Dense module enumeration in biological networks

Koji Tsuda¹ and Elisabeth Georgii²,³

¹Computational Biology Research Center, National Institute of Advanced Industrial Science and Technology (AIST), Tokyo, Japan
²Max Planck Institute for Biological Cybernetics, Tübingen, Germany
³Friedrich Miescher Laboratory of the Max Planck Society, Tübingen, Germany

E-mail: koji.tsuda@aist.go.jp

Abstract. Analysis of large networks is a central topic in various research fields including biology, sociology, and web mining. Detection of dense modules (a.k.a. clusters) is an important step to analyze the networks. Though numerous methods have been proposed to this aim, they often lack mathematical rigorousness. Namely, there is no guarantee that all dense modules are detected. Here, we present a novel reverse-search-based method for enumerating all dense modules. Furthermore, constraints from additional data sources such as gene expression profiles or customer profiles can be integrated, so that we can systematically detect dense modules with interesting profiles. We report successful applications in human protein interaction network analyses.

1. Introduction

Today, a large number of databases provide access to experimentally observed protein-protein interactions. The analysis of the corresponding protein interaction networks can be useful for functional annotation of previously uncharacterized genes as well as for revealing additional functionality of known genes. Often, function prediction involves an intermediate step where clusters of densely interacting proteins, called modules, are extracted from the network; the dense subgraphs are likely to represent functional protein complexes [1]. However, the experimental methods are not always reliable, which means that the interaction network may contain false positive edges. Therefore, confidence weights of interactions should be taken into account.

A natural criterion that combines these two aspects is the average pairwise interaction weight within a module (assuming a weight of zero for unobserved interactions, cf. Ulitsky et al.[2]). We call this the module density, in analogy to unweighted networks [3]. We present a method to enumerate all modules that exceed a given density threshold. It solves the problem efficiently via a simple and elegant reverse search algorithm, extending the unweighted network approach in [4].

There is a large variety of related work on module discovery in networks. The most common group are graph partitioning methods[5, 6, 7]. They divide the network into a set of modules, so their approach is substantially different from DME, which provides an explicit density criterion for modules (Fig. 1A). Another group of methods define explicit module criteria, but employ heuristic search techniques to find the modules [8, 3]. This contrasts with complete enumeration algorithms, which form the third line of research: they give explicit criteria and return all modules that satisfy them. For example, clique search has been frequently applied [9, 10]. The enumeration of cliques can be considered as a special case of our approach, restricting it to
unweighted graphs and a density threshold of 1. Further enumerative approaches use different module criteria assuming unweighted graphs [11, 12].

In recent years, many module finding approaches which integrate protein-protein interaction networks with other gene-related data have been published. One strategy, often used in the context of partitioning methods, is to build a new network whose edge weights are determined by multiple data sources [13]. Tanay et al. [14] also create one single network to analyze multiple genomic data at once; however, they use a bipartite network where each edge corresponds to one data type only. In both cases, the different data sets have to be normalized appropriately before they can be integrated. In contrast to that, other approaches keep the data sources separate and define individual constraints for each of them. Consequently, arbitrarily many data sets can be jointly analyzed without the need to take care of appropriate scaling or normalization. Within this class of approaches, there exist two main strategies to deal with profile data like gene expression measurements. In the first case, the profile information is transformed into a gene similarity network, where the strength of a link between two genes represents the global similarity of their profiles [15, 2, 16]. In the second case, the condition-specific information is kept to perform a context-dependent module analysis [17, 18, 19]. Our approach follows along this line, searching for modules in the protein interaction network that have consistent profiles with respect to a subset of conditions. In contrast to the previous methods, our algorithm systematically identifies all modules satisfying a density criterion and optional consistency constraints.

In this study, we evaluate our approach on the yeast interaction network in comparison with four other methods. Also, we discuss our results obtained from human protein interactions in the context of gene expression data.

2. Dense Module Enumeration

In this section, we describe the basic idea of dense module enumeration (DME) using the examplar graph shown in Fig. 2. Our method is based on the reverse search paradigm [20], which is quite popular in the algorithm community[^1], but only in a limited degree known in the data mining community. A weighted graph is represented as a symmetric association matrix (edges that are not shown have zero weight). We denote by \( w_{ij} \) the weight between two nodes, and define the density of a node subset \( U \) as

\[
\rho(U) = \frac{\sum_{i,j \in U, i<j} w_{ij}}{|U|(|U| - 1)/2}.
\]

We would like to enumerate all subsets \( U \) with \( \rho(U) \geq \theta \), where \( \theta \) is a prespecified constant.

All subsets form a natural graph-shaped search space, where one can move downwards or upwards by adding or removing a node, respectively (Figure 3a). Here, the root node corresponds to the empty set. For efficient traversal, however, one needs a spanning tree, not a graph. When a tree is made by lexicographical ordering (Figure 3b), the search tree is not anti-monotonic with respect to the density. Namely, the density is not monotonically decreasing when the tree is traversed from the root to a leaf. This property disallows early pruning and makes the enumeration difficult. However, there exists indeed a search tree which is anti-monotonic (Figure 3c). It can be constructed by reverse search.

In reverse search, the search tree is specified by defining a reduction map \( f(U) \) which transforms a child to its parent. In our case, the parent is created by removing the node with minimum degree from the child. Here, the degree of a node is defined as the sum of weights of all adjacent edges within \( U \). If there are multiple nodes with minimum degree, the one with the smallest index is removed. In [21], it is proven that the density of a parent is at least as

[^1]: See a list of applications at [http://cgm.cs.mcgill.ca/~avis/doc/rs/applications/index.html](http://cgm.cs.mcgill.ca/~avis/doc/rs/applications/index.html)
**Figure 1. Dense Module Enumeration approach.** A. DME versus partitioning. While partitioning methods return one clustering of the network, DME discovers all modules that satisfy a minimum density threshold. B. Combination with profile data. Integration of protein-protein interaction (PPI) and external profile data allows to focus on modules with consistent behavior of all member proteins in a subset of conditions. The top module has two conditions where all nodes are positive and one condition where all nodes are negative. The arrows in the profile shows such consistent conditions. On the other hand, the bottom module does not have such consistency.

**Figure 2.** An examplar graph for dense module enumeration.
high as the maximum density among the children, ensuring that the search tree induced by the reduction map is anti-monotonic.

In addition to the anti-monotonicity property, a valid reduction map must satisfy the following reachability condition [20]: starting from any node of the search tree, we can reach a root node after applying the reduction map a finite number of times. This condition ensures that the induced search tree is indeed spanning. For the reduction map stated above, it is trivial to show that the reachability condition is satisfied, because any cluster shrinks to the empty set by removing nodes repeatedly.

To enumerate all clusters with density \( \geq \theta \), one has to traverse this implicitly defined search tree in a depth-first or breadth-first manner. During traversal, children are generated on demand. As the reduction map defines how to get from children to parents and not vice versa, we cannot directly derive the children from a given parent. Instead, to generate the children of a cluster \( U \), we have to consider all candidates \( U \cup \{i\} \), \( i \notin U \) and apply the reduction map to every candidate (reverse search principle). Qualified candidates with \( f(U \cup \{i\}) = U \) are then taken as children. A naive implementation of this child generation process can make the algorithm very slow. Thus, it is important to engineer this process well. As the search tree is anti-monotonic, one can prune the tree whenever the density goes below \( \theta \). See [22] for detailed evaluation of computational complexity.

The definition of a search tree is not an issue in the context of frequent pattern mining [23], because frequency is anti-monotonic in any tree. Reverse search is interesting because it provides a systematic way of defining an anti-monotonic tree. Notice, however, that it is not applicable...
Cluster density is an example where reverse search can be applied most effectively.

### 3. Integration of Additional Constraints

The DME framework makes it easy to incorporate and systematically exploit constraints from additional data sources. For illustration, consider the case where we have an additional data set which provides profiles of proteins or genes across different conditions (Fig. 1B). For simplicity, let us assume binary profiles, being 1 if the protein is positively associated with the corresponding condition, and 0 otherwise. Then, dense modules where all member proteins share the same profile across a certain number of conditions are of particular interest; we call these modules consistent. The problem of dense module enumeration with consistency constraints is formalized as follows.

**Definition 1.** Given a graph with node set \( V \) and weight matrix \( W \), a density threshold \( \theta > 0 \), a profile matrix \((m_{ij})_{i\in V, j\in C}\), and non-negative integers \( n_0 \) and \( n_1 \), find all modules \( U \subset V \) with \( \rho_W(U) \geq \theta \) s.t. there exist at least \( n_0 \) conditions \( c \in C \) with \( m_{uc} = 0 \) \( \forall u \in U \) and there exist at least \( n_1 \) \( c \in C \) with \( m_{uc} = 1 \) \( \forall u \in U \).

Given such a consistency constraint, we can stop the module extension during the dense module mining as soon as the constraint is violated. This is due to the fact that the number of consistent profile conditions cannot increase while extending the module; more generally, this property is called anti-monotonicity. So we simply add to the module enumeration algorithm a condition which checks for the consistency requirements. These are then automatically taken into account in the check for local maximality. The use of additional constraints can restrict the search space considerably, so it accelerates the computation and helps to focus on biologically interesting solutions.

### 4. Module Ranking

The exhaustiveness of our DME approach enables us to exactly determine the uncommonness of the discovered substructures with respect to the network at hand. Let \( W = (w_{ij})_{i,j \in V} \) be the matrix representation of the given network; the total number of nodes is denoted by \( |V| \). Let \( U \) be a module with \( |U| \) nodes and density \( \rho_W(U) \). Then, the probability that a random selection of \( |U| \) nodes in the network produces a module with at least the same density as \( U \) is given by

\[
\left| \left\{ U' \subset V : |U'| = |U| \land \rho_W(U') \geq \rho_W(U) \right\} \right| \left/ \binom{|V|}{|U|} \right. .
\]

The exact value of the numerator can be obtained as a side product of the DME algorithm. In the case of additional constraints, it includes only modules that satisfy them. The modules in the DME output are sorted by their probability values (in ascending order). This ranking scheme captures the intuition that the rank of a module should increase with its size and density, but from a theoretical point of view it is more principled than the ranking criterion used in [3], which is the product of size and density. Furthermore, our probability calculation refers specifically to the network at hand, in contrast to measures derived from network models [24].

### 5. Experimental Results

First, we validated the performance of DME on the yeast interaction network in comparison with four other methods: clique detection (Clique)\(^2\), the clique percolation method (CPM)\(^2\) [9], a procedure for joining cliques of a certain size to larger clusters, CPMw\(^2\) [25], an extension

\(^2\) implementation from [http://www.cfinder.org](http://www.cfinder.org)
Recall
Precision
DME
Clique
CPM
CPMw
MCL

Figure 4. Comparative precision-recall analysis. To account for module overlap, the measures are based on protein pairs, see text.

of CPM which includes an additional clique filtering step, and Markov clustering (MCL)\(^3\) [6], a popular graph clustering method simulating random walks. As a reference set of confirmed complexes, we used the manually curated protein complexes provided by MIPS [26]. To properly assess methods which can produce overlapping modules, we chose performance measures that are based on protein pairs rather than modules; in that way, we avoid taking the same subset of nodes several times into account even if it occurs in more than one module. Defining the intersection of pairs from predicted modules and pairs from known complexes as correctly predicted pairs, we calculated precision and recall as follows.

\[
\text{Precision} = \frac{\text{nr of correctly predicted protein pairs}}{\text{nr of protein pairs in predicted modules}}
\]

\[
\text{Recall} = \frac{\text{nr of correctly predicted protein pairs}}{\text{nr of protein pairs in known complexes}}
\]

To obtain precision-recall curves, we iteratively calculated the precision and recall values, each time extending the set of considered modules by the next highest ranking module. As the other methods do not provide a module ranking and our criterion is only applicable to enumerative approaches, we used the scoring scheme by Bader et al.[3]. In fact, it produced for our DME results almost the same ranking as our criterion; the corresponding precision-recall curves are virtually equivalent. Clique and CPM cannot handle edge weights directly, but they preselect edges according to a minimum weight threshold. Due to space constraints, we cannot describe the full details of protein interaction networks and their statistics. Please refer to [21].

The precision-recall curves are shown in Fig. 4. Overall, DME shows the best prediction performance. It has high precision with respect to the highest ranking modules and then shows a sudden drop, which is due to a big module not annotated as a known complex. Clique detects the same module, but there are some other higher ranked modules, so the drop happens later. MCL and CPM stay always below DME. Clique works quite well, however the precision drops quickly for higher recall values because edge weights are not taken into account. It seems that DME has a clear advantage compared to CPM: by explicitly using the edge weights and tuning the density parameter, it allows for more flexibility than the two-stage procedure of CPM, first

\(^3\) implementation from http://micans.org/mcl
selecting edges and subsequently joining together cliques that satisfy an overlap criterion. While CPMw allows to refine the module search, it still differs significantly from our approach. As it joins preselected cliques, it does not control directly the density of the produced modules and might also miss some dense modules. In our analysis, CPMw improved the result obtained by CPM, but is mostly inferior to Clique or DME. We deliberately determined the parameters of our method and the competing methods to make a fair comparison. Please refer to the supplementary report of [21].

Table 1 summarizes further statistics for the predicted modules. As DME and Clique produced a large number of very similar modules, we computed for better comparability the number of distinct modules. For that purpose, we grouped matching modules into clusters; each cluster was represented by its top-ranking module. To decide whether two modules match each other, we here computed the overlap score proposed by Bader et al.[3], using a stringent cutoff of 0.5. It is defined as the fraction of overlapping proteins with respect to the size of the first module multiplied by the fraction of overlapping proteins with respect to the size of the second module. The same criterion was used to determine matches between predicted modules and known complexes. While DME and Clique discovered a comparable number of distinct modules, the DME modules match many more known complexes. Among these, we also find small-sized complexes, so the overall average size of retrieved complexes is lower than for Clique. In addition, we report the number of complexes from which at least one protein pair was recovered as well as the area under the precision-recall curve from the pairwise analysis (Fig. 4). In both cases, DME is leading. Furthermore, we investigated the enrichment of the distinct modules with respect to Gene Ontology (GO) terms. For that purpose, we applied the TANGO tool [27] using the default setting with \( P \)-value threshold 0.05 after correction for multiple testing. Beside the total number of enriched modules, we also counted the number of enriched modules among the top-50 distinct modules, showing that for each method that produced more than 50 modules, most of the high-ranking modules satisfy the enrichment threshold. For small modules the enrichment test fails even if they are totally pure.

Table 1. Module statistics of the comparative analysis (see text for details). The average size of the raw modules can be larger than for the distinct modules because larger modules allow for more variants. The time measurements were performed on a 2.2 GHz processor.

|                      | DME  | Clique | CPM  | CPMw | MCL |
|----------------------|------|--------|------|------|-----|
| # distinct modules   | 1083 | 916    | 19   | 32   | 648 |
| average size of distinct modules | 3    | 4      | 16   | 14   | 3   |
| # raw modules        | 2480 | 3      | 1971 | 19   | 33  |
| average size of raw modules | 10   | 6      | 16   | 14   | 3   |
| # matched complexes  | 84   | 54     | 9    | 20   | 59  |
| average complex size | 5    | 7      | 19   | 14   | 7   |
| # partially recovered complexes | 133  | 107    | 20   | 33   | 117 |
| area under prec.-rec. curve (AUC) | 0.183 | 0.166 | 0.107 | 0.153 | 0.148 |
| # enriched distinct modules | 112  | 131    | 18   | 32   | 69  |
| # enriched among top-50 | 47   | 44     | -    | -    | 45  |
| # overlapping proteins | 1010 | 1113   | 12   | 38   | 1   |
| # overlapping interactions | 3664 | 4340   | 24   | 114  | 0   |
| AUC for overlapping interactions | 0.152 | 0.082 | 0.000 | 0.001 | -   |
| # recovered complex overlaps | 18   | 16     | 0    | 4    | 0   |
| running time (in seconds) | 2667 | 6      | 5    | 457  | 4   |
6. Tissue-Specific Modules in the Human Interaction Network

In this section, we discover tissue-specific modules of the human interaction network. As side information, we downloaded the gene expression profiles by Su et al. [28], containing measurements in 79 different human tissues and present/absent/marginal calls. For our purposes, we considered a gene to be expressed in a given tissue only if it was called as present in both of the duplicated measurements. In order to find complexes which are present in several, but not all tissues, we applied DME to enumerate all modules that are consistently expressed in at least three tissues and consistently not expressed in at least ten tissues. We used again the same procedure for selecting the density parameter and ended up with 460 distinct modules.

The two top-ranking modules cover the MCM complex (Fig. 5A). As reference, we used a manually curated set of human complexes collected by MIPS [29]. MCM is a hexameric protein complex required for the initiation and regulation of eukaryotic DNA replication. The DME modules contain two additional proteins, Ssrp1 and Orc6l. Orc6l is a member of the origin recognition complex (ORC), which plays a central role in replication initiation; in fact, the MCM and ORC complexes form the key components of the pre-replication complex [30]. This is nicely reflected by the high interaction density as well as the common expression profiles of the proteins: the module is fully expressed in three different types of bone marrow cells and fully non-expressed in 42 tissues like brain, liver, and kidney, where cells are differentiated and divide less. Ssrp1 is a member of the FACT complex, which is involved in chromatin reorganization [31].

Moreover, our analysis yields some insights about the tissue-specific reorganization of the SCF E3 ubiquitin ligase complex, which marks proteins for degradation. Fig. 5B depicts the five top-ranking modules that cover the complex (beyond those, there were three other modules covering only a single protein of the complex). One of them contains as additional component Cand1, a regulatory protein that inhibits the interaction of Cul1 with Skp1 [32]. The other four peripheral proteins are F-box proteins, which serve as substrate recognition particles for the SCF complex. Interestingly, the corresponding modules show different tissue specificities, indicating that the target proteins of SCF are selected in a tissue-dependent manner. This finding is in accordance with experimental studies [33, 34, 35]. On one hand, it has been shown that in human cells multiple variants of the SCF complex exist, each one containing a different F-box protein for substrate recognition. On the other hand, brain and blood cells have been identified as tissues of major expression for some F-box components, and expression variation of F-box components has been observed in several tissues like testis, prostate, and placenta. In our results, all detected module variants are present in natural killer (nk) cells, which play an important role in immune response [36], whereas only a few are present in B-cells and testis; in certain brain regions, for instance medulla oblongata, only the module variant with Fbxw7 is predicted to be active. As illustrated by this example, DME integrated with gene expression data can be a powerful tool to reveal functional and condition-specific variants of protein complexes.

7. Conclusion

Our algorithm, DME, extracts all densely connected modules from a given weighted interaction network. In addition to its completeness guarantee, a strength of the method lies in the possibility of transparent data integration, which is of crucial importance in biological applications. Due to its generality, we believe that DME is a useful tool in many different systems biology approaches. Our framework can also solve more general problems arising in the analysis of structured data, like dense subgraph detection in multi-partite graphs [14, 8] or in hypergraphs [37]. Moreover, module finding can assist in network comparison and classification tasks [38].
Figure 5. Tissue-specific modules in human. A. The two top-ranking modules, covering the MCM complex. Known complexes are indicated as solid ellipses, modules as dashed ellipses. B. Top-five modules around the SCF ubiquitin ligase complex, revealing its tissue-specific organization. Boxes show the tissues of consistent positive expression for the respective module. Tissues associated with all modules are marked in bold, uniquely appearing tissues in italics.

8. Acknowledgment

This proceedings paper summarizes two of our previous papers [21, 22] for the talk at IW-SMI 2009. A C++ implementation of the algorithm is available at http://www.kyb.tuebingen.mpg.de/~georgii/dme.html. KT is supported by MEXT Kakenhi 21680025 and the Functional RNA Project of New Energy and Industrial Technology Development Organization (NEDO).

Reference

[1] Sharan R, Ullitksy I and Shamir R 2007 Mol. Syst. Biol. 3 88
[2] Ulitsky I and Shamir R 2007 BMC Syst. Biol. 1 8
[3] Bader G D and Hogue C W 2003 BMC Bioinformatics 4 2
[4] Uno T 2007 in Proc. of ISAAC 2007 pp 402–14
[5] Chen J and Yuan B 2006 Bioinformatics 22(18) 2283–90
[6] van Dongen S 2000 Graph Clustering by Flow Simulation Ph.D. thesis University of Utrecht
[7] Newman M E 2006 Proc. Natl. Acad. Sci. USA 103(23) 8577–82
[8] Everett L, Wang L S and Hannenhalli S 2006 Bioinformatics 22(14) e117–23
[9] Palla G, Derenyi I, Farkas I and Vicsek T 2005 Nature 435(7043) 814–8
[10] Spirin V and Mirny L A 2003 Proc. Natl. Acad. Sci. USA 100(21) 12123–8
[11] Haraguchi M and Okubo Y 2006 in Federation over the Web vol 3847 of Lecture Notes in Computer Science (Springer)
[12] Zeng Z, Wang J, Zhou L and Karypis G 2006 in KDD '06: Proc. of the 12th ACM SIGKDD Int. Conf. on Knowledge Discovery in Data Mining (New York, NY, USA: ACM) pp 797–802 ISBN 1-59593-339-5
[13] Hanisch D, Zien A, Zimmer R and Lengauer T 2002 Bioinformatics 18(suppl_1) S145–54
[14] Tanay A, Sharan R, Kupiec M and Shamir R 2004 Proc. Natl. Acad. Sci. USA 101(9) 2981–6
[15] Segal E, Wang H and Koller D 2003 Bioinformatics 19(suppl_1) i264–71
[16] Pei J, Jiang D and Zhang A 2005 in ICDE ’05: Proc. of the 21st Int. Conf. on Data Engineering (ICDE’05) (Washington, DC, USA: IEEE Computer Society) pp 353–4 ISBN 0-7695-2285-8
[17] Ideker T, Ozier O, Schwikowski B and Siegel A F 2002 Bioinformatics 18(suppl_1) S233–40
[18] Huang Y, Li H, Hu H, Yan X, Waterman M S, Huang H and Zhou X J 2007 Bioinformatics 23(13) i222–9
[19] Yan X, Mehan M R, Huang Y, Waterman M S, Yu P S and Zhou X J 2007 Bioinformatics 23(13) i577–86
[20] Avis D and Fukuda K 1996 Discrete Appl. Math. 65 21–46
[21] Georgii E, Dietmann S, Uno T, Pagel P and Tsuda K 2009 Bioinformatics 25 933–40
[22] Georgii E, Tsuda K and Schölkopf B 2009 in Proc. of the 2009 Workshop on Data Mining using Matrices and Tensors (DMMT’09) pp 32–41
[23] Han J and Kamber M 2006 Data Mining: Concepts and Techniques 2nd ed The Morgan Kaufmann Series in Data Management Systems (San Francisco, CA: Morgan Kaufmann Publishers)

[24] Koyuturk M, Szpankowski W and Grama A 2007 J. Comput. Biol. 14(6) 747–64

[25] Farkas I J, Abel D, Palla G and Vicsek T 2007 New J. Phys. 9(180)

[26] Guldener U, Munsterkotter M, Kastenmuller G, Strack N, van Helden J, Lemer C, Richelles J, Wodak S J, Garcia-Martinez J, Perez-Ortin J E, Michael H, Kaps A, Talla E, Dujon B, Andre B, Souciet J L, De Montigny J, Bon E, Gaillardin C and Mewes H W 2005 Nucl. Acids Res. 33(suppl_1) D364–8

[27] Shamir R, Maron-Katz A, Tanay A, Linhart C, Steinfeld I, Sharan R, Shiloh Y and Elkon R 2005 BMC Bioinformatics 6(1) 232 ISSN 1471-2105 URL http://www.biomedcentral.com/1471-2105/6/232

[28] Su A I, Wiltshire T, Batalov S, Lapp H, Ching K A, Block D, Zhang J, Soden R, Hayakawa M, Kreiman G, Cooke M P, Walker J R and Hogenesch J B 2004 Proc. Natl. Acad. Sci. USA 101(16) 6062–7

[29] Ruepp A, Brauner B, Dunger-Kaltenbach I, Frishman G, Montrone C, Stransky M, Waegle B, Schmidt T, Donidieu O N, Stumpflen V and Mewes H W 2008 Nucl. Acids Res. 36(suppl_1) D646–50

[30] Lei M and Tye B K 2001 J. Cell Sci. 114(Pt 8) 1447–54

[31] Orphanides G, Wu W H, Lane W S, Hampsey M and Reinberg D 1999 Nature 400(6741) 284–8

[32] Zheng J, Yang X, Harrel J M, Ryzhikov S, Shim E H, Lykke-Andersen K, Wei N, Sun H, Kobayashi R and Zhang H 2002 Mol. Cell 10(6) 1519–1526

[33] Cenciarelli C, Chiaur D, Guardavaccaro D, Parks W, Vidal M and Pagano M 1999 Curr. Biol. 9 1177–9

[34] Kirov E and Pagano M 2000 Genome Biology 1(5) reviews3002.1–reviews3002.7 ISSN 1465-6906

[35] Koepp D M, Schaefer L K, Ye X, Keyomarsi K, Chu C, Harper J W and Elledge S J 2001 Science 294(5540) 173–7

[36] Janeway C, Travers P, Walport M and Shlomchik M 2005 Immunobiology: Immune System in Health and Disease (Garland Publishing)

[37] Zhao L and Zaki M J 2005 in SIGMOD ’05: Proc. of the 2005 ACM SIGMOD Int. Conf. on Management of Data (New York, NY, USA: ACM) pp 694–705 ISBN 1-59593-060-4

[38] Chuang H Y, Lee E, Liu Y T, Lee D and Ideker T 2007 Mol. Syst. Biol. 3 140