Title: Constructing module maps for integrated analysis of heterogeneous biological networks

Running title: From heterogeneous networks to module maps

David Amar[1] and Ron Shamir[1*]

1. Blavatnik School of Computer Science, Tel Aviv University, Tel Aviv, Israel

* E-mail: rshamir@tau.ac.il

Keywords: modularity, biological networks, protein-protein interactions, genetic interactions, systems biology.

Abstract

Improved methods for integrated analysis of heterogeneous large-scale omic data are direly needed. Here we take a network-based approach to this challenge. Given two networks, representing different types of gene interactions, we construct a map of linked modules, where modules are genes strongly connected in the first network and links represent strong inter-module connections in the second. We develop novel algorithms that considerably outperform prior art on simulated and real data from three distinct domains. First, by analyzing protein-protein interactions and negative genetic interactions in yeast we discover epistatic relations among protein complexes. Second, we analyze protein-protein interactions and DNA damage-specific positive genetic interactions in yeast, and reveal functional rewiring among protein complexes, suggesting novel mechanisms of DNA damage response. Finally, using transcriptomes of non-small cell lung cancer patients, we analyze networks of global co-expression and disease-dependent differential co-expression, and identify a sharp drop in correlation between two modules of immune activation processes, with possible microRNA control. Our study demonstrates that module maps are a powerful tool for deeper analysis of heterogeneous high throughput omic data.

Introduction

Biological networks provide a comprehensive overview of biological systems. They enable better understanding of the system and can shed light on the function of genes and other molecular compounds. Among other applications, they have been utilized for discovery and prediction of gene interactions, gene functions, and disease-genes associations (1-9).

In these networks, nodes represent molecular entities and the edges represent interdependencies. For example, in protein-protein interaction (PPI) networks nodes represent proteins and edges represent physical interactions. In genetic interaction (GI) networks, nodes represent genes and edges represent the organism fitness for double knockout perturbations, yielding two major types of edges: alleviating GIs and aggravating GIs. In alleviating GIs, also called positive GIs, the organism fitness after the double-knockout perturbation is better than expected based on the single knockout results. In aggravating or negative GIs, the fitness is worse than expected. In gene co-expression networks, nodes represent genes and edges score the correlation in expression between the two genes (10,11). In gene differential correlation (DC) networks, edges score the change in gene pairwise correlation between one set of samples to another (e.g., cases and controls) (12-14). With the growing use and number of types of biological networks, computational methods that exploit these rich data are of great importance.

Computational methods that make use of several networks are often better than methods that analyze only a single network (4,7,8,15-19). For example, combined analysis of PPI networks and gene co-expression networks was utilized to detect gene sets that are co-expressed and are connected in the PPI network. Such analysis outperformed standard clustering algorithms, and was successfully utilized for gene function prediction (5,8,16,19). Alleviating and aggravating GI data were used to find epistasis among and within gene sets. Under the premise that negative GIs tend to occur between compensatory pathways and positive GIs occur within pathways (or complexes), analysis of GIs was used to suggest a map of epistatic relations among functional gene modules (15,17,20-22). A marked improvement was reported after adding a connectivity constraint in a PPI network of the modules (15,17). The ability to construct a summary map of several networks allows identifying associations among discovered modules, thus improving the interpretability of the results compared to standard clustering of a single network.

Building on prior studies of specific pairs of networks, we introduce and study the fundamental problem of constructing a summary map of two biological networks H and G, where the nodes of both are the same genes or proteins, and the edges in each represent a distinct type of relations (see **Figure 1D**). The map nodes are gene sets that are strongly connected in H, and pairs of sets are connected by links. A link represents strong connection between two gene sets in G. The goal is to find gene modules in H simultaneously with finding module-to-module interactions according to G, by optimizing a specific objective function. We call this computational problem the *module map problem.*

Most algorithms for the module map problem to date were used to find a summary map of epistatic interactions among pathways (15,17,20-22). Kelley and Ideker (15) proposed a method that is based on local searches in the graphs to find pairs of connected modules. Ulitsky et al. (17) used a clustering of H as a starting point and then improved the solution by merging modules. An

algorithm akin to (15) has been recently proposed for analyzing gene co-expression and DC networks. The joint analysis of these networks revealed gene groups that are much more (or much less) correlated in one class of individuals (23). Although previous algorithms for the module map problem proved valuable, a thorough analysis of the problem and of the merits and weaknesses of these algorithms in different scenarios is required.

The problem of finding an optimal module map is NP hard under most formulations, as it contains the clustering of H as a subproblem. Hence, heuristics are used. These algorithms usually contain two phases. We call the first phase *initiators*: algorithms for finding an initial solution that may contain many small modules. The second phase employs *improvers*: algorithms for improving an initial solution according to a predefined objective function. A variety of algorithms can be formed from different combinations of initiators and improvers.

Here, we study novel and extant initiators and improvers. We show that a new initiator based on maximal bicliques in G together with a statistically formulated global improver strategy performs consistently better or equal to extant methods on synthetic and real networks of several types. We call the resulting algorithm ModMap. We apply ModMap to experimental data in three biological scenarios: (1) using yeast PPIs and negative GIs, we find epistatic relations among protein complexes, (2) using yeast PPIs and DNA damage-specific positive GIs, we detect emerging connections among protein complexes involved in DNA damage response, and (3) using differential correlation analysis of gene expression profiles of non small cell lung cancer tissues, we identify disease specific loss of correlation between immune activation processes, and detect disease-specific microRNAs.

Materials and Methods

Definition of the module map problem

The input to the problem is a pair of networks $H=(V,E_H,W_H)$ and $G=(V,E_G,W_G)$ defined on the same set of vertices. These networks can be weighted or un-weighted. The goal is to find a module map that summarizes both networks. A *module map* is a graph F=(M,L) where M is a collection of disjoint node sets, called *modules*, $M=\{M_1,\ldots, M_p\}$, $M_i \subseteq V$, $M_i \cap M_j = \emptyset$, and L is a set of module pairs $\{(U_1,V_1), \ldots, (U_p,V_p)\}$, where each U_i and V_i are in M. These pairs are called the map *links*. In addition, each module must be linked to at least one other module. Roughly speaking, our goal is to find a module map such that each module corresponds to a heavy sub-graph of H, and each link represents a heavy bipartite sub-graph in G between a pair of modules. (A formal notion of heavy subgraphs will be introduced later.) **Figure 1D** shows a toy example of two unweighted networks and their module map.

Previous algorithms for constructing module maps vary in the way they define the objective function and the links. The DICER algorithm (23) seeks one pair of linked modules at a time. A pair of modules is defined as linked if the sum of weights W_G between them is high enough. We call the approach of DICER *local*, as it finds one module pair at a time. The algorithm of Ulitsky et al. (17) aims to maximize the *global score*, namely, the total sum of scores within modules in H plus the sum of scores of links in G. In addition to increasing the global score, links between

modules are accepted only if they pass a statistical significance test. We call the second approach *global*. Note that both methods identify the links and the modules simultaneously.

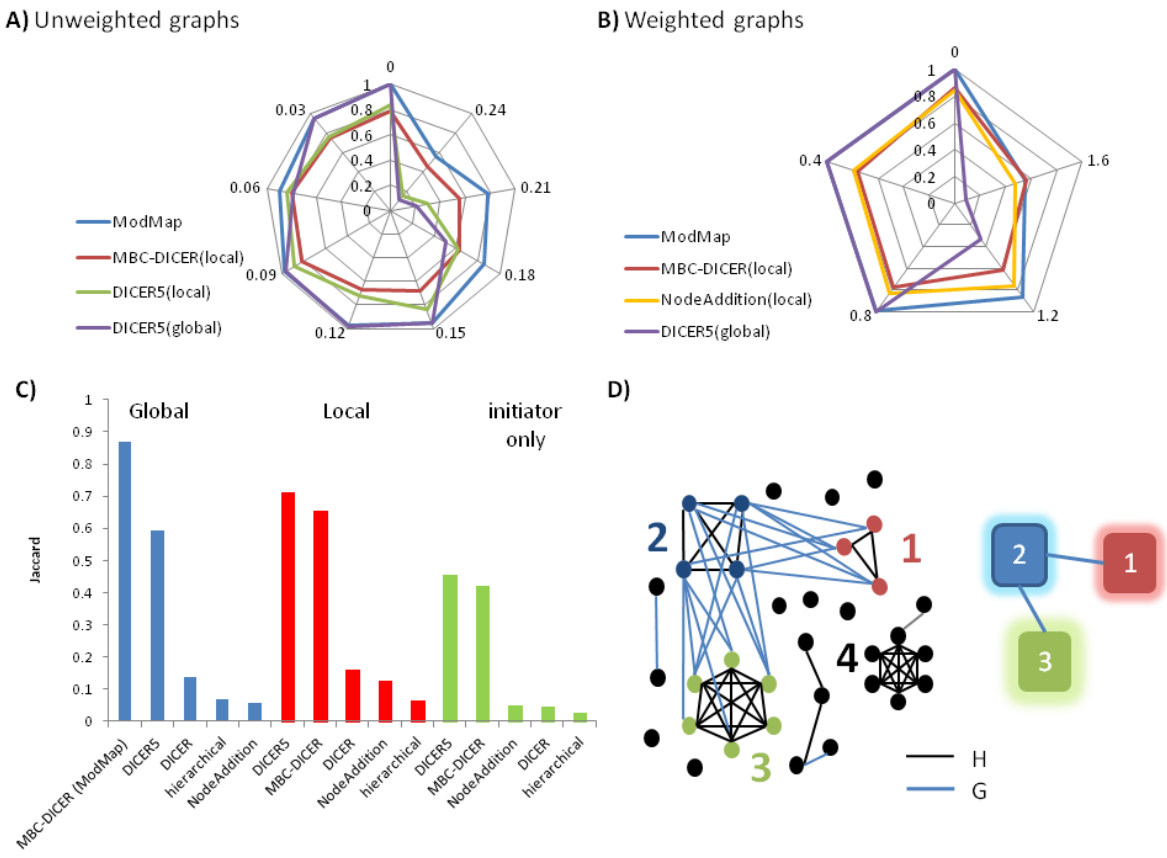

Figure 1 Module map: example and simulation results. A-B) Performance of module map algorithms on 500-node graphs. A) Unweighted graphs. B) Weighted graphs. Each simulated pair of graphs contained an embedded module map of six modules in a tree structure. In addition, two random cliques and two bicliques were embedded in the graphs as decoys. Module, clique, and biclique size was chosen uniformly at random between 10 and 20. In the un-weighted model (A) each edge was replaced by a non-edge with probability p, and vice versa. In the weighted model (B) edge weights are sampled from the normal distribution N(1,σ), and non-edge weights are sampled from the normal distribution N(-1, σ). Results are averages of 10 simulations for each data point. The four top performing algorithms for each simulation are presented using radar plots. MBC-DICER with global improvement is denoted as ModMap. The Jaccard coefficient between the modules produced by each algorithm and the true modules is shown as the distance from the center. Consecutive spokes from the top anticlockwise show increasing values of p in A and of σ in B. C) Comparison of module map algorithms on unweighted graphs with 1000 nodes, containing a map of 10 modules and five decoys and p=0.15. D) A toy example of the module map problem; left: the two networks. Nodes are genes, H edges are black and G edges are blue; right: The module map. Nodes are modules and edges are links. Colors and numbers are the same on the left and right. The map contains three modules: Module 2 is linked to modules 1 and 3, whereas module 1 and 3 are not linked. Black nodes are not part of the module map. Note that the graph H (black edges) contains a clique that is not linked in G to another module and thus is not a part of the map. The example also demonstrates the difference between the local and global approaches. The local approach identifies modules 1 and 2 as linked, while the global approach also identifies module 3 as linked to module 2. See text.

Figure 1D demonstrates the differences between the local and global approaches. Assume that in both graphs edge weights are 1, non-edge weights are -1, and that the local approach uses a threshold of 0 on the sum of W_G weights between two modules for reporting a link. In both approaches modules are clusters of nodes with high density in H. According to both approaches, module 1 is linked to module 2: the local score is 4 (8 edges and 4 non-edges), the global analysis p-value for linkage is lower than 0.05, and the total score for the module pair is 13 (module score 6+3 + link score 4). The sum of W_G weight between modules 2 and 3 is -4 (10 edges and 14 non-edges) and the local method rejects that link. However, the global approach will also link module 2 and 3: the linkage p-value is significant (p=0.039), and adding this link will improve the global map score to 24 (13 for the (1,2) pair + 15 for module 3 – 4 for the (2,3) link). This example illustrates the advantage of the global approach on sparse graphs, in which large modules are not expected to be densely inter-connected.

Algorithms

We conducted a systematic study and developed further a family of two-phase algorithms for module map detection that find an initial solution (possibly consisting of many small modules), and then improve it. We call algorithms for the first phase *initiators* and algorithms for the second phase *improvers*. For simplicity, we describe the algorithms assuming that edges with positive weight are considered heavy. For un-weighted graphs we assume edge weights to be 1 and non-edge weights to be -1. For weighted graphs all node pairs (edges) have weights, so there are no non-edges.

Initiators

We tested five different initiators: (1) DICER (23), which finds one pair of linked modules at a time, (2) hierarchical clustering of the graph H (26), which finds a set of modules, (3) a greedy node addition algorithm for finding modules in H, (4) $DICER_k$ a variant of DICER wherein the minimum module size is set to k, and (5) an algorithm based on enumeration of maximal bicliques in G using an exhaustive solver (24,25), followed by the cleaning process of DICER. We call the latter algorithm MBC-DICER, see **Supplementary Text** and **Supplementary Figure 1** for a full description of all initiators. Each initiator creates an initial module set, but note that modules in the map constructed by clustering algorithms are not necessarily linked.

Improvers

The local improver (23) extends module map links by either adding a single node to a module or by merging two module map links. One drawback of this approach is that it cannot create new modules that are not represented in the initial solution. Another disadvantage is that it cannot merge a module whose two parts are linked to different modules that are unlinked. See **Supplementary Figure 2** for examples. Below we introduce the global improver, which can often overcome both problems.

Our **global improver** is based on the procedure in (17). Let $M=\{M_1,\ldots, M_n\}$ be a collection of disjoint node sets (e.g., a set can be a single gene or not linked to any other set). Given sets (U,V), $U,V \in M$, and $x \in U$, the significance of the linkage of x with V is calculated using Wilcoxon

rank-sum test by comparing the edge weights W_G between x and V to the edge weights between x and all nodes not in V. Such p-values are calculated for all nodes in U and V, and they are combined using Stoufer's method (27). If the final p-value p(U,V) is at most α then U and V are connected by a *link* in the map. Let L={ (U_1,V_1), …, (U_p,V_p)} be the resulting set of links.

The *global score* of the solution is the sum W_H of edge weights within each M_i plus the sum of W_G edge weights between the linked node sets:

$$S(M,L) = \sum_{i|M_i \in M} \sum_{s,t \in M_i} W_H(s,t) + \sum_{k,l \mid (M_k,M_l) \in L} \sum_{i \in M_k, j \in M_l} W_G(i,j)$$

$$s.t. \forall_{(M_k,M_l \mid k \neq l, M_k, \in M, M_l \in M)} \ (M_k,M_l) \in L \ \Leftrightarrow \ p(M_k,M_l) \leq \alpha$$

The improvement stage merges a pair of node sets (two modules or a module and a single gene) if the global score increases and the new link passes the significance test. Note that considering a merge that creates a new module Y requires recalculating p(Y,Z) for all other modules Z in M, in order to calculate the global score. This process is done greedily: iteratively, the merge that yields the best improvement is performed until no possible merge can improve the global score.

We modified the above method to allow for fast analysis of large graphs, as follows. First, when calculating p(U,V) we consider the links in G' (the un-weighted version of G). We use a hyper-geometric test to evaluate if a node has significant number of edges in G' to the opposite set (e.g., from a node v ∈ V to the set U), and then all node p-values are merged using Fisher's method (28). (U,V) are linked if the resulting value <α. This test is much faster and provides maps of equal quality to using the Wilcoxon test on G (see **Supplementary Text**). Note that weighted tests, such as the Wilcoxon test, are not always appropriate for detecting linkage among gene modules. For example, in the differential correlation graphs, a strong link must contain many positive edges, while the Wilcoxon test only looks at the ranks of the edge scores.

Second, we set another parameter β>>α, and if at some point the p-value for the possible link between two sets is at least β, we say that the sets are *anti-linked*. In the original algorithm, when considering merging two sets U and V into W, possible links between W and every other set Y must be calculated. However, if U and Y are anti-linked or V and Y are anti-linked then we mark W and Y as anti-linked, avoiding the need to consider the possible link (W,Y). In practice we used α=0.005 for the yeast data as suggested in (17), and tested several options in the gene expression data (see **Supplementary Text**). In all cases we used β=0.2. Finally, we perform multiple merge steps simultaneously in a single iteration in a way that guarantees that the global score improves (see **Supplementary Text)**. This provides a speed up of two-fold or more in practice without loss of solution quality.

Simulations

We constructed initially empty 500-node graphs H and G and then added edges creating a perfect module map in which modules are cliques in H and links are bicliques in G. The module map topology (M,L) was a random tree with |M|=6. We then added two H-cliques and two G-bicliques to the graphs to represent additional "decoy" structures that are not part of the map. Clique,

biclique, and module sizes were randomly selected in the range 10-20 with uniform distribution and disjoint node sets. Call the resulting edge sets E_H* and E_G*. Finally, we modified these graphs by introducing random noise: each edge in G and H was deleted with probability p, and each non-edge was replaced by an edge with probability p. All reversal steps were done independently. For creating weighted graphs, the same procedure was used, but all possible edges are present in the final H and G: w(u,v) is sampled from $N(1,\sigma)$ if (u,v) is in E_H* or E_G*, and from $N(-1,\sigma)$ otherwise. We also generated in this manner 1000-node graphs with 10 or 20 modules and five decoys (cliques and bicliques).

Analysis of negative genetic interactions and protein-protein interactions in yeast

Protein-protein interactions (PPIs) and negative genetic interactions (GIs) were downloaded from BIOGRID (29). These networks were used to find epistatic relations among protein complexes. The PPI network was used as H and the GI network was used as G (See **Supplementary Table 1**).

Analysis of DNA damage-specific genetic interactions data

We used the data of (21), in which all pairwise GIs among 418 genes were tested, and of (30), which tested GIs between 55 query genes and 2,022 genes. A *DNA damage-specific positive GI* was defined as one that had $S<0$ in the untreated cells, $S>0.5$ in the treated cells, and the p-value for differential GI was <0.01. This analysis yielded 840 interactions from (21) and 1677 interactions from (30). We additionally defined a positive GI as *stable* if it had $S>1.5$ both in the untreated cells and in the DNA damage cells. This analysis provided 491 interactions in (21) and 3139 interactions in (30). Note that due to the different experimental setups most of these GIs are not directly comparable.

Calculating differential correlation scores

Given a training set containing gene expression profiles of subjects, we used the statistical method of (23) to compute for each gene pair its consistent correlation and differential correlation (DC) scores. First, DC scores are computed using the real labels of the samples. Then, the scores are transformed to log-likelihood ratio (LLR) scores by comparing the original DC scores to scores calculated on the same data with randomly shuffled labels. Thus, positive LLR scores mark gene pairs with significant change in DC. The prior probability of real DC changes was set so that only correlation changes of at least 0.4 will have a positive LLR score. This approach guarantees a similar yet slightly more stringent acceptance threshold compared to (23). See **Supplementary Text** for additional information.

GO and microRNA enrichment analysis

We used TANGO (31) for GO molecular function and biological process enrichment analysis of modules, and FAME (32) for microRNA enrichment analysis. Both tools are available as part of the EXPANDER software (33). When a set of modules was analyzed, we corrected for multiple testing using FDR with $q=0.05$. Notably, the background set for the enrichment analysis was defined as the set of genes in the networks and not all genes in the organism. This filtering step reduces bias in case of over-representation of GO terms in the networks.

Network visualization

Network visualization was done using Cytoscape (34).

Availability

A command line tool for running ModMap is freely available for academic use at http://acgt.cs.tau.ac.il/modmap/.

Results

Simulations

We first tested the different algorithms on synthetic graphs H and G. Starting from a perfect module map, we first added cliques in H and bicliques in G to represent additional structures that are not part of the map, and then introduced random noise to the edges. To generate both sparse and denser graphs, we tested a wide range of the noise parameters σ and p in the weighted and the unweighted simulations, respectively (see Materials and Methods). The results presented here are for graphs with 500 nodes and six modules per map. We also tested larger graphs with similar results (See **Supplementary Figures 3, and 4**).

We tested 10 combinations of initiator and improver on 10 random datasets for each value of p and σ. We measured the quality of produced solutions using Jaccard coefficient between the reported modules and the known modules. The results of the un-weighted and weighted models are shown in **Figure 1A** and **Figure 1B,** respectively. Only the four algorithms that performed best on average in each simulation are shown. **Supplementary Table 2** contains the results for all combinations. The local improvement algorithms did not reach perfect scores even on noiseless data. In contrast, MBC-DICER and $DICER_5$ followed by global improver reached perfect Jaccard scores when there was no statistical noise. The high performance of MBC-DICER remained robust even when noise levels were as high as p=0.15 in the un-weighted model and σ=1.2 in the weighted model. A comparison of all algorithms on unweighted graphs with 1000 nodes and 10 modules for noise level p=0.15 is shown in **Figure 1C.** Performance remains high although the graphs are much larger. Using the improvers was beneficial compared to using only the initiator solutions, especially for the DICER variants. MBC-DICER with the global improver reached highest performance (0.87). Interestingly, the local improver was better than the global improver for all other algorithms (e.g., 0.71 vs. 0.59 for $DICER_5$). This is probably because the MBC-DICER initiator detects robust fully-connected modules, which are a better starting point to the global improver at high noise levels. Tests with different values of k for the $DICER_k$ algorithm led us to choosing k=5 (**Supplementary Figure 4**). In addition, we compared the performance of the global improver with the hyper-geometric test and with the Wilcoxon rank-sum test, which was used in previous studies. Our results show that using the hyper-geometric test reaches similar quality of results, but is much faster (see **Supplementary Text**). Overall, the results indicate that MBC-DICER followed by the global improver achieved the best performance on both un-weighted and weighted data. We call the resulting algorithm **ModMap** and will use it as the algorithm of choice from now on.

Yeast protein-protein interaction and negative genetic interaction data

We used protein-protein interactions (PPIs) and negative genetic interactions (GIs) from BIOGRID (29) to find epistatic relations among protein complexes. Only genes that had both types of interactions were used. Overall, the networks contained 3979 genes, 45,456 PPIs, and 76,237 negative GIs (the interactions are listed in **Supplementary Table 1**). Note that this number of genes and edges is larger than in previous studies. For example, (22) covered 1460 genes, and (17) covered 743 genes. Therefore, our networks have the potential to provide a broader overview of the yeast interactome and allow for a comprehensive performance testing of the different algorithms.

As done in previous studies, we evaluated solutions by their statistics and the functional characterization of the modules (17,22). The calculated solution statistics included the number of modules, the number of genes covered, and the maximal module size. We used TANGO (33) to measure module functional enrichment, and reported the number of discovered GO terms, the percent of enriched modules, and the percent of module map links for which both modules are enriched (with the same or with different functions), which we call *enriched links*. Enriched links represent dense GIs among known biological terms.

The solution statistics of all algorithms are shown in **Supplementary Table 3.** One can observe clear superiority of global over local improvers. While global improvers reported at least 100 modules and covered 800-1000 genes, the local improvers found 2-28 modules covering only 15-192 genes. Except for DICER, the results of all solutions were similar and of high quality. ModMap was the best in terms of the percent of enriched modules (87%) and percent of enriched links (80%). Taken together, the map of ModMap was best in combining functional comprehensiveness and quality. We also compared ModMap to other weighted approaches for GI data analysis (22,35) on the data of Collins et al (36). See **Supplementary Text** for details. Our results show that ModMap produces high quality maps and improves upon extant weighted approaches.

Figure 2 shows a portion of the map constructed by ModMap where links were restricted to p<10E-50 (for details see **Supplementary Tables 4-6**). Each node represents a module and edges represent map links. All modules in the presented map are enriched at 0.05 FDR with at least one GO term. The node labels show the most significantly enriched term. Three major hubs are marked in green: Rpd3L complex (14 genes, p=4.35E-38), Swr1 complex (13 genes, p=1.08E-35), and the mediator complex (17 genes, p=4.89E-43). The Rpd3L and Swr1 complexes are chromatin related and were previously annotated as hubs of GIs in a gene based study (37). Bandyopadhyay et al. (21) discovered some of the same links; however, module annotation there was manual, whereas our analysis was completely automatic, and produced a much larger map. Moreover, our map extends upon the previous observations by showing that the three hubs are linked and by providing additional links for the Rpd3L complex. In **Figure 3** we focus on the three most significant links in the map (p<1E-70). **Figure 3A** shows the connections between the Rpd3L and Set3 complexes, and between the Rpd3L and Swr1 complexes. Rpd3L and Set3 are both histone deacetilases and negative GI between them was reported in (20). Notably, the Rpd3L complex was split into two disjoint modules, while in our map it is detected as a single module,

containing all 14 Rpd3L genes. **Figure 3B** shows a connection between two well established subunits of the proteasome complex (38). This example shows how joint analysis of PPIs and GIs correctly detects core functional subunits even when they are connected by many PPIs.

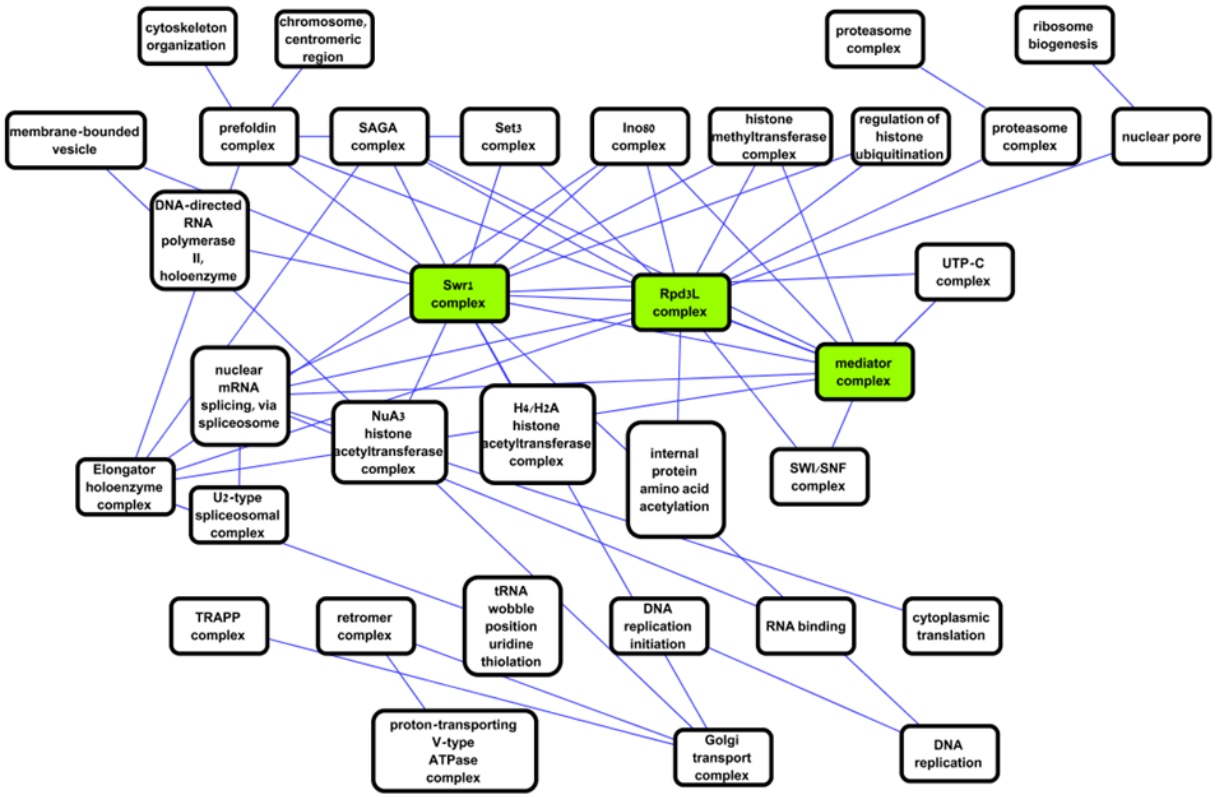

Figure 2 The yeast module map. Each node is a module in the yeast PPI network. The name of a node is the most significantly enriched GO term for that module. Each edge represents a highly significant link between two modules in the negative GI network ($p<1E-50$). Modules that were not enriched for any GO term at 0.05 FDR are not shown. Three main chromatin related hubs are marked in green. Some links connect disjoint modules enriched with similar GO terms (e.g., proteasome-proteasome link, top right), and other links show epistasis between different biological processes (e.g., nuclear pore and ribosome biogenesis, top right).

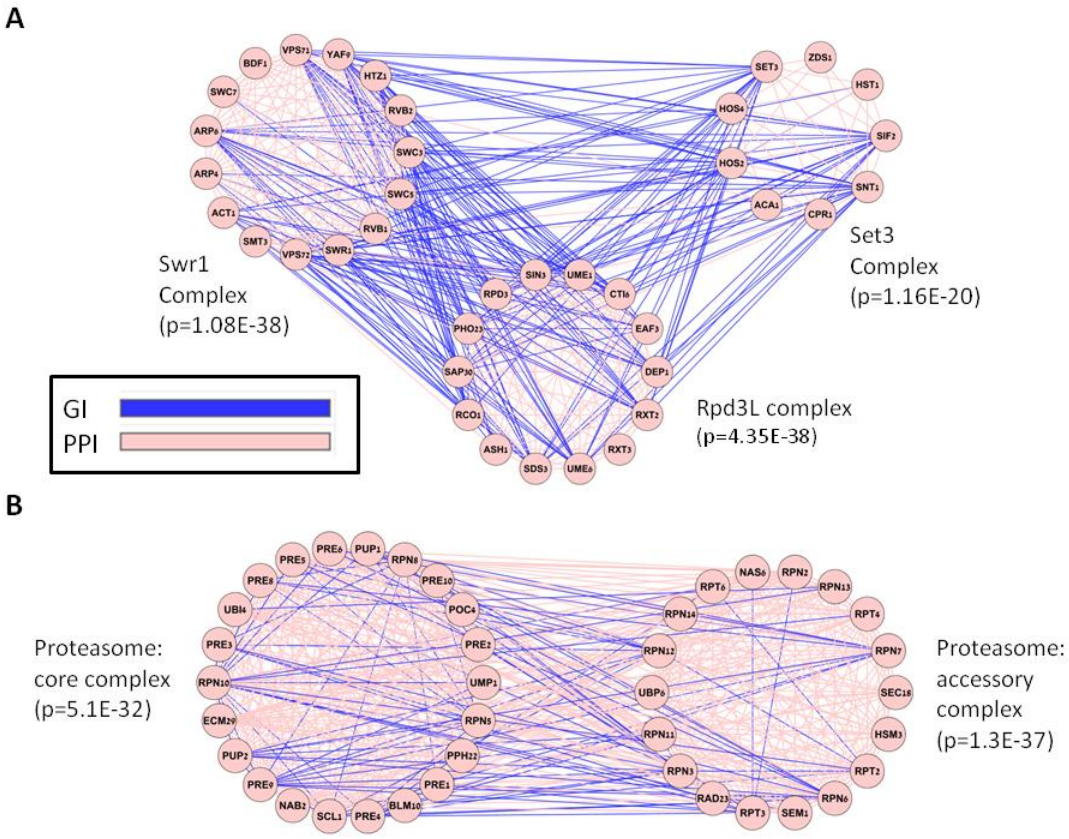

Figure 3 Examples of linked modules in the yeast module map. The genes of each module are arranged in a circle. Blue edges represent negative GIs and pink edges represent PPIs. For each module the most enriched GO term is shown along with its enrichment p-value. A) Linkage among different protein complexes. The significance of the links between Rpd3L and the Set3 complexes, and between Swr1 and Rpd3L complexes is < 10E-70. The link between Swr1 and Set3 is also highly significant (p=4.29E-59). B) Detection of sub-complexes. The joint analysis of the PPI and GI networks partitions the proteasome complex into its two sub-complexes: the accessory and the core complex.

Analysis of DNA damage response networks in yeast

The module map described above was obtained by analyzing the entire set of known negative GIs. Recent studies have gone beyond static analysis to detect changes in the GI network in response to DNA damage (21,30). In these studies, GIs were measured in untreated cells and following perturbation by the DNA-damaging agent methyl methanesulfonate (MMS) (41). We combined two such datasets (21,30) to detect *DNA damage-specific positive GIs*, i.e., differential positive GIs that emerge in the treated cells and are not observed in the untreated cells (see Materials and Methods). Negative GIs are typically observed between genes working in parallel, such as genes that are involved in two compensatory complexes or pathways that backup each other, and thus the loss of one is buffered by the other. Positive GIs are more likely to be observed between genes from the same complex or pathway, where most of the phenotypic effect is already observed in each single knockout. Hence, DNA damage-specific positive GIs are expected to represent changes of the network in response to MMS, revealing DNA damage

specific interactions within pathways or between different pathways or complexes working in series. In total, 1078 genes were included in both studies, with 2227 DNA damage-specific positive GIs among them (See **Supplementary Table 7**). There were 6771 PPIs within that gene set.

We applied ModMap with the PPI network as H and the DNA damage-specific positive GI network as G. Since these networks were much smaller than in the previous analysis, we set the minimal module size to 3. The small module sizes also affected the attainable p-values for links. Here, a pair of modules was defined as linked if its p-value was < 0.05 after Bonferonni correction, considering all statistical tests done by the algorithm during the improvement steps.

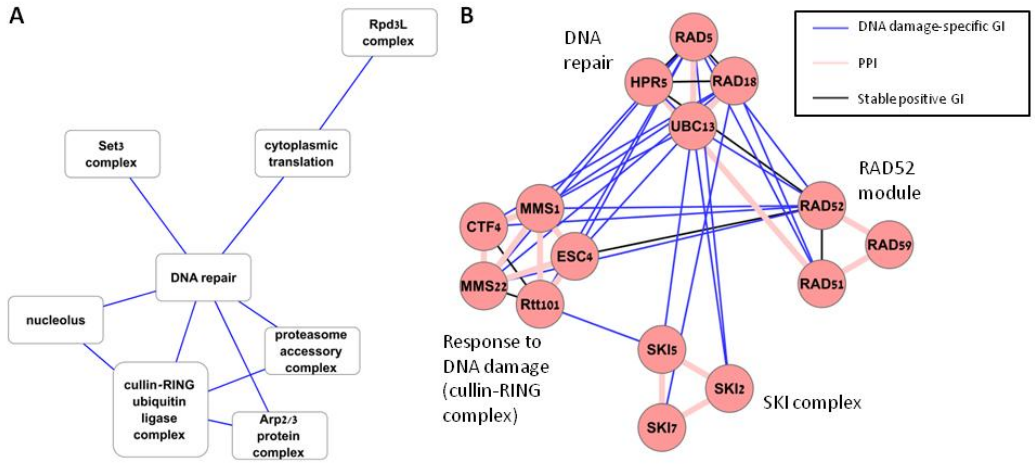

Figure 4 A module map of DNA damage-specific positive GIs. A) A module map of the significantly enriched modules. Nodes represent modules and edges represent significant links (Bonferonni corrected p < 0.05). The name of a node is the most significantly enriched GO term. B) A closer look at the DNA repair module and three linked modules. Nodes represent genes and edges represent interactions: blue – DNA damage-specific positive GIs; pink – PPIs; black – stable positive GIs, which are observed both in the untreated and in the treated cells. This map shows the emerging connections between functional modules upon DNA damage response covering DNA repair and checkpoint responses in the DNA repair module, response to damaged replication forks (the DNA damage response module), DNA double stranded response genes (RAD52 module) and RNA degradation related genes (SKI complex module). Note that the RAD52 and SKI modules do not appear in A since they reflect functions that do not have established GO terms.

The generated module map contained 78 genes in 12 modules, with 17 links among them. Module sizes ranged between 3 and 15. A complete description of the map is provided in **Supplementary Tables 8-10**. A map of the modules that were significantly enriched with GO terms is shown in **Figure 4A**. The hub in this map is a module enriched with DNA repair genes, linked to six modules that cover a large variety of functions. In **Figure 4B** we focus on the DNA repair related module and on three of the modules linked to it. The DNA repair module contains four genes: RAD5, RAD18, HPR5, and UBC13. Interestingly, while UBC13 is known to physically interact with the three other genes, positive GIs that are consistently stable across experiments (see Materials and Methods) connect the other three genes, providing further evidence that the four genes are involved in a common process. The RAD5, RAD18, and UBC13

genes are known to be involved in postreplication repair (42-44) and HPR5 is involved in checkpoint recovery (45,46).

The DNA repair hub module is linked to a module associated with response to DNA damage. It contains five genes: CTF4, ESC4, MMS1, MMS22, and Rt101. The last four genes are part of the cullin-RING ubiquitin ligase complex (GO:0031461). The last three genes were shown to form a complex that stabilizes the replisome during replication stress (47,48). The CTF4 gene is related to DNA repair and DNA replication initiation according to its GO annotations. The link suggests that this complex might work together with the DNA repair module for coping with damaged replication forks. Interestingly, the two MMS genes were originally detected in MMS sensitivity tests but are not expected to be required for double-stranded repair (48). The RAD52 module (RAD51, RAD52, and RAD59), is related to double-stranded DNA damage repair (49), and is linked both to the DNA damage repair module and to the DNA damage response module, suggesting these modules work together in the same pathway as a result of DNA damage to cope both with damaged replication forks and with double-stranded DNA breaks. The fourth linked module contains three genes of the Ski complex (SKI2, SKI5, and SKI7). These genes are involved in 3-5 RNA degradation in the cytoplasmatic exosome (50,51). Our analysis suggests that this complex might also be involved in response to DNA damage. Previous studies have shown that RNA degradation cytoplasmatic genes might play a role in DNA damage response separately from their cytoplasmatic activity (52,53). The suggested roles of RNA degradation genes in DNA damage response include DNA stability and telomere stability related functionality (52), mediating the assembly of multi-protein complexes in double-stranded breaks (53), and specific mRNA degradation upon DNA damage (54). Hence, our findings match prior studies, and strengthen the role of the SKI complex in the response to DNA damage.

Analysis of human co-expression and differential correlation networks

We applied ModMap on case-control gene expression data of non-small cell lung cancer (NSCLC) to reveal differential correlation among highly correlated gene modules. The contribution of this part is twofold. First, we show that differential correlation among gene modules is reproducible in cross-validation tests. Second, we analyze the map of DC patterns between gene modules discovered by ModMap.

Given a dataset of gene expression profiles from cases and controls, we used the method of (23) to compute two scores for each gene pair: the consistent correlation (CC) score, which is positive if the gene pair is consistently correlated across phenotypes, and the differential correlation (DC) score, which is positive if the correlation difference between the cases and controls is higher than expected by chance. These scores were then used as edge weights in networks H and G, respectively, on which a module map was learned. The methodology was evaluated using cross validation: Given a module map constructed on a set of profiles (the *training set*) and a disjoint set of samples (the *test set*), the quality of the predicted map was evaluated on the test set by comparing the DC of links and of non-links using Wilcoxon rank-sum test, where the null hypothesis is that there is no difference in DC between links and non-links. This measure is parameter-free and reflects all DC changes.

We tested several variants of the algorithm using 2-fold cross-validation. The maps produced by the local improver received low p-values, but suffered from low coverage. For example, for the MBC-DICER initiator, the local improver achieved a p-value of 4.43E-4, but the map covered only 197 genes. In contrast, when applying ModMap (i.e., MBC-DICER with the global improver), the map covered 1289 genes, with p-value of 1.54E-10. **Supplementary Text** contains further results of testing different parameters of the global improver, as well as tests on Alzheimer's disease (55), which got similar cross validation results. The full results are shown in **Supplementary Table 11** for lung cancer and in **Supplementary Table 12** for Alzehimer's disease. Taken together, ModMap produces large maps that are robust when tested on independent datasets.

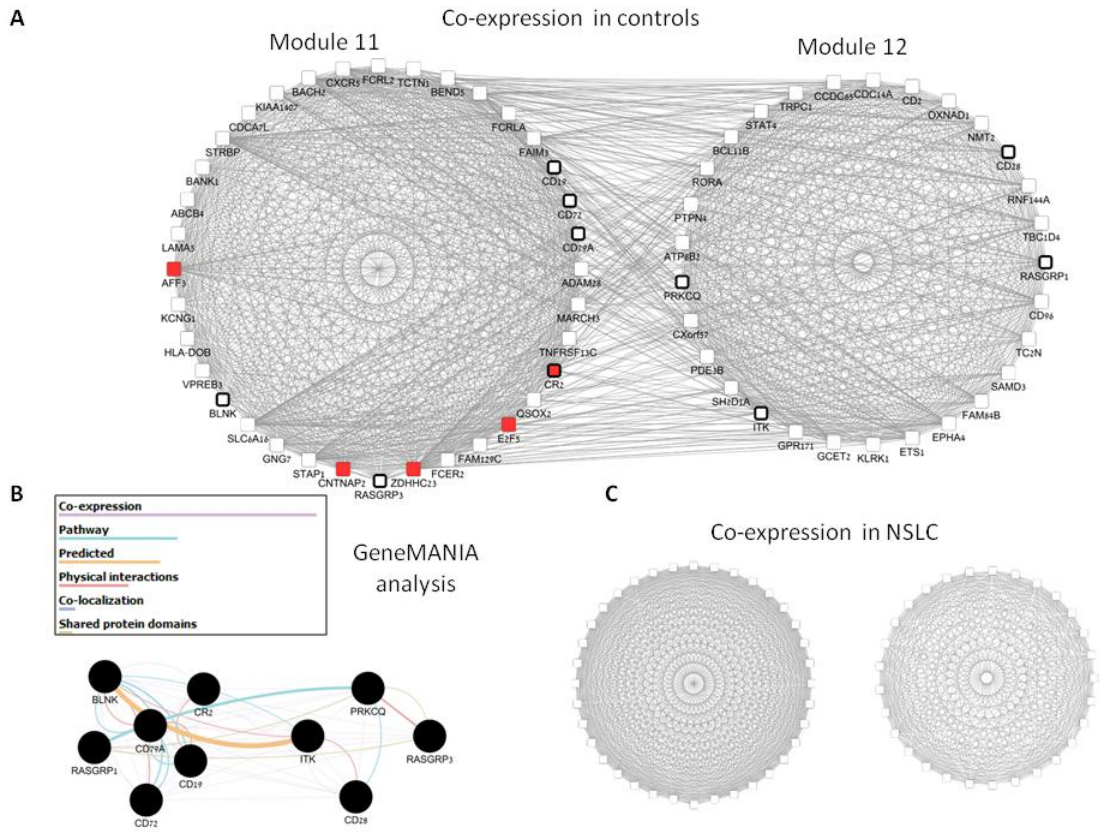

Figure 5 A pair of immune activation related modules differentially correlated in NSCLC. A) Two linked modules, which are a part of the constructed module map. Nodes are genes and edges represent correlation > 0.4 between the genes in the expression patterns of control class. Note that edges here correspond to high co-expression between two genes and do not reflect the weights in the CC or DC networks. We observe strong co-expression both within and between the modules. Nodes with black frames are related to immune activation response (six T-cell activation genes in module 11, and four B-cell activation genes in module 12). Red nodes in module 11 are targets of mir-34 family. B) GeneMANIA analysis of the T-cell and B-cell signaling pathway genes shows that the genes of both modules are expected to interact in healthy controls. C) The same two modules and their co-expression network in the NSCLC class. As in A, the genes within each module are highly co-expressed. In contrast to A, co-expression between the modules is completely diminished.

Next, we analyzed the module map obtained by running ModMap on all samples of the NSCLC data. The map covered 1921 genes in 76 modules, connected by 405 links (See **Supplementary Tables 13-14** for details). To focus on strong changes in correlation between modules we compared the DC of each link in the map to the DC calculated between random gene sets of the same sizes in 200 repeats, and calculated the fold-change between the real link and the best random link, as proposed in (23). The link fold change scores are given in **Supplementary Table 14**. 150 links had fold change ≥ 1.5, with the top five links exceeding 2.3. This indicates that the DC of the linked modules is far stronger than expected by chance. We also analyzed the modules of the top links using pathway enrichment analysis and miRNA enrichment analysis (See **Supplementary Table 15** for details). One of the links connected two modules related to immune response activation. The linked modules are shown in **Figure 5**. In **Figure 5A** we observe many high co-expression edges between the modules (gene pairs with $r>0.4$) in the control class. Module 11 is enriched with B cell receptor signaling pathway genes (6 genes, p=3.1E-8). Module 12 is enriched with T cell receptor signaling pathway genes (4 genes, p=1.37E-4). **Figure 5B** shows GeneMANIA analysis of these 10 genes (7,56), which confirms that they are connected by several types of interactions. **Figure 5C** shows the co-expression of the same modules in the NSCLC class. Within each of the modules a strong level of co-expression is preserved, but the co-expression between the modules is abolished, suggesting that co-regulation of the different immune responses is lost in NSCLC. Finally, module 11 is highly enriched with targets of miRNA 34-a,b,c family (red nodes in **Figure 5A**) whose members are annotated as causal to NSCLC according to the mir-2-disease database (57). Taken together, these results show the ability of our analysis to detect NSCLC related functional modules without using any prior knowledge.

Discussion

In this paper we presented a methodology for joint analysis of two gene networks, each representing a different type of omic relation between genes. The method identifies gene sets as modules and the complex structure of relations among them, and summarizes the analysis in a module map. Modules correspond to interacting gene sets in the first network, and links in the module map correspond to interacting modules in the second. The map is constructed based on both networks simultaneously, and thus can capture and reveal structures that are not identifiable when analyzing each data type separately. Our novel algorithms recovered the planted map structure in simulated data, even when the noise level in the data was very high. We tested our methods in three biological applications: (1) yeast PPIs and negative GIs, (2) yeast PPIs and DNA damage-specific positive GIs, and (3) differential correlation analysis of human disease expression profiles. In all cases, certain parts of our maps are supported by prior biological knowledge, while other parts reveal novel structure and suggest new biological findings. The module map paradigm can be applied in principle on any two types of networks with underlying common nodes.

Our analysis of the yeast PPI and negative GI data constructed a large map describing epistatic relations among complexes. Our findings are in agreement with previous studies and show a complex map of interactions among chromatin modification-related complexes, but also provide interactions with other functions, such as protein modification-related complexes. The analysis of

the yeast PPIs and DNA damage-specific positive GIs produced a smaller map, which contains a DNA repair module as a central hub. The interactions of this module suggest that several mechanisms emerge simultaneously in response to MMS, including double strand repair, damaged replication fork repair, and exosome complex activity. In the map constructed based on human NSCLC blood expression profiles, modules represent gene sets that are highly co-expressed both in cases and in healthy controls, whereas the map links correspond to specific rewiring of the co-expression network in NSCLC patients. In particular, we identified two modules enriched with immune activation genes manifesting a sharp drop in correlation in the NSCLC patients, suggesting diminished coordination between the T-cell and the B-cell enriched modules.

The concept of a module map can be viewed as a higher level combination of clustering and biclustering. Each of those problems has been extensively studied and was applied successfully to numerous single-type genomic and proteomic studies (1,58-69). By performing joint analysis on two different data types we allow some relaxation of the objective function in each of the networks, for the sake of obtaining an overall clearer structure. Therefore, the new analysis can yield results when clustering or biclustering of one data type fails. One of the difficulties in clustering and biclustering is that module (or module-pair) sizes must be large enough to obtain highly significant sets. As our analysis demonstrates, the added power of the module map approach can identify relatively small, precise groups that are beyond the detection ability of those prior methods.

Only a handful of studies have addressed the module map problem to date, and most of them focused on joint analysis of yeast PPI and GI networks. Ulitksy et al. (17) and Bandyopadhyay et al. (39) developed clustering methods that seek a map in which the likelihoods of the edge weights of PPIs and GIs within clusters or of GIs between linked clusters are higher than a given background distribution. Leiserson et al. (22,35) sought local maximum cuts in the weighted graph of the GIs by a greedy incremental approach, producing a collection of linked pairs of modules. Kelley and Ideker (20) developed a clustering algorithm that is based on graph compression, where the original GI graph is compressed to a module map. Hence, both (22,35) and (20) look for approximate bicliques that connect gene modules. In contrast, we enumerate the maximal bicliques of GIs, analyze them by taking into consideration the two interaction types to ensure that the initial solution contains dense, strongly connected modules, and improve the solution using our global improver. Since our approach is generic, it does not exploit the specific probabilistic nature of the GI data as other methods do (22,35). Nevertheless, we show that our method outperforms these and other extant methods in several criteria on GI data. In addition, since our algorithm is not limited by the type of the input data, we are able to combine many heterogeneous datasets (e.g., using all GIs of BioGRID) in our analysis.

When dissecting human expression profiles of disease patients and healthy controls, differential correlation analysis was proposed as a way to discover gene modules whose inter-module correlation levels are altered in disease (12,14,23,70). We previously developed DICER (23), which uses a local approach to detect module pairs. Here we go beyond it by finding maximal bicliques in the DC graph, and by concurrently constructing a global map of modules. As we showed here, in most cases the map links are highly significant. However, we also observed cases

where the absolute correlation change of modules might be mild even though the DC of the module-pair is significant. A possible remedy is to give more emphasis to high absolute DC of map links in order to see the DC signal better. Another possible improvement is to enumerate bicliques using established heuristics (e.g. (69)).

A key factor in the performance of the ModMap algorithm is the objective function optimized. Here, we chose to maximize the sum of weights within modules plus the sum of weights of module links, and assigned these weights based on a probabilistic model. On un-weighted networks, such as the PPI and GI yeast networks, we set the weight of an edge to 1 and the weight of a non-edge to -1, thereby promoting strongly connected modules and links. This setting produced good results and revealed functional interactions among protein complexes. By setting different weights to non-edges in the graphs, future analyses can promote modules that are sparser, thus enabling better detection of interactions among complete pathways.

Author contribution

Conceived the method and designed the experiments: DA RS. Designed and wrote the software used in analysis: DA. Performed the experiments: DA. Analyzed the data: DA RS. Wrote the paper: DA RS.

Funding

This study was supported in part by the Israel Science Foundation (grants 802/08 and 317/13), the Israel Cancer Research Fund and by a grant from the Lee Perlstein Kagan Charitable Trust. DA is grateful to the Azrieli Foundation for the award of an Azrieli Fellowship. DA was also supported in part by fellowships from the Edmond J. Safra Center for Bioinformatics at Tel Aviv University, and the Israeli Center of Research Excellence (I-CORE), Gene Regulation in Complex Human Disease, Center No 41/11. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Acknowledgements

We thank Yaron Orenstein for his comments on the manuscript.

Conflict of Interests

The authors declare that they have no conflict of interest.

References

1. Enright, A.J., Van Dongen, S. and Ouzounis, C.A. (2002) An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Research*, **30**, 1575-1584.
2. Deng, M.H., Zhang, K., Mehta, S., Chen, T. and Sun, F.Z. (2003) Prediction of protein function using protein-protein interaction data. *J Comput Biol*, **10**, 947-960.
3. Kharchenko, P., Chen, L., Freund, Y., Vitkup, D. and Church, G.M. (2006) Identifying metabolic enzymes with multiple types of association evidence. *BMC Bioinformatics*, **7**, 177.

4. Pandey, G., Myers, C.L. and Kumar, V. (2009) Incorporating functional inter-relationships into protein function prediction algorithms. *BMC Bioinformatics*, **10**, 142.
5. Sharan, R., Ulitsky, I. and Shamir, R. (2007) Network-based prediction of protein function. *Molecular Systems Biology*, **3**.
6. Kourmpetis, Y.A.I., van Dijk, A.D.J., Bink, M.C.A.M., van Ham, R.C.H.J. and ter Braak, C.J.F. (2010) Bayesian Markov Random Field Analysis for Protein Function Prediction Based on Network Data. *PLoS One*, **5**.
7. Warde-Farley, D., Donaldson, S.L., Comes, O., Zuberi, K., Badrawi, R., Chao, P., Franz, M., Grouios, C., Kazi, F., Lopes, C.T. *et al.* (2010) The GeneMANIA prediction server: biological network integration for gene prioritization and predicting gene function. *Nucleic Acids Res*, **38**, W214-220.
8. Tzfadia, O., Amar, D., Bradbury, L.M., Wurtzel, E.T. and Shamir, R. (2012) The MORPH algorithm: ranking candidate genes for membership in Arabidopsis and tomato pathways. *The Plant cell*, **24**, 4389-4406.
9. Piro, R.M. and Di Cunto, F. (2012) Computational approaches to disease-gene prediction: rationale, classification and successes. *Febs J*, **279**, 678-696.
10. Boone, C. (2007) Global mapping of the yeast genetic interaction network. *Febs J*, **274**, 342-342.
11. Tong, A.H.Y., Lesage, G., Bader, G.D., Ding, H.M., Xu, H., Xin, X.F., Young, J., Berriz, G.F., Brost, R.L., Chang, M. *et al.* (2004) Global mapping of the yeast genetic interaction network. *Science*, **303**, 808-813.
12. de la Fuente, A. (2010) From 'differential expression' to 'differential networking' - identification of dysfunctional regulatory networks in diseases. *Trends Genet*, **26**, 326-333.
13. Mentzen, W.I., Floris, M. and de la Fuente, A. (2009) Dissecting the dynamics of dysregulation of cellular processes in mouse mammary gland tumor. *BMC Genomics*, **10**, 601.
14. Tesson, B.M., Breitling, R. and Jansen, R.C. (2010) DiffCoEx: a simple and sensitive method to find differentially coexpressed gene modules. *BMC Bioinformatics*, **11**.
15. Kelley, R. and Ideker, T. (2005) Systematic interpretation of genetic interactions using protein networks. *Nature biotechnology*, **23**, 561-566.
16. Shamir, R. and Ulitsky, I. (2009) Identifying functional modules using expression profiles and confidence-scored protein interactions. *Bioinformatics*, **25**, 1158-1164.
17. Ulitsky, I., Shlomi, T., Kupiec, M. and Shamir, R. (2008) From E-MAPs to module maps: dissecting quantitative genetic interactions using physical interactions. *Molecular Systems Biology*, **4**.
18. Shamir, R. and Ulitsky, I. (2007) Identification of functional modules using network topology and high-throughput data. *Bmc Systems Biology*, **1**.
19. Narayanan, M., Vetta, A., Schadt, E.E. and Zhu, J. (2010) Simultaneous Clustering of Multiple Gene Expression and Physical Interaction Datasets. *Plos Computational Biology*, **6**.

20. Kelley, D.R. and Kingsford, C. (2011) Extracting between-pathway models from E-MAP interactions using expected graph compression. *J Comput Biol*, **18**, 379-390.
21. Bandyopadhyay, S., Mehta, M., Kuo, D., Sung, M.K., Chuang, R., Jaehnig, E.J., Bodenmiller, B., Licon, K., Copeland, W., Shales, M. *et al.* (2010) Rewiring of genetic networks in response to DNA damage. *Science*, **330**, 1385-1389.
22. Leiserson, M.D.M., Tatar, D., Cowen, L.J. and Hescott, B.J. (2011) Inferring Mechanisms of Compensation from E-MAP and SGA Data Using Local Search Algorithms for Max Cut. *J Comput Biol*, **18**, 1399-1409.
23. Amar, D., Safer, H. and Shamir, R. (2013) Dissection of Regulatory Networks that Are Altered in Disease via Differential Co-expression. *PLoS Comput Biol*, **9**, e1002955.
24. Li, J.Y., Li, H.Q., Soh, D. and Wong, L. (2005) A correspondence between maximal complete bipartite subgraphs and closed patterns. *Lect Notes Artif Int*, **3721**, 146-156.
25. Li, J.Y., Liu, G.M., Li, H.Q. and Wong, L. (2007) Maximal biclique subgraphs and closed pattern pairs of the adjacency matrix: A one-to-one correspondence and mining algorithms. *Ieee T Knowl Data En*, **19**, 1625-1637.
26. Defays, D. (1977) Efficient Algorithm for a Complete Link Method. *Comput J*, **20**, 364-366.
27. Hedges, L.V. and Olkin, I. (1985) *Statistical methods for meta-analysis*. Academic Press, Orlando.
28. Schmid, J.E., Koch, G.G. and LaVange, L.M. (1991) An overview of statistical issues and methods of meta-analysis. *J Biopharm Stat*, **1**, 103-120.
29. Chatr-aryamontri, A., Breitkreutz, B.J., Heinicke, S., Boucher, L., Winter, A., Stark, C., Nixon, J., Ramage, L., Kolas, N., O'Donnell, L. *et al.* (2013) The BioGRID interaction database: 2013 update. *Nucleic Acids Research*, **41**, D816-D823.
30. Guenole, A., Srivas, R., Vreeken, K., Wang, Z.Z., Wang, S.Y., Krogan, N.J., Ideker, T. and van Attikum, H. (2013) Dissection of DNA Damage Responses Using Multiconditional Genetic Interaction Maps. *Molecular Cell*, **49**, 346-358.
31. Shamir, R., Maron-Katz, A., Tanay, A., Linhart, C., Steinfeld, I., Sharan, R., Shiloh, Y. and Elkon, R. (2005) EXPANDER - An integrative program suite for microarray data analysis. *BMC Bioinformatics*, **6**.
32. Ulitsky, I., Laurent, L.C. and Shamir, R. (2010) Towards computational prediction of microRNA function and activity. *Nucleic Acids Res*, **38**, e160.
33. Shamir, R., Ulitsky, I., Maron-Katz, A., Shavit, S., Sagir, D., Linhart, C., Elkon, R., Tanay, A., Sharan, R. and Shiloh, Y. (2010) Expander: from expression microarrays to networks and functions. *Nature Protocols*, **5**, 303-322.
34. Smoot, M.E., Ono, K., Ruscheinski, J., Wang, P.L. and Ideker, T. (2011) Cytoscape 2.8: new features for data integration and network visualization. *Bioinformatics*, **27**, 431-432.
35. Gallant, A., Leiserson, M.D., Kachalov, M., Cowen, L.J. and Hescott, B.J. (2013) Genecentric: a package to uncover graph-theoretic structure in high-throughput epistasis data. *BMC Bioinformatics*, **14**, 23.

36. Collins, S.R., Miller, K.M., Maas, N.L., Roguev, A., Fillingham, J., Chu, C.S., Schuldiner, M., Gebbia, M., Recht, J., Shales, M. *et al.* (2007) Functional dissection of protein complexes involved in yeast chromosome biology using a genetic interaction map. *Nature*, **446**, 806-810.
37. Baryshnikova, A., Costanzo, M., Kim, Y., Ding, H.M., Koh, J., Toufighi, K., Youn, J.Y., Ou, J.W., San Luis, B.J., Bandyopadhyay, S. *et al.* (2010) Quantitative analysis of fitness and genetic interactions in yeast on a genome scale. *Nature Methods*, **7**, 1017-U1110.
38. Dahlmann, B. (2005) Proteasomes. *Essays in biochemistry*, **41**, 31-48.
39. Bandyopadhyay, S., Kelley, R., Krogan, N.J. and Ideker, T. (2008) Functional maps of protein complexes from quantitative genetic interaction data. *PLoS Comput Biol*, **4**, e1000065.
40. Ma, X.T., Tarone, A.M. and Li, W.Y. (2008) Mapping Genetically Compensatory Pathways from Synthetic Lethal Interactions in Yeast. *Plos One*, **3**.
41. Lundin, C., North, M., Erixon, K., Walters, K., Jenssen, D., Goldman, A.S.H. and Helleday, T. (2005) Methyl methanesulfonate (MMS) produces heat-labile DNA damage but no detectable in vivo DNA double-strand breaks. *Nucleic Acids Research*, **33**, 3799-3811.
42. Parker, J.L. and Ulrich, H.D. (2012) A SUMO-interacting motif activates budding yeast ubiquitin ligase Rad18 towards SUMO-modified PCNA. *Nucleic Acids Research*, **40**, 11380-11388.
43. Blastyak, A., Pinter, L., Unk, I., Prakash, L., Prakash, S. and Haracska, L. (2007) Yeast Rad5 protein required for postreplication repair has a DNA helicase activity specific for replication fork regression. *Molecular Cell*, **28**, 167-175.
44. Brusky, J., Zhu, Y. and Xiao, W. (2000) UBC13, a DNA-damage-inducible gene, is a member of the error-free postreplication repair pathway in Saccharomyces cerevisiae. *Curr Genet*, **37**, 168-174.
45. Keogh, M.C., Kim, J.A., Downey, M., Fillingham, J., Chowdhury, D., Harrison, J.C., Onishi, M., Datta, N., Galicia, S., Emili, A. *et al.* (2006) A phosphatase complex that dephosphorylates gamma H2AX regulates DNA damage checkpoint recovery. *Nature*, **439**, 497-501.
46. Yeung, M. and Durocher, D. (2011) Srs2 enables checkpoint recovery by promoting disassembly of DNA damage foci from chromatin. *DNA Repair*, **10**, 1213-1222.
47. Mimura, S., Yamaguchi, T., Ishii, S., Noro, E., Katsura, T., Obuse, C. and Kamura, T. (2010) Cul8/Rtt101 Forms a Variety of Protein Complexes That Regulate DNA Damage Response and Transcriptional Silencing. *Journal of Biological Chemistry*, **285**, 9858-9867.
48. Vaisica, J.A., Baryshnikova, A., Costanzo, M., Boone, C. and Brown, G.W. (2011) Mms1 and Mms22 stabilize the replisome during replication stress. *Molecular Biology of the Cell*, **22**, 2396-2408.
49. Mortensen, U.H., Bendixen, C., Sunjevaric, I. and Rothstein, R. (1996) DNA strand annealing is promoted by the yeast Rad52 protein. *Proceedings of the National Academy of Sciences of the United States of America*, **93**, 10729-10734.

50. Araki, Y., Takahashi, S., Kobayashi, T., Kajiho, H., Hoshino, S. and Katada, T. (2001) Ski7p G protein interacts with the exosome and the Ski complex for 3 '-to-5 ' mRNA decay in yeast. *Embo J*, **20**, 4684-4693.
51. Anderson, J.S.J. and Parker, R. (1998) The 3 ' to 5 ' degradation of yeast mRNAs is a general mechanism for mRNA turnover that requires the SKI2 DEVH box protein and 3 ' to 5 ' exonucleases of the exosome complex. *Embo J*, **17**, 1497-1506.
52. Azzalin, C.M. and Lingner, J. (2006) The double life of UPF1 in RNA and DNA stability pathways. *Cell Cycle*, **5**, 1496-1498.
53. Arora, C., Kee, K., Maleki, S. and Keeney, S. (2004) Antiviral protein Ski8 is a direct partner of Spo11 in meiotic DNA break formation, independent of its cytoplasmic role in RNA metabolism. *Molecular Cell*, **13**, 549-559.
54. Hieronymus, H., Yu, M.C. and Silver, P.A. (2004) Genome-wide mRNA surveillance is coupled to mRNA export. *Genes & Development*, **18**, 2652-2662.
55. Myers, A.J., Webster, J.A., Gibbs, J.R., Clarke, J., Ray, M., Zhang, W.X., Holmans, P., Rohrer, K., Zhao, A., Marlowe, L. *et al.* (2009) Genetic Control of Human Brain Transcript Expression in Alzheimer Disease. *American Journal of Human Genetics*, **84**, 445-458.
56. Montojo, J., Zuberi, K., Rodriguez, H., Kazi, F., Wright, G., Donaldson, S.L., Morris, Q. and Bader, G.D. (2010) GeneMANIA Cytoscape plugin: fast gene function predictions on the desktop. *Bioinformatics*, **26**, 2927-2928.
57. Jiang, Q., Wang, Y., Hao, Y., Juan, L., Teng, M., Zhang, X., Li, M., Wang, G. and Liu, Y. (2009) miR2Disease: a manually curated database for microRNA deregulation in human disease. *Nucleic Acids Res*, **37**, D98-104.
58. Ben-Hur, A., Elisseeff, A. and Guyon, I. (2002) A stability based method for discovering structure in clustered data. *Pac Symp Biocomput*, 6-17.
59. Chia, B.K. and Karuturi, R.K. (2010) Differential co-expression framework to quantify goodness of biclusters and compare biclustering algorithms. *Algorithms for molecular biology : AMB*, **5**, 23.
60. Dembele, D. and Kastner, P. (2003) Fuzzy C-means method for clustering microarray data. *Bioinformatics*, **19**, 973-980.
61. Eisen, M.B., Spellman, P.T., Brown, P.O. and Botstein, D. (1998) Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences of the United States of America*, **95**, 14863-14868.
62. Gasch, A.P. and Eisen, M.B. (2002) Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering. *Genome Biology*, **3**.
63. McLachlan, G. (1998) Mathematical classification and clustering. *Psychometrika*, **63**, 93-95.
64. Ravasz, E., Somera, A.L., Mongru, D.A., Oltvai, Z.N. and Barabasi, A.L. (2002) Hierarchical organization of modularity in metabolic networks. *Science*, **297**, 1551-1555.
65. Sharan, R., Maron-Katz, A. and Shamir, R. (2003) CLICK and EXPANDER: a system for clustering and visualizing gene expression data. *Bioinformatics*, **19**, 1787-1799.
66. Van Dongen, S. (2000) Graph Clustering by Flow Simulation. *In PhD Thesis. University of Utrecht.*

67. Vlasblom, J. and Wodak, S.J. (2009) Markov clustering versus affinity propagation for the partitioning of protein interaction graphs. *BMC Bioinformatics*, **10**.
68. Madeira, S.C. and Oliveira, A.L. (2004) Biclustering algorithms for biological data analysis: A survey. *Ieee Acm T Comput Bi*, **1**, 24-45.
69. Tanay, A., Sharan, R., Kupiec, M. and Shamir, R. (2004) Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data. *Proceedings of the National Academy of Sciences of the United States of America*, **101**, 2981-2986.
70. Watson, M. (2006) CoXpress: differential co-expression in gene expression data. *BMC Bioinformatics*, **7**.

Supplementary Information

Supplementary Figure 1: Illustration of the DICER algorithm local search

Supplementary Figure 2: Possible pitfalls of local improver that can be solved by the global improver

Supplementary Figure 3: Performance of module map algorithms on simulated data with 1000 nodes and 20 modules.

Supplementary Figure 4: Performance of $DICER_k$ variants on simulated unweighted data with 1000 nodes and 20 modules

Supplementary Table 1: Yeast interactions used in the analysis of PPI and negative GI networks

Supplementary Table 2: Results of the simulations

Supplementary Table 3: Performance of algorithms in the analysis of the yeast PPI and negative GI networks

Supplementary Table 4: ModMap solution on the yeast PPI and negative GI networks

Supplementary Table 5: Enrichment analysis of the ModMap solution on the yeast PPI and negative GIs networks

Supplementary Table 6: The highly significant module links in the ModMap solution on yeast PPIs and negative GIs networks

Supplementary Table 7: The set of positive GIs that are specific to yeast cells treated with DNA damage agent MMS

Supplementary Table 8: ModMap solution on the PPI and DNA damage-specific GI data

Supplementary Table 9: Enrichment analysis of the ModMap modules in the PPI and DNA damage-specific GI data

Supplementary Table 10: The module links of the ModMap solution in the PPI and DNA damage-specific GI data

Supplementary Table 11: Cross-validation results in the NSCLC data

Supplementary Table 12: Cross-validation results in the AD data

Supplementary Text

Supplementary Table 13: ModMap solution on the NSCLC data

Supplementary Table 14: Module map links of the ModMap solution in the NSCLC data

Supplementary Table 15: Enrichment analysis of the modules in the ModMap solution in the NSCLC data