependent adhesion to the extracellular matrix. *Mol Cancer Res.*, 4(12):915–926, 2006.

[19] W. Liu, H. Lähdesmäki, E. R. Dougherty, and I. Shmulevich. Inference of boolean networks using sensitivity regularization. *EURASIP J. Bioinformatics and Systems Biology*, 2008, 2008.

[20] I. Nazarenko, S.-M. Hede, X. He, A. Hedrén, J. Thompson, M. S. Lindstroem, and M. Nistér. Pdgf and pdgf receptors in glioma. *Ups J Med Sci.*, 117(2):99–112, 2012.

[21] N. Noman and H. Iba. Inferring gene regulatory networks using differential evolution with local search heuristics. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4(4):634–647, Oct. 2007.

[22] I. Shmulevich and E. R. Dougherty. *Probabilistic Boolean Networks - The Modeling and Control of Gene Regulatory Networks*. SIAM, 2010.

[23] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Z. 0011. Probabilistic boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics*, 18(2):261–274, 2002.

[24] B. Steinbach and C. Posthoff. *Boolean Differential Equations*. Synthesis Lectures on Digital Circuits and Systems. Morgan & Claypool Publishers, 2013.

[25] M. Stone. The theory of representation for boolean algebras. *Trans. Amer. Math. Soc.*, 1936.

[26] M. Stone. Applications of the theory of boolean rings to general topology. *Trans. Amer. Math. Soc.*, 1937.

[27] Y. Tamada, H. Bannai, S. Imoto, T. Katayama, M. Kanehisa, and S. Miyano. Utilizing evolutionary information and gene expression data for estimating gene networks with bayesian network models. *J. Bioinformatics and Computational Biology*, 3(6):1295–1314, 2005.

[28] Y. Tamada, S. Imoto, H. Araki, M. Nagasaki, C. G. Print, S. D. Charnock-Jones, and S. Miyano. Estimating genome-wide gene networks using nonparametric bayesian network models on massively parallel computers. *IEEE/ACM Trans. Comput. Biology Bioinform*, 8(3):683–697, 2011.

[29] S. Zhang, M. Hayashida, T. Akutsu, W.-K. Ching, and M. K.-P. Ng. Algorithms for finding small attractors in boolean networks. *EURASIP J. Bioinformatics and Systems Biology*, 2007, 2007.

In Silico Prediction of 3D Structure of *Anopheles Gambiae* CYP6Z3 Protein

Marion Adebiyi[1] and Adaobi Okafor[2]
[1]Department of Computer and Information Sciences and Covenant University Bioinformatics Research (CUBRe), Covenant University, Ota, Nigeria
[2]Department of Biological Sciences, Covenant University, Ota, Nigeria

Abstract— The CYP6Z3 protein found in *Anopheles gambiae*, the disease vector for the malaria parasite, *Plasmodium falciparum*, has been implicated in pyrethroid resistance. It belongs to the Cytochrome P450 family and functions in oxidation-reduction processes. It has the VECTORBASE Annotation AGAP008217. The *Anopheles gambiae* CYP6Z3 protein sequence was obtained from UNIPROT database. BLAST analysis was performed to extract our homologous template protein sequence, 1TQN_A, the human microsomal P450 3a4. Using the comparative modeling program, Swiss Model, the CYP6Z3 sequence was combined with the 3D structure of our template protein, 1TQN_A, to determine our target structure. Both Clustal omega and Clustalw2 were used to generate alignment files as input into the alignment mode and both alignment files generated different models. The structure validation program, PROCHECK was used to evaluate the best of all obtained models from alignment mode and automated mode. It was observed that the template protein structures, 1tqn.2.A extracted from Clustalw2 alignment and 1tqn.1.A extracted from Clustal Omega alignment were similar, having the highest percentage of residues in the most favoured regions on the ramachandran plot than the rest of the models generated via the automated mode. MolProbity was used to generate the summary statistics and SaliLab Model Evaluation Server (ModEval) was used to estimate the quality of our best model. This work therefore emphasizes the importance of quality of structure predicted during homology modeling. Such quality is important in deducing the biochemical functions of CYP6Z3. Its role in physiological processes including hormone and pheromone metabolism as well as insecticide detoxification, especially of pyrethroid, in *Anopheles gambiae* was also deduced from this prediction. Targeting this gene would therefore serve as an effective mechanism for malaria control by inhibiting or suppressing the *gambiae's* detoxicative capacity for insecticides.

Keywords: *Anopheles gambiae*, pyrethroid, resistance, homology.

1. Introduction

The *Anopheles gambiae* CYP6Z3 protein is a protein in the Cytochrome P450 family functioning in oxidation-reduction processes. It has the VECTORBASE Annotation AGAP008217, primary (citable) accession number Q86LT6 and entry name Q86LT6_ANOGA. It has a sequence length of 492AA and is located on Chromosome 3R: 6,971,669-6,973,290. It has a mass of 56,490(Da). [1,2]. It has been implicated in pyrethroid resistance [3].

2. Literature Review

The CYP6Z3 protein is expressed during the mosquito's larval stages [4]. Over 30 species of *Anopheles* transmit malaria [5], hence identification of resistance mechanisms in other species is a focus of much research. *Anopheles funestus* is the second major malaria vector in Sub-Saharan Africa. With the use of quantitative trait loci (QTL), several genes strongly associated with pyrethroid resistance in *A. funestus* have been identified including CYP6Z3 [6]. In a recent study, transcription of genes from the four main families of detoxification genes (cytochrome P450s, glutathione transferases, carboxylesterases and UDP glucornyltransferases) were reported to be generally enriched in the midgut and malpighian tubules of *A. gambiae*. Specifically, the CYP6Z family was found to be highly enriched in the malpighian tubules consistent with its role in detoxification [7]. The malpighian tubules therefore display roles similar to the vertebrate liver, kidney and immune system. Despite the wide distribution of detoxifying enzymes in insects, baseline protection against insecticides resides mainly in an insect's excretory system which corresponds to less than 0.1% of its mass [8].

3. Materials And Methods

Target sequence

The *Anopheles gambiae* CYP6Z3 protein sequence was obtained from UNIPROT database in the fasta format (UniProt entry Q86LT6).

>tr|Q86LT6|Q86LT6_ANOGA Cytochrome P450 OS=Anopheles gambiae GN=CYP6Z3 PE=2 SV=1
MFVYTLALVAAVIFLVLRYIYSHWERHGLPHLKP
EIPYGNIRTVAEKKESFGIAINNLYH
KSSDRLLGIYLFFRPAILIRDPHLAKRIMVNDFQN
FHDRGVYCNEEHDPFSANLFALPGQ
RWKNLRAKLTPTFTSGQLRNMLPTLLDVGNKLID
RMNKVADEKAIVDMRDIASRFVLDTI
ASVFFGFEANCIHNSEDPFLSTLQRLTKSRKFMDN
FRTSGVFICPGLLKLTRITSLPPEL
ISFVMEIITHQIDHREKNQITRKDFVQLLIDLRREA
ENGSEKALSIEQCAANVFLFYIAG
AETSTATISFTLHELSHNPEAMAKLQQEIDEMME
RYNGEITYENIKEMKYLDLCVKETLR
KYPGLPILNRECTIDYKVPDSDVVIRKGTQVIIPL
WSISMNEKYFPDPELHSPERFDEAT
KNYDADAYYPFGAGPRNCIGLRQGVFVSKIGLVL
LLSKYNFQATTPAKVKFAVATVVVTP
EDGFPMRVEHRC

Template Selection
BLAST Program [9] (blastp variant) was used to the search for homologous sequences within Protein Data Bank (PDB) database [10]. The BLAST search returned many sequences that were homologous with the target sequence but only one was selected as the template.

Criteria for Selection
Sequence identity of 30% or more was considered. Generally, the homologous sequences returned had low sequence identity between 19%-33%. E-value less than 0.001 and query cover above 50% were also considered. Potential templates were filtered based on these criteria and the atomic resolution of their experimentally-derived 3D structures as viewed from PDB. Structures that failed to meet one or more of these criteria were excluded and the protein with accession 1TQN_A was selected as the template, with query cover 96%, E-value 2e-68 and sequence identity 31%. It has a medium atomic resolution of 2.05Å and was determined by X-ray crystallography. The 1TQN_A protein is a crystal structure of the human microsomal P450 3a4. It has a sequence length of 486AA. It is also a Cytochrome P450 enzyme functioning in oxidation-reduction processes. The sequence for the template was downloaded directly from the BLAST webpage in the fasta format.

>gi|51247719|pdb|1TQN|A Chain A, Crystal Structure Of Human Microsomal P450 3a4
MALYGTHSHGLFKKLGIPGPTPLPFLGNILSYHKG
FCMFDMECHKKYGKVWGFYDGQQPVLAITDPD
MIKTVLVKECYSV
FTNRRPFGPVGFMKSAISIAEDEEWKRLRSLLSPT
FTSGKLKEMVPIIAQYGDVLVRNLRREAETGKPV
TLKDVFGAYSM
DVITSTSFGVNIDSLNNPQDPFVENTKKLLRFDFL
DPFFLSITVFPFLIPILEVLNICVFPREVTNFLRKSV
KRMKESRL
EDTQKHRVDFLQLMIDSQNSKETESHKALSDLEL
VAQSIIFIFAGYETTSSVLSFIMYELATHPDVQQKL
QEEIDAVLPN
KAPPTYDTVLQMEYLDMVVNETLRLFPIAMRLE
RVCKKDVEINGMFIPKGVVVMIPSYALHRDPKY
WTEPEKFLPERFSK
KNKDNIDPYIYTPFGSGPRNCIGMRFALMNMKLA
LIRVLQNFSFKPCKETQIPLKLSLGGLLQPEKPVV
LKVESRDGTVS
GAHHHH

Sequence Alignment
Clustalw2 [11] and Clustal Omega [12] were used to perform sequence alignments between the target sequence and template sequence using the default settings. The resulting .clustalw2 and .clustalo files were downloaded to serve as an input files for the next step.

Building the Model
The Swiss Model Server [13] was used to build the 3D structure of the target.

4. Results

Using .clustalw Alignment File
The "Alignment Mode" was used where the target-template alignment from Clustalw2 was submitted to the server. The server generated a model built from the template 1tqn.2.A (Table I).

TABLE I. FEATURES OF MODEL BUILT FROM THE TEMPLATE 1tqn.2.A USING .CLUSTALW ALIGNMENT FILE

Template	Seq ID	Oligo-state	Method	Seq similarity	Coverage
1tqn.2.A	30.30 %	Homo-tetramer	X-ray, 2.05Å	0.36	0.96

Using .clustalo Alignment File
First, the "Alignment Mode" was used where the target-template alignment from Clustal Omega was submitted to the server (TABLE II). Next, the "Automated Mode" was used where another entry was submitted into the server with only the target protein as input data, allowing Swiss Model Server to search through databases for its template(s) of choice (TABLE III).

Alignment Mode

TABLE II. FEATURES OF MODEL BUILT FROM THE TEMPLATE 1tqn.1.A USING .CLUSTALO ALIGNMENT FILE

Template	**Seq ID**	**Oligo-state**	**Method**	**Seq similarity**	**Coverage**
1tqn.1.A	30.85 %	Monomer	X-ray, 2.05Å	0.36	0.96

Automated Mode

The server found 50 templates and built 3 models

TABLE III. FEATURES OF ALL 3 MODELS BUILT VIA AUTOMATED MODE

	Template	**Seq ID**	**Oligo-state**	**Method**	**Seq similarity**
Model 1	1tqn.1.A	30.11%	Monomer	X-ray, 2.05Å	0.36
Model 2	4k9w.4.A	30.19%	Monomer	X-ray, 2.4Å	0.36
Model 3	3ua1.1.A	30.19%	Monomer	X-ray, 2.15Å	0.36

From the models generated, it was observed that using Clustalw2 for sequence alignment, the model was predicted to be a homo-tetramer while for Clustal Omega, all models were predicted to be monomers. Also, both alignment modes (ie from .clustalw and .clustalo files) generated structures for 1TQN_A proteins (1tqn.2.A and 1tqn.1.A respectively). It was hypothesized that these two proteins were more closely related based on their coverage of 0.96 although they differed in sequence identity compared with a similarly named model (1tqn.1.A) built via automated mode which had a coverage of 0.95.

Model Evaluation

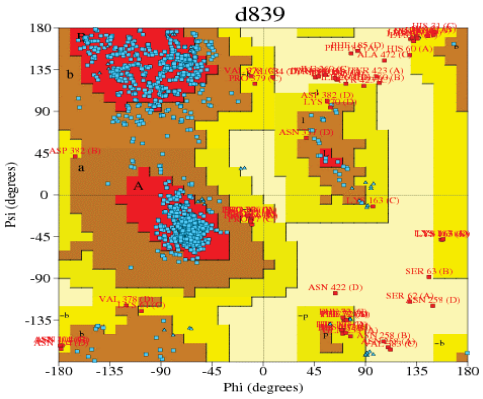
PROCHECK [14,15] was used to do an analysis of the Ramachandran plot for the modelled structures (FIG. 1.).

TABLE IV. RESULT OF RAMACHANDRAN PLOT FOR THE MODELLED STRUCTURES SHOWING MOST FAVOURED AND DISALLOWED REGIONS

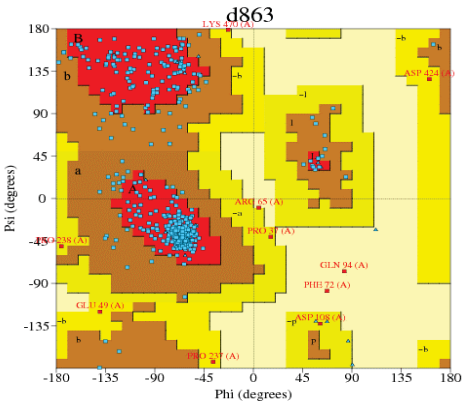
Alignment mode	**PDBSum**	**Most favoured regions**	**Disallowed regions**
1tqn.2.A	d839	87.9%	1.2%
1tqn.1.A	d860	87.7%	0.7%
Automated mode	**PDBSum**	**Most favoured regions**	**Disallowed regions**
1tqn.1.A	d861	80.9%	1.4%
4k9w.4.A	d862	83.2%	1.7%
3ua1.1.A	d863	83.6%	0.7%

From the above tables, as hypothesized, the structures derived from Clustalw2 and Clustal Omega sequence alignments were similar, having the highest percentage of residues in the most favoured regions than the rest of the models generated via the automated mode (TABLE IV). These structures could have represented the same protein but clustalw2 'misintepreted' it to be a homo-tetramer. This shows that the quality of the sequence alignment used as input data into any modelling server, Swiss model in this case, is very important.

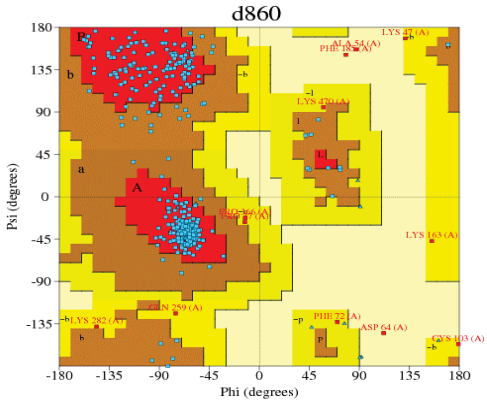
For the automated mode, the 3ua1.1.A model was the best with the highest percentage of residues in the most favoured regions. Comparing this with those obtained via alignment mode, the model 1tqn.1.A generated from clustalo alignment file was the overall best structure having 87.7% residues in the most favourable regions and 0.7% residues in the disallowed regions. All model proteins derived from both alignment and automated modes are shown in Fig. 2.



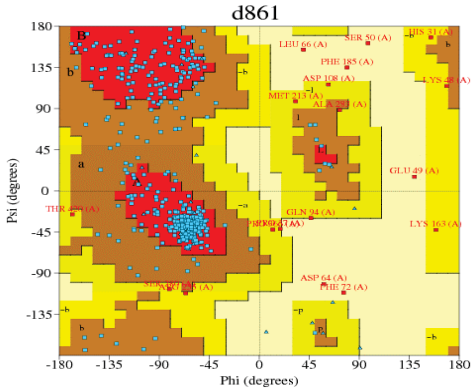
A)



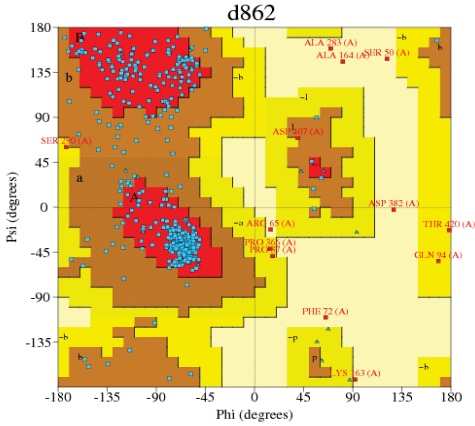
B)



C)



D)

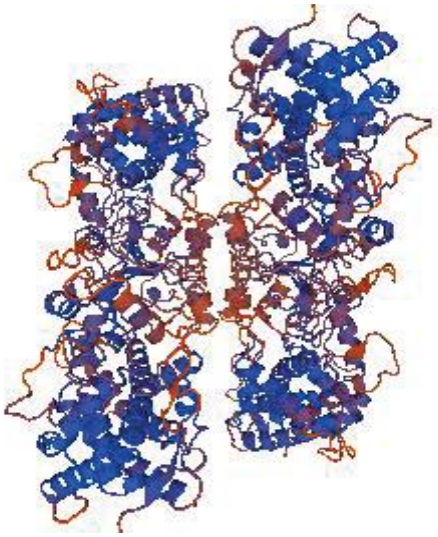


E)

Fig. 1A-E. Ramachandran plots for all the modelled structures.

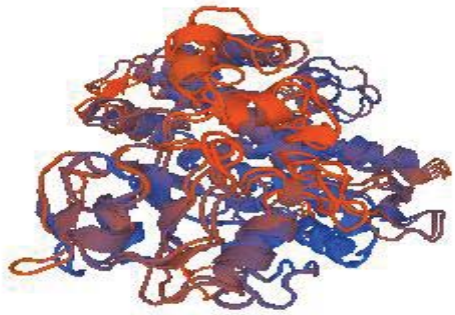
Alignment Mode

1tqn.1.A

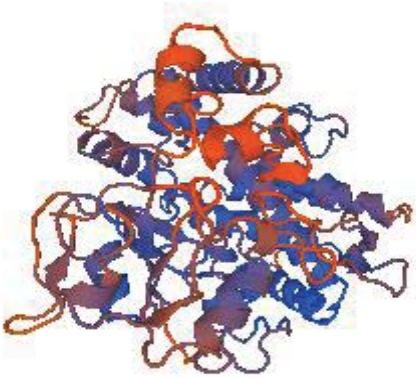


1tqn.2.A

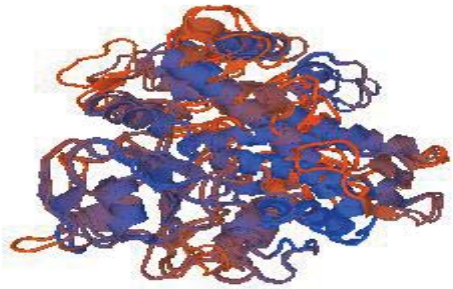
Automated Mode



1tqn.1.A



4k9w.4.A



3ua1.1.A

Fig. 2. Model proteins derived from alignment and automated modes

MolProbity [16] was also used to generate the summary statistics of the best model (1tqn.1.A from alignment mode). Also, the SaliLab Model Evaluation Server (ModEval) was used to estimate the quality of the best model.

5. Conclusion

We obtained our best model by comparing predicted structures generated from .clustalo and .clustalw alignment files. We have been able to show that the quality of structure predicted during homology modeling is very important. The role of CYP6Z3 in physiological processes including hormone and pheromone metabolism as well as insecticide detoxification, especially of pyrethroid, in *Anopheles gambiae* was also deduced. This gene could be targeted to serve as an effective mechanism for malaria control by inhibiting or suppressing the *gambiae's* detoxicative capacity for insecticides.

References

[1] P.J. Kersey, J.E. Allen, M. Christensen, P. Davis, L.J. Falin, C. Grabmueller, et al., "Ensembl Genomes 2013: scaling up access to genome-wide data", Nucl. Acids Res., vol. 42, pp. D546-D552, 2014.

[2] UniProt-Consortium, "Ongoing and future developments at the Universal Protein Resource", Nucl. Acids Res., vol. 39, pp. D214-219, 2011.

[3] P. Müller, M.J. Donnelly, and H. Ranson, "Transcription profiling of a recently colonised pyrethroid resistant *Anopheles gambiae* strain from Ghana", BMC Genomics, vol. 8, pp. 36, 2007.

[4] D. Nikou, H. Ranson, and J. Hemingway, "An adult-specific CYP6 P450 gene is overexpressed in a pyrethroid-resistant strain of the malaria vector, *Anopheles gambiae*", Gene, vol. 318, pp. 91-102, 2003.

[5] Centers for Disease Control and Prevention http://www.cdc.gov/malaria/about/biology/mosquitoes/

[6] H. Irving, J. Riveron, S. Inbrahim, N. Lobo, and C. Wondji, "Positional cloning of rp2 QTL associates the P450 genes CYP6Z1, CYP6Z3 and CYP6M7 with pyrethroid resistance in the malaria vector *Anopheles funestus*", Heredity, vol. 109, pp. 383–392, 2012.

[7] V.A. Ingham, C.M. Jones, P. Pignatelli, V. Balabanidou, J. Vontas, S.C. Wagstaff, et al., "Dissecting the organ specificity of insecticide resistance candidate genes in *Anopheles gambiae*: known and novel candidate genes", BMC Genomics, vol. 15, pp. 1018, 2014.

[8] J. Yang, C. McCart, D.J. Woods, S. Terhzaz, K.G. Greenwood, R.H. ffrench-Constant, and J.A. Dow, "A *Drosophila* systems approach to xenobiotic metabolism", Physiol. Genomics, vol. 30, pp. 223-231, 2007.

[9] S.F. Altschul, T.L. Madden, A.A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D.J. Lipman, "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs", Nucl. Acids Res., vol. 25, pp. 3389-3402, 1997.

[10] H. Berman, K. Henrick, H. Nakamura, and J.L. Markley, "The worldwide Protein Data Bank (wwPDB): ensuring a

single, uniform archive of PDB data", Nucl. Acids Res., vol. 35, pp. D301-303, 2007.
[11] M.A. Larkin, G. Blackshields, N.P. Brown, R. Chenna, P.A. McGettigan, H. McWilliam, et al., 'ClustalW and ClustalX version 2", Bioinformatics, vol. 23, pp. 2947-2948, 2007.
[12] F. Sievers, A. Wilm, D.G. Dineen, T.J. Gibson, K. Karplus, W. Li, et al., "Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega", Mol. Syst. Biol., vol. 7 Article number: 539, pp. 1-6, 2011.
[13] M. Biasini, S. Bienert, A. Waterhouse, K. Arnold, G. Studer, T. Schmidt, et al., "SWISS-MODEL: modelling protein tertiary and quaternary structure using evolutionary information", Nucl. Acids Res., vol. 2014, pp. 1-7, 2014.
[14] R.A. Laskowski, M.W. MacArthur, D.S. Moss, and J. M.Thornton, "PROCHECK: a program to check the stereochemical quality of protein structures", J. Appl.Cryst., vol. 26, pp. 283-291, 1993.
[15] R.A. Laskowski, V.V. Chistyakov, and J.M. Thornton, "PDBsum more: new summaries and analyses of the known 3D structures of proteins and nucleic acids," Nucl. Acids Res., vol. 33, pp. D266-268, 2005.
[16] V.B. Chen, W.B. Arendall III, J.J. Headd, D.A. Keedy, R.M. Immormino, G.J. Kapral, et al., "MolProbity: all-atom structure validation for macromolecular crystallography", Acta Crystallogr., vol. D66, pp. 12-21, 2010.

Epidemic Analysis Using Traditional Model Checking and Stochastic Simulation

Kanishka Chauhan, Mh. Moiez Gohar and M. V. Panduranga Rao
Indian Institute of Technology Hyderabad, India

Abstract— *Stochastic model checking has been the mainstay for formal analysis of epidemic progression in recent years. However, such methods are sensitive to inaccuracies in estimating stochastic parameters like infection transmission and recovery rates. In this work, we revert to traditional model checking (specifically, for timed automata) to absorb inaccurately provided parameters into the nondeterminism inherent in such traditional formalisms. Parameters obtained through stochastic simulation are used by the timed automata, with sufficiently wide windows of nondeterminism to account for error. A positive side effect of this approach is that separating the probabilistic component helps focus on the progression logic while building the model.*

We demonstrate this approach using the model checker UPPAAL in conjunction with the stochastic epidemic modeling and simulation tool GLEAM.

Keywords: Epidemic Analysis, Timed Automata, TCTL Model Checking, Stochastic Simulation of Epidemics.

1. Introduction

Mathematical modeling of epidemics helps in predicting the progression of an epidemic among populations and across geographies. This in turn, helps in planning measures to control it. Epidemic modeling has been studied extensively in the past. While the initial models were deterministic in nature [1], recognition of the fact that epidemic propagation is inherently stochastic shifted the focus to probabilistic models. Indeed, several epidemics have been modeled in this fashion in the past. Initial stochastic models assumed homogenous mix in the populations–each individual in a population is connected to every other individual in the same manner [2]. Therefore, an infected individual can transmit the virus to any other individual with the same probability. However, it was realized that this is not a very realistic model as individuals have limited contacts. Edge weighted contact networks have since been used to capture neighbourhood [3]. Modeling and simulation efforts in these areas have yielded several interesting results as well as tools for studying the progression of epidemics. While simulations are useful for studying simple properties of the progression of an epidemic, more complex questions are difficult to answer. Moreover, a quantitative assurance of accuracy in the results is desirable.

Corresponding Author: M. V. Panduranga Rao email: mvp@iith.ac.in

With these in mind, researchers turned to statistical model checking. Mathematically precise formalisms (like Continuous Timed Markov Chains (CTMC)) are used to model epidemic progression among populations and queries regarding the model are framed in a suitable logic (like Probabilistic Computational Tree Logic (PCTL), Probabilistic Timed Computation Tree Logic (PTCTL), Continuous Stochastic Logic (CSL) etc). Following are some example queries of potential interest:

- What is the likelihood that at time t, at least c% of the population of a geographic region is affected?
- What is the likelihood that at time t, the carrier who imports the disease for the first time to a given geographic region arrives?

Model checking algorithms based on state space exploration or using statistical methods like hypothesis testing are then used to answer such queries with the required degree of confidence.

On the other hand, traditional model checking is done as follows. Given the description of a system in a mathematically precise formalism and the requirements expected of it as a formula in an appropriate (nonprobabilistic) system of logic, model checking algorithms attempt to verify if the described system satisfies the specified properties.

Among the two, statistical model checking gained traction in the field of epidemic analysis for several reasons. To begin with, the problem is fundamentally stochastic in nature and thus amenable to stochastic model checking. Secondly, this approach generally offers decent speed, if only at the expense of accuracy. And last but not the least, statistical model checking offers scope for more fine grained analysis. A statement like “there is a 10% chance of the epidemic reaching a particular geographic region in the next 50 days" is more useful than a simple affirmation that it is possible for the epidemic to reach in the next 50 days.

We report an approach that combines traditional model checking with stochastic simulation methods. The motivations are following. Firstly, this approach separates the stochastic element from the logic of epidemic progression and simplifies the process of designing the model. We appeal to stochastic simulations only to get an estimate of some of the delays required in the traditional model. Secondly, it is often difficult to accurately establish various probabilistic parameters of an epidemic model. We compensate for the loss of accuracy by absorbing it into the nondeterminism

available in the modeling formalism. Suppose that through stochastic simulation, we have estimates of the time t when $r\%$ of the population of a given geographic region gets infected by the epidemic. Then, our approach centers on allowing for these events to occur at *any* time during a window of time using nondeterminism–for example, between $t+\delta t$ and $t-\delta t$. The size of the window of nondeterminism can be chosen based on one's confidence on the accuracy of the stochastic simulation. Such a use of nondeterminism has been reported in the past in different contexts [4]. A final motivation is that there are differences in the expressiveness of various logic systems necessitating the framing of some queries in nonprobabilistic logics. Having traditional model checking approaches as well in the repertoire plugs that gap.

Timed automata, defined by Alur and Dill [5], have proved to be very useful for modeling and verification of timed systems. Together with formal specifications required of the system (in a temporal logic) and algorithms for model checking, it provides a technique for analyzing temporal behaviour of such systems. Model checking tools like UPPAAL [6], [7] have been developed and widely used for the purpose [8], [9], [10], [11].

Our approach involves the temporal parameters like (i) given that a certain geographic region is affected by the epidemic, how long does it take for the infection to appear in another region? (ii) how long does it take for the infection to assume epidemic proportions in a region, and (iii) how long does it take for a region to recover from the epidemic.

These parameters in turn depend on various socio-economic parameters like the connectivity between two regions, the strength of the healthcare system of the involved countries etc. For instance, the more connected a given country is to an affected country, the faster the epidemic reaches there. On the other hand, a strong healthcare system minimizes the duration for which a country remains affected. However, it is difficult to translate these parameters into the temporal parameters that we are looking for, for use in the timed automata model.

We get around this by leveraging the Global Epidemic and Mobility model, a tool that integrates real world data of the kind described above with stochastic models of epidemic propagation.

Following are some example queries as per our approach. Given that an infection starts in a certain part of the world on day 0,

- Will a certain geographic region be necessarily infected between days 120 and 150? Choice of venues for mass international events like sports meets in a certain interval of time, and planning medical logistics.
- Is it guaranteed that there will be some date after day 200 when all geographic regions will be free of the infection?

The paper is arranged as follows. The next section provides a brief discussion of existing work in the area. Section 3 provides a brief introduction to timed automata, UPPAAL and the GLEAM model. Section 4 describes our approach and discusses example queries and their results.

2. Previous Work

In this section, we briefly review some important recent work in the area of epidemic model checking. As discussed in the previous section, most of the literature uses stochastic model checking for analyzing epidemic progression. Drabik and Scatena [12] model the progression of an epidemic in a homogenous population using the PRISM model checker and ask queries framed in PCTL. Sam Huang [13] strikes a compromise between homogenous population models and contact networks among individuals by treating population that exhibit similar behavior as a node and establishing contact network between these nodes. This model is encoded as a CTMC and the queries are framed in CSL. However, it is not made clear how one obtains some crucial parameters like the weights on the edges between two meta-nodes, and their accuracy. This is not crucial for demonstrating the efficacy of vaccination strategies, as was indeed one of the chief aims of that work. Both approaches use the PRISM model checker for analysis and report experiments for small scale scenarios.

Ciochetta and Hillston [14] report an approach on the Bio-PEPA process algebra. In this approach also, the population is divided into homogenous subpopulations and the progression of the epidemic acorss various contact network topologies is investigated. For model checking, they derive a PRISM model from the Bio-PEPA model and frame queries in PCTL. They report that for the derived PRISM model, the verification of properties is not very efficient. In all the above mentioned works, the reported experiments are small scale–small populations divided among a small number of patches.

In a seminal work, Bortolussi and Hillston [15] use fluid approximation techniques in reducing the state space of the agent based CTMS for approximate model checking of epidemic propagation. The approach is complex, as the limit model of an individual is a time inhomogenous CTMC.

Complex networks and social networks provide a close imitation of interactions of individuals. Therefore, as research in these areas matured, results therein have been increasingly applied in understanding epidemic dynamics, cf. [16], [17], [18], [19], [20] and references therein. This also led to advances in related problems like propagation of computer viruses across networks [21].

3. Preliminaries

3.1 Timed automata

Following is a brief discussion about timed automata. Please read [5] for a detailed treatment. A timed automaton is essentially a finite state machine extended with clock

variables. It uses a dense time model where a clock variable evaluates to a real number and all clock variables progress synchronously.

Definition : A Timed Automaton A is a tuple (Q, Σ, C, E, q_0) where

- Q, a finite set of *locations* of A,
- q_0, an element of Q, called the initial location,
- Σ, a finite set called the alphabet or actions of A,
- C, a finite set called the clocks of A,
- $E \subseteq Q \times \Sigma \times B(C) \times P(C) \times Q$, a set of edges, called transitions of A, where
  - $B(C)$ is the set of boolean clock constraints involving clocks from C, and
  - $P(C)$ is the powerset of C,
- and a function $Inv :\rightarrow I(C)$ that associates with each location a set of boolean constraints on the clocks.

Informally, the automaton can remain in a location only as long as the clock values does not violate an *invariant* condition associated with that location as defined by the Inv function. On the other hand, it can jump to another location across an edge if a *guard* condition associated with the edge is satisfied. On taking a jump, it is possible to reset a subset of clocks. Thus, an edge (l, σ, b, p, l') represents a transition from location l to l', where σ is an action symbol and b is the guard condition associated with the edge, and p is the subset of clocks that need to be reset.

Such automata can be constructed to model real time reactive systems. An extremely useful fact is that it is possible to define parallel composition of timed automata. This facilitates writing timed automata models for complex systems.

Having constructed the timed automata model, one asks queries framed in a temporal logic regarding the temporal behaviour of such an automata. Model checking algorithms are designed to answer such queries.

There exists a variety of tools for the analysis of timed automata and its extensions, including model checkers like UPPAAL [6], [7]. In this paper we use UPPAAL for verification of our timed automata. The query language of UPPAAL, for specification of the properties to be verified in the timed automata model, is a subset of TCTL.

This subset includes queries of the form:

- $A <> \phi$: Always, ϕ will eventually hold true
- $A[]\phi$: Invariantly, ϕ is true
- $E <> \phi$: Eventually, ϕ will be true on some path
- $E[]\phi$: Potentially, ϕ is always true,

where $\phi := A.L \mid g_d \mid g_c \mid \phi$ and $\phi \mid \phi$ or $\phi \mid$ not $\phi \mid \phi$ imply $\phi \mid (\phi)$ for automaton A, location L, data guard g_d and clock guard g_c. Clock guards are predicates involving clocks: $x * c$ where $* \in \{<, <=, ==, >=, >\}$ for clock variable x and natural number c. Data guards are predicates over standard arithmetic expressions: $E * E$ where E is a standard arithmetic expression $* \in \{<, <=, ==, >=, >, ! =\}$. Finally, it is also possible to use more sophisticated queries like the "bounded liveness" query provided in UPPAAL: $\phi - - > (\psi$ and $x < c)$ which says that whenever ϕ is true, ψ is true within c time units.

3.2 The GLEAM model

In epidemiology, infections are often characterized using the *compartmental model*. At any given point in time, each individual of the population is in one of several possible "compartments"–susceptible, latent, infected and symptomatic, infected but not symptomatic, recovered etc. The individual then transits from one compartment to another with time. Together with interconnections between compartments, and the rates of inter-compartment transitions, one can characterize the infection. For example, an S-E-I-R-S model of infection makes an individual cycle through being susceptible, exposed, infected, recovered and being susceptible again.

To obtain values for the parameters to be used in the timed automata, we rely on the GLobal Epidemic and Mobility model (available publically as the GLEAMviz tool [22]), which models the progression of an epidemic as a stochastic process. GLEAM takes as input this characterization of the infection, together with other parameters like places where the infection originates, the fraction of the population infected initially in these places, connectivity parameters like flight occupance rates, commute times etc. GLEAM uses real-world data for populations, human mobility, and health care conditions in a locality, to enhance accuracy in prediction and analysis of the future course of an epidemic outbreak. The interested reader is referred to [23] for details.

Indeed, GLEAM has been used to predict the progression of epidemic outbreaks like the 2002-3 SARS [24], the 2009 H1N1 [25] and more recently, the Ebola outbreak of 2014 [26].

4. Modeling epidemic dynamics

We model geographical *regions* as vertices of a complete graph–that is, there exists an edge between every pair of regions. This essentially means that every region is connected to every other region through different means of transport.

One of these regions is designated as the originator of the epidemic. Corresponding to each geographic region, we construct a timed automaton that models the temporal behaviour of the epidemic in that region. We use a slightly different automaton for the geographic region where the infection originates. The automata are shown in Figures 1 and 2. Table 1 provides a look-up for the variables used in the automata. There are locations in each automaton that signify (i) when the region is unaffected by the epidemic (labeled by A, and an additional one time starting location A_begin for the originating region) (ii) the elapse of time at the region before the first infection appears because of some other region getting into its epidemic phase (labeled by B,

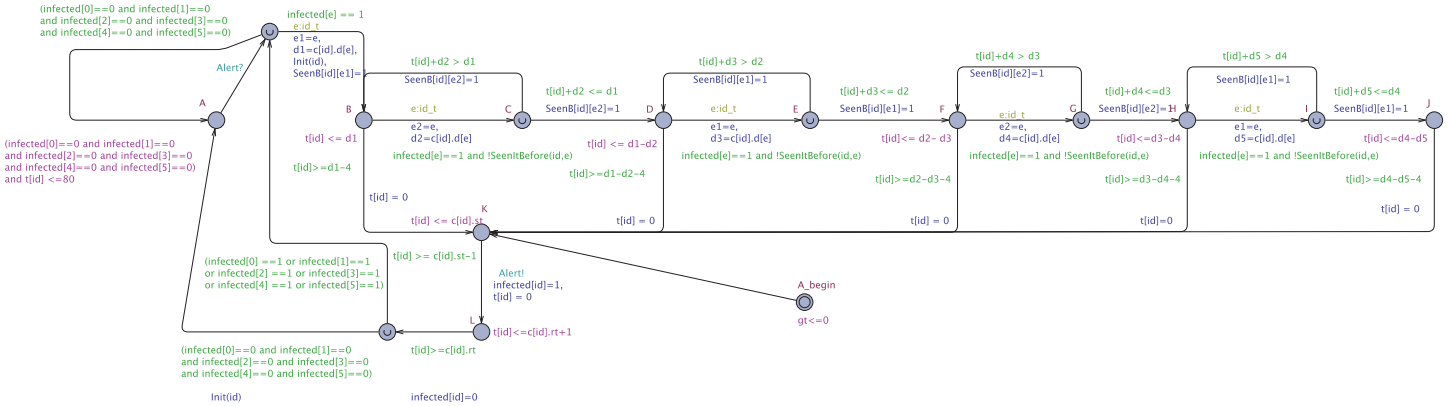


Fig. 1: Timed Automata for first infected region

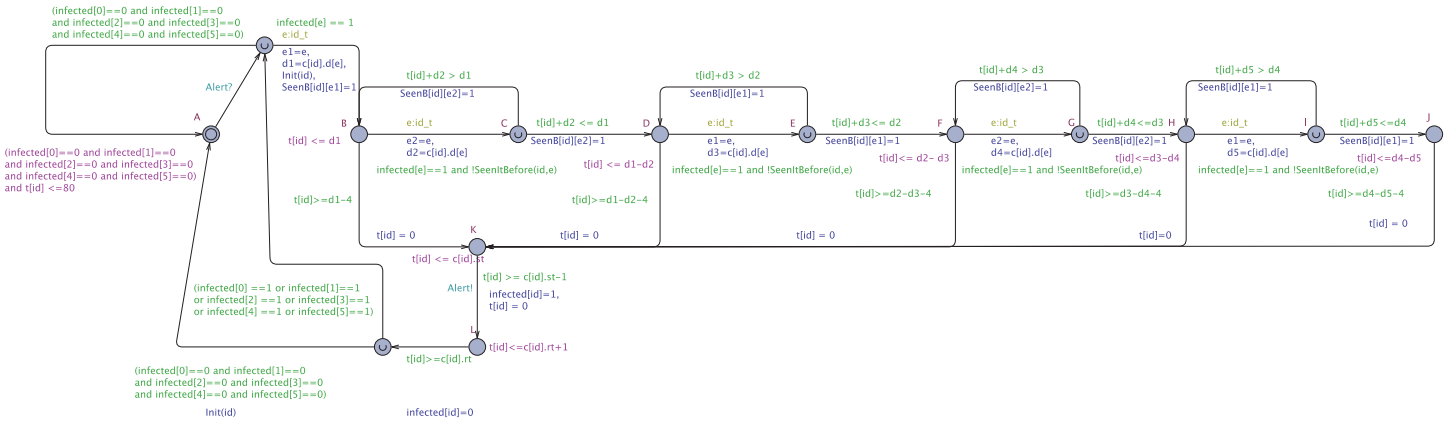


Fig. 2: Timed Automata for all other regions

Table 1: Notation

Term	Meaning
id	region identifier
t[id]	clock associated with region c[id].
e:id_ t	nondeterministic selection of region id
c[id].d[e]	delay at region c[id] before the first occurence of infection is brought in from c[e].
infected[i]	a boolean array that records whether a region i is affected by the epidemic or not.
infectCount	number of times the region c[id] goes into the epidemic phase in a given period of time.
c[id].st	time after the first infection appears but before the region reaches its epidemic phase.
c[id].rt	time needed at c[id] to recover from the epidemic.
init(id)	resets a region's local parameters like t[id] and c[id].infected.
seenB[]	an array for recording the regions from which epidemic alerts have appeared during the current epidemic cycle
seenItBefore[id][e]	to decide if an epidemic alert from region e has already been considered in the current cycle (makes use of seenB[])
gt	clock that keeps global time

D, F, H, J; detailed explanation later in the section) (iii) the elapse of time between the first appearance of the infection (carried in from abroad) in the region and its reaching the epidemic stage (labeled by K) and (iv) the time taken for the region to recover from the epidemic (labeled by L).

We first describe a simplistic scenario for ease of exposition. The automaton for every region is initially in its unaffected location A. The automaton of a region, say R_1, leaves this location only if some other region (say R_2) gets into the epidemic phase. The movement out of the unaffected location is triggered by "alerts" from R_2 in the form of synchronization and shared variables provided in UPPAAL. Even after R_2 gets into the epidemic phase, it takes some time for the infection to reach R_1 through a carrier. This delay[1], say d_{21}, depends primarily on the connectivity from region R_2 to R_1[2]. Moreover, let us assume an inaccuracy of δ in the stochastic estimation of d_{21}. Then, the automaton for R_1 waits nondeterministically in location B for a delay between $d_{21} - \delta$ and d_{21} time units. After this delay, the first incidence of the infection in R_1 occurs due to its connectivity with R_2. It takes st time units for R_1 to enter into epidemic phase. Assuming and error of δ' time units, the automaton waits nondeterministically in location K for a delay between $st - \delta'$ and st units. On completion of this delay, it transits to location L. This location signifies the full blown epidemic. On nondeterministically completing between rt and (a conservative) $rt + \delta''$ time units in the location L, the automaton returns to the unaffected location A if all other regions are uninfected. Otherwise, the infection cycle starts all over again for the region. As before, δ'' is the error that we anticipate in the stochastic simulation.

We emphasize that error estimates have been made on the conservative side. For first arrival and spread, we extend the window of nondeterminism in the negative side, whereas for recovery time, we extend it in the positive side.

This simplistic scenario can get complicated when another region (say R_3) with which R_1 has better connectivity gets into its epidemic phase while R_1 is still waiting for the infection to be brought in from R_2. Let the delay of getting the infection from R_2 be d_{31}. Further, let t_1 be the time at which region R_1 becomes susceptible on account of R_2 getting into its epidemic phase. Let the first infection carrier come to R_1 from R_2 at $t_1 + d_{21}$. Suppose at t such that $t_1 \leq t \leq t_1 + d_{21}$, R_3 enters into its epidemic phase. If $t_1 + d_{21} - t \geq d_{31}$, a infection carrier from R_3 will arrive at R_1 earlier than one from R_2, and the automaton changes location to reflect this. Otherwise, the automaton maintains status quo and waits for d_{21} units for the carrier from R_2. Note that multiple regions can enter into their epidemic locations at different points in time, potentially modifying the time of first arrival of a infection carrier. Locations B, D, F, H and J capture this–one, two, three, four and five regions getting into their respective epidemic location.

[1]Delays like these are obtained from GLEAMviz simulations.

[2]While the connectivity in terms of air, road or sea transport from R_2 to R_1 is expected to be the same in the opposite direction, we do not assume this.

4.1 GLEAM paramaters

The delays st and rt for a region to recover depends on the strength of its healthcare system. Parameters like the number of hospital beds per 10000 people are good indicators of this strength.

It remains to determine values of various parameters like these and the delays d_{ij} in the timed automata. As mentioned earlier, we use the GLEAM model and simulator for the purpose. We choose an SEIRS infection model with parameters as shown in Table 2. The simulation is run for a period of 365 days.

To obtain the delays d_{ij}, we choose i as the originating region and measure the delay before an infection appears first in region j. Doing this for all pairs in both the directions, we obtain the "Delay" columns of Table 3. The "spread time" is defined as the number of days elapsed between the first appearance of the infection in a region and 1% of the population getting infected. This is when the region enters the epidemic phase. Similarly, the "recovery time" is the number of days elapsed between entering the epidemic phase and the first day when no new infection appears[3]. These times are recorded; see columns labeled "Spread time" and "Recovery time" in Table 3.

In this discussion, we choose geographic regions at the granularity of continents for brevity of explanation. Thus, the world is divided into the six major continents. GLEAMviz allows a choice of finer granularity as well, namely subcontinental regions etc. For each continent, we identify some important, well connected cities as the origin of the infections. These values serve as parameters to the timed automata implemented in UPPAAL. In the timed automata, we choose Asia as the region where the infection originates in the first place.

4.2 The Queries

Table 4 shows some example queries and their answers. Queries 1 and 2 concern all the regions in different intervals of time, while 3 and 4 concern a specific region. For example, query number 1 asks whether at any time between day 300 and day 400 all the regions are necessarily affected by the epidemic. Similarly, query number 3 asks whether region 3 is necessarily affected between day 200 and day 300. Finally, the result of a bounded liveness queries says that it is not possible for region 2 to be affected on day 300 because region 0 is affected on day 260.

For these queries, we have taken $\delta = 4$ and $\delta' = \delta'' = 1$. A change in these values would change verdicts accordingly.

[3] These definitions can be changed as required.

References

[1] W. Kermack and A. McKendrick, "A contribution to the mathematical theory of epidemics," *Proc. R. Soc. A*, vol. 115, pp. 700–721, 1927.

[2] H. Abbey, "An examination of the reed frost theory of epidemics," *Human Biology*, vol. 24, pp. 201–233, 1952.

[3] L. A. Meyers, "Contact network epidemiology: Bond percolation applied to infectious disease prediction and control," *Bull. Amer. Math. Soc.*, vol. 44, pp. 63–86, 2007.

[4] L. de Alfaro, T. Henzinger, and R. Jhala, "Compositional methods for probabilistic systems," in *CONCUR 2001 âĂŤ Concurrency Theory*, ser. Lecture Notes in Computer Science, K. Larsen and M. Nielsen, Eds. Springer Berlin Heidelberg, 2001, vol. 2154, pp. 351–365. [Online]. Available: http://dx.doi.org/10.1007/3-540-44685-0_24

[5] R. Alur and D. L. Dill, "A theory of timed automata," *Theor. Comput. Sci.*, vol. 126, no. 2, pp. 183–235, Apr. 1994.

[6] K. G. Larsen, P. Pettersson, and W. Yi, "Uppaal in a nutshell," *International Journal on Software Tools for Technology Transfer*, vol. 1, no. 1-2, pp. 134–152, 1997.

[7] G. Behrmann, A. David, and K. Larsen, "A tutorial on uppaal," in *Formal Methods for the Design of Real-Time Systems*, ser. Lecture Notes in Computer Science, M. Bernardo and F. Corradini, Eds. Springer Berlin Heidelberg, 2004, vol. 3185, pp. 200–236.

[8] M. Lindahl, P. Pettersson, and W. Yi, "Formal Design and Analysis of a Gear-Box Controller," in *Proc. of the 4*th *Workshop on Tools and Algorithms for the Construction and Analysis of Systems*, ser. Lecture Notes in Computer Science, no. 1384. Springer–Verlag, Mar. 1998, pp. 281–297.

[9] J. Bengtsson, K. G. Larsen, F. Larsson, P. Pettersson, and W. Yi, "UPPAAL — a Tool Suite for Automatic Verification of Real–Time Systems," in *Proc. of Workshop on Verification and Control of Hybrid Systems III*, ser. Lecture Notes in Computer Science, no. 1066. Springer–Verlag, Oct. 1995, pp. 232–243.

[10] J. Bengtsson, W. D. Griffioen, K. J. Kristoffersen, K. G. Larsen, F. Larsson, P. Pettersson, and W. Yi, "Verification of an audio protocol with bus collision using UPPAAL," in *CAV96*, ser. LNCS, R. Alur and T. A. Henzinger, Eds., no. 1102. Springer–Verlag, Jul 1996, pp. 244–256.

[11] F. Larsson, P. Pettersson, and W. Yi, "On Memory-Block Traversal Problems in Model Checking Timed Systems," in *Proc. of the 6*th *Conference on Tools and Algorithms for the Construction and Analysis of Systems*, ser. Lecture Notes in Computer Science, S. Graf and M. Schwartzbach, Eds., no. 1785. Springer–Verlag, 2000, pp. 127–141.

[12] P. Drabik and G. Scatena, "An application of model checking to epidemiology (extended abstract)."

[13] S. Huang, "Probabilistic model checking of disease spread and prevention," *Technical Report, Computer Science Department, University of Maryland*, 2010.

[14] F. Ciocchetta and J. Hillston, "Bio-pepa for epidemiological models," *Electronic Notes in Theoretical Computer Science*, vol. 261, no. 0, pp. 43 – 69, 2010, proceedings of the Fourth International Workshop on the Practical Application of Stochastic Modelling (PASM 2009). [Online]. Available: http://www.sciencedirect.com/science/article/pii/S157106611000006X

[15] L. Bortolussi and J. Hillston, "Fluid model checking," in *CONCUR 2012 âĂŞ Concurrency Theory*, ser. Lecture Notes in Computer Science, M. Koutny and I. Ulidowski, Eds. Springer Berlin Heidelberg, 2012, vol. 7454, pp. 333–347. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-32940-1_24

[16] M. J. Keeling and K. T. D. Eames, "Networks and epidemic models," *Interface*, vol. 2, pp. 295–307, 2005.

[17] Y. Wang, D. Chakrabarti, C. Wang, and C. Faloutsos, "Epidemic spreading in real networks: an eigenvalue viewpoint," in *Reliable Distributed Systems, 2003. Proceedings. 22nd International Symposium on*, Oct 2003, pp. 25–34.

[18] M. Kuperman, "Invited review: Epidemics on social networks," *Papers in Physics*, vol. 5, pp. 050 003–1–050 003–17, 2013.

[19] M. Newman, "The spread of epidemic diseases on networks," *Physical Review E*, vol. 66, p. 016128, 2002.

[20] M. Deijfen, "Epidemics on social network graphs," 2000.

[21] D. Chakrabarti, Y. Wang, C. Wang, J. Leskovec, and C. Faloutsos, "Epidemic thresholds in real networks," *ACM Trans. Inf. Syst. Secur.*, vol. 10, no. 4, pp. 1:1–1:26, Jan. 2008.

[22] [Online]. Available: http://www.gleamviz.org

[23] W. Broeck, C. Gioannini, B. GonÃąalves, M. Quaggiotto, V. Colizza, and A. Vespignani, "The gleamviz computational tool, a publicly available software to explore realistic epidemic spreading scenarios at the global scale," *BMC Infectious Diseases*, vol. 11, no. 37, 2011.

[24] V. Colizza, A. Barrat, M. Barthelemy, and A. Vespignani, "Predictability and epidemic pathways in global outbreaks of infectious diseases: the sars case study," *BMC Medicine*, vol. 5, no. 34, 2007.

[25] P. Bajardi, C. Poletto, J. J. Ramasco, M. Tizzoni, V. Colizza, and A. Vespignani, "Human mobility networks, travel restrictions, and the global spread of 2009 h1n1 pandemic," *PLoS ONE*, vol. 6, no. 1, 2011.

[26] M. Gomes, A. P. y. Piontti, L. Rossi, D. Chao, I. Longini, M. Halloran, and A. Vespignani, "Assessing the international spreading risk associated with the 2014 west african ebola outbreak. plos currents outbreaks." *PLoS Currents Outbreaks*, vol. 1, 2014.

Table 2: GLEAM parameters

Continent	**Originating cities**
Asia	Beijing, Shanghai, Tokyo, Seoul, Mumbai
Africa	Lagos, Cairo, Johannesberg, Nairobi
Europe	London, Paris, Madrid, Rome, Berlin, Istanbul
South America	Sao Paulo, Rio de Janeiro, Buenos Aires, Lima
North America	New York, Los Angeles, Chicago, Mexico City, Toronto
Australia	Sydney, Melbourne, Brisbane
Compartment transition rates (SEIRS)	
Susceptible–Latent	0.835, 0.417 Due to Infectious and Asymptomatic Infectious respectively
Latent–Infectious and Symptomatic	2/3.3
Latent–Infectious but Asymptomatic	1/3.3
Infected and Symptomatic–Recovered	0.6
Infected but Asymptomatic–Recovered	0.6
Recovered–Symptomatic	0.0001
Miscellaneous	
Start Date	18/12/2014
Flight Occupancy Rate	90%
Time Spent at commuting destination	8 hrs
Fraction of population infected at origin (Mumbai)	0.00042

Table 3: Parameters of the system

Continent	**Delay**						**Spread time**	**Recovery time**
Asia	2	18	4	23	8	8	40	165
Africa	10	2	6	22	18	20	65	175
Europe	6	6	2	6	4	23	57	170
South America	27	21	10	2	8	24	76	169
North America	8	18	4	2	2	8	60	174
Australia	8	20	21	32	15	2	61	172

Table 4: Example Queries

S. No.	TCTL Query	Verdict
1	A<> gt >= 300 and gt<=400 and infected[0]==1 and infected[1]==1 and infected[2]==1 and infected[3]==1 and infected[4]==1 and infected[5]==1	N
2	A<> gt>=200 and infected[0]==0 and infected[1]==0 and infected[2]==0 and infected[3]==0 and infected[4]==0 and infected[5]==0	N
3	A<> gt<=300 and gt>=200 and infected[3]==1	Y
4	A<> gt<=150 and gt>=120 and infected[3]==0	Y
5	infected[0]==1 and gt==260 −− > infected[2]==1 and gt==300	N