DiffNet: Automatic Differential Functional Summarization of dE-MAP Networks

Boon-Siew Seah [1,3], Sourav S. Bhowmick [1,3,*] and C. Forbes Dewey, Jr [2,3]

[1] School of Computer Engineering, Nanyang Technological University, Singapore

[2] Biological Engineering Department, Massachusetts Institute of Technology, Massachusetts, USA

[3] Singapore-MIT Alliance, Nanyang Technological University, Singapore

Corresponding email: assourav@ntu.edu.sg

August 20, 2014

Abstract

The study of genetic interaction networks that respond to changing conditions is an emerging research problem. Recently, Bandyopadhyay et al., 2010 [2] proposed a technique to construct a *differential network* (*dE-MAP network*) from two static gene interaction networks in order to map the interaction differences between them under environment or condition change (*e.g.,* DNA-damaging agent). This differential network is then manually analyzed to conclude that DNA Repair is differentially effected by the condition change. Unfortunately, manual construction of *differential functional summary* from a dE-MAP network that summarizes all pertinent functional responses is time-consuming, laborious and error-prone, impeding large-scale analysis on it. To this end, we propose DiffNet, a novel data-driven algorithm that leverages Gene Ontology (GO) annotations to automatically summarize a dE-MAP network to obtain a high-level map of functional responses due to condition change.

We tested DiffNet on the dynamic interaction networks following MMS treatment and demonstrated the superiority of our approach in generating differential functional summaries compared to state-of-the-art graph clustering methods. We studied the effects of parameters in DiffNet in controlling the quality of the summary. We also performed a case study that illustrates its utility.

Keywords: Differential Network, Gene Interaction Network, Differential Functional Summarization.

Highlights:

- A problem model for summarizing differential networks is described.
- Summary finds key functional modules that respond to condition change.
- A solution that solves the problem model is proposed (DiffNet).
- We report the functional responses of the yeast network after MMS treatment.

1 Introduction

High-throughput mapping of genetic interaction networks of a set of genes is an important and emergent research problem [5]. The networks constructed with these methods, however, only represent a *static* "snapshot" of the genetic interaction map under a particular context or condition. Recent studies have shown that genetic interaction maps are in fact *dynamic* and *context-dependent* [18]. Consequently, there is a growing interest in studying the system-wide responses of interaction networks following environmental or condition change [10, 15]. For instance, one may be interested in elucidating the genetic interaction differences between cancer cells and normal cells. Specifically, some interactions may appear or disappear in the disease state, intensity of some interactions may alleviate or aggravate when in disease state compared to healthy condition, and others may remain strong irrespective of the state.

One representative method that has been recently proposed for mapping the genetic interaction responses following environment change is the `dE-MAP` approach [2]. In this method, two static gene interaction networks [5] for each condition are first obtained using the *epistatic miniarray profile* (`E-MAP`) approach [17], which constructs a quantitative genetic interaction landscape of *S. cerevisiae* by first identifying a set of genes of interest. Double mutant strains of all pairwise genes from this set of genes are then grown and their colony size measured. Genetic interaction occurs between a pair of mutant genes when one observes greater or lesser than expected colony growth rate when compared to their respective single mutant strains. When the growth rate is greater than expected, the interaction is deemed *positive* (alleviating); when it is lesser, it is deemed *negative* (aggravating). Using the two static `E-MAP` networks, a *differential network* (`dE-MAP` network) is then computed that maps the interaction differences between the two static networks. For example, in [2], *S. cerevisiae* `E-MAP` networks are obtained for cells grown under two conditions: (a) cells which are treated with methyl methanesulfonate (`MMS`), a well known DNA-damaging agent and (b) cells which are untreated. Large-scale genetic interaction network among 418 yeast genes are quantitatively extracted using the `E-MAP` method under the `MMS`-treated condition (stressed) and untreated condition (unstressed) and the differential network that maps the genetic interaction changes due to `MMS` challenge is computed. Figure 1 depicts an example of a differential network (partial view) that is obtained from two static `E-MAP` networks under `MMS`-treated and untreated condition.

Naturally, it is important to analyze this differential network to investigate the system-wide impact of the DNA-damaging agent on the functional roles of various components. Consequently, the authors obtained physical protein-protein interactions corresponding to these genes and performed graph clustering to find protein complexes[1] enriched with differential interactions. The functional identity of each cluster is then *manually*[2] determined. Particularly, the authors concluded that these complexes tend to be stable across conditions and differential interactions largely lie between complexes, rather than within complexes. Unfortunately, modules constructed in this manner poorly represent the functional responses of the differential network. Hence, to find a functional response, the authors manually selected a subset of 31 genes associated with `DNA repair` to test for differential interaction enrichment, concluding that `DNA repair` is a pertinent functional response following `MMS`-treatment.

[1] The topology of the differential network can be mined to identify gene clusters using techniques such as [1, 8, 14].

[2] A function can also be associated with each cluster by leveraging a *functional enrichment* technique [3].

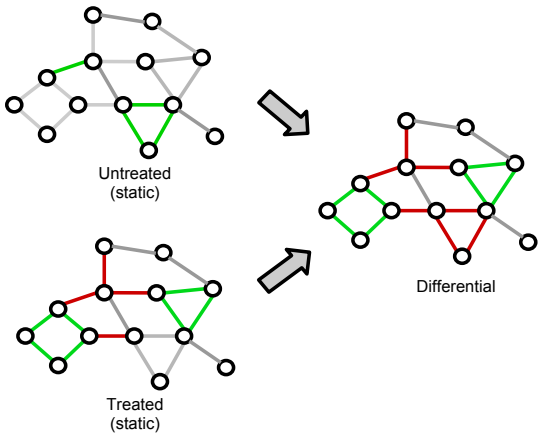


Figure 1. The differential network that arises from two static E-MAP networks under different conditions. Red interactions – positive differential; green interactions – negative differential.

However, it is time-consuming, laborious and error-prone to perform large-scale analysis of dE-MAP interactomes to map all pertinent functional responses. *In this paper, we propose a novel technique called DiffNet that addresses this impediment by automatically constructing a high quality differential summary of two E-MAP networks under environmental change.* Figure 2 highlights some of these functional modules that are differentially effected by the DNA-damaging agent.

At first glance, the aforementioned failure of traditional graph clustering techniques to capture differential summaries in its modules may seem surprising. However, as we shall see in Section 4, these techniques are largely designed for static networks and are less suitable for differential networks that contain both positive and negative weights. Furthermore, since most methods rely solely on topology of the network, there is also no guarantee that each cluster corresponds well to a representative biological function response. In fact, as remarked earlier, in [2] the functional identity of each cluster following graph clustering is manually determined. Furthermore, the authors failed to assign function to a significant number of these clusters.

In fact, algorithms that perform genome-wide functional analysis of gene responses under multiple conditions have been proposed in the literature [9, 19, 20]. Particularly, these approaches perform functional analysis based on the expression levels of genes. In contrast, in our problem we focus on genome-wide functional analysis of the *gene interactions* and their responses.

Given the differential network generated from dE-MAP interactions, DiffNet greedily constructs a *differential summary* comprising of a set of *skewed* and *coherent functional subgraphs*, representing significant functional responses following environment or condition change. Specifically, it leverages Gene Ontology (GO) annotations to identify these functional subgraphs, each of which represents a group of interactions corresponding to a specific biological function. A key characteristic of these functional subgraphs is that the interactions together respond *significantly* in one direction, either positively or negatively, to the condition change. That is, unlike standard graph clustering methods, DiffNet is specifically designed to handle *differential interactions*,

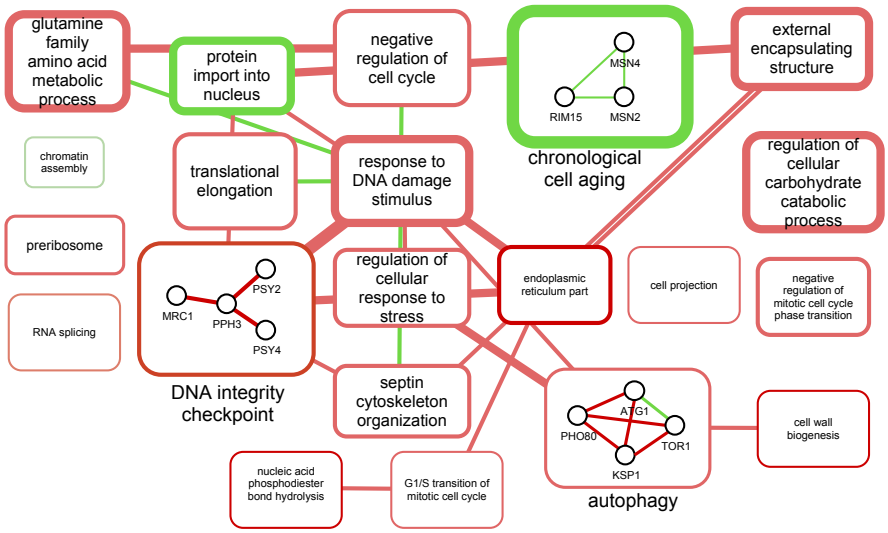


Figure 2. Differential functional summary of `MMS`-induced/untreated yeast `de-MAP` network in [2]. The color of the functional modules and gene interactions indicate either positive differential (red) or negative differential (green). The thickness of the lines indicate the strength of the differential response. Gene interaction subgraphs of selected functional modules are also shown. Edges between functional modules depict differential interactions that occur between functional modules. The thickness of these edges represent the skewness of the differential interactions between a pair of functional modules. The most significant of such edges are shown.

---

which can be positively or negatively weighted. Figure 3 illustrates the idea of the `DiffNet` algorithm. We shall elaborate on it in the next section.

2 Summary of Proposed Method

`DiffNet` is a novel data-driven algorithm that automatically summarize a `dE-MAP` network to obtain a high-level map of functional responses due to condition change.

- **Input:** A `dE-MAP` network
- **Output:** A high-level summary of functional responses (both positive and negative reponses) due to condition change.
- **Tools used in the proposed method:** Scala
- **Databases, if any, used in the proposed method**: Gene Ontology Annotations dataset (GOA)

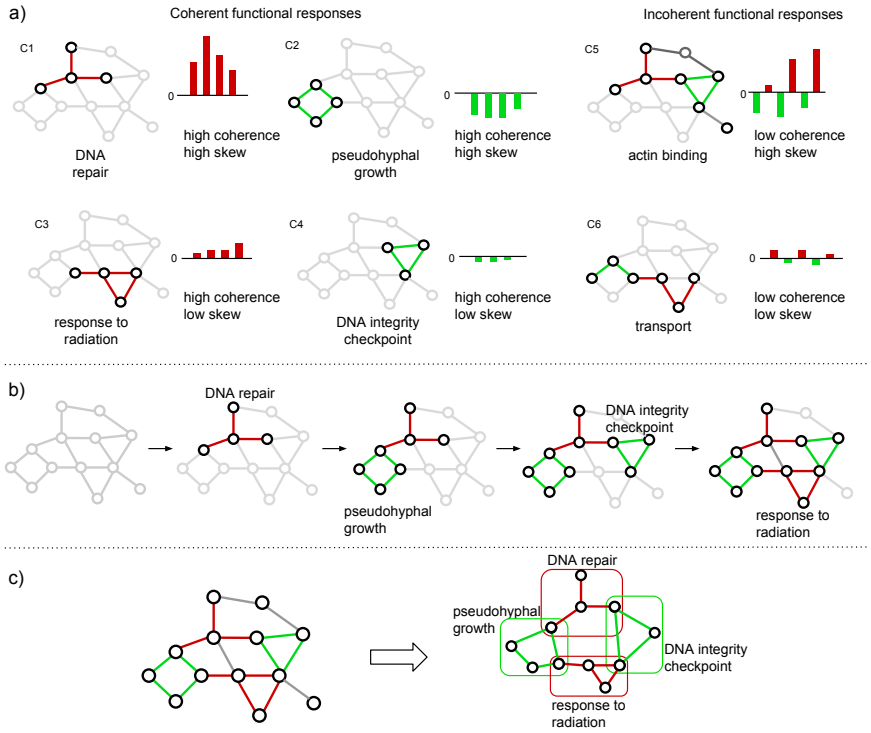


Figure 3. Illustration of `DiffNet`. Red interactions are positive differential, while green interactions are negative differential. a) A functional subgraph represents interacting genes that share a specific function (*e.g.*, C1 represents gene interactions associated with `DNA repair`). A coherent functional subgraph has differential interactions that mostly respond in one direction. We say that a functional subgraph has high skew if the differential interaction weights have high magnitude; it has high coherence when the interactions largely respond in one direction. A functional subgraph with high coherence and skew represents a concerted, significant functional response due to the condition change. b) The `DiffNet` algorithm implements a greedy heuristic that selects, at each iteration, the functional subgraph with highest coherence and skew from the remaining unselected interactions. c) The output of `DiffNet` is a decomposition that summarizes the relevant functional responses due to condition change.

3 Materials and Methods

3.1 Constructing Differential Networks

The set of genes of interest together with their genetic interactions can be modeled as a gene-gene interaction network, denoted by $G = (V, E, w)$, where V is a set of genes selected for `E-MAP` study, E denotes the pairwise interactions between genes, and w is a function that assigns each pairwise interaction $e \in E$ a weight that represents its interaction strength. In `E-MAP` studies, $w(e)$ of $e \in E$ is given by its genetic interaction score *S-score* [17]. A positive *S-score* indicates the degree of alleviating interaction between the two genes whereas a negative *S-score* indicates the degree of aggravating interaction. Therefore, $w(e)$ can be positive or negative.

Consider now two `E-MAP` networks $G_t = (V, E, w_t)$ and $G_c = (V, E, w_c)$ that represent two conditions: (a) the treated condition (G_t) and (b) the untreated condition (G_c). Observe that G_t and G_c share the same set of vertices and pairwise interactions. Given G_t and G_c, the differential network of G_t and G_c is a graph $G_d = (V, E, w_d)$ such that $\forall e \in E$:

$$w_d(e) = \left(1 + e^{-\frac{w_t(e) - w_c(e)}{|w_c(e)|}}\right)^{-1} - 0.5 \tag{1}$$

We apply the logistic function $(1 + e^{-x})^{-1}$ (shifted by 0.5 to make it an odd function) to "clip" potentially large magnitudes of differential responses. This is inspired by a similar approach used in activation functions in neural networks to bound the response of signals [11].

Intuitively, a differential network models gene interaction responses due to condition change. The differential weight $w_d(e)$ represents the normalized difference in *S-scores* between the two conditions for a pair of genes represented by e. We call $w_d(e)$ *positive differential* when $w_d(e) > 0$, and *negative differential* when $w_d(e) < 0$. A positive (*resp.* negative) differential response indicates increased alleviating (*resp.* aggravating) interaction between the two genes in treated condition compared to untreated condition. The magnitude of $w_d(e)$ reflects the strength of interaction response due to condition change. Figure 4 shows a toy differential network of positive (red) and negative (green) differential interactions. Grey colored interactions do not respond to condition change (*i.e.,* $w_d(e) \approx 0$). The interaction between `RAD52` and `SIN3`, for instance, has a positive differential response due to condition change.

It is worth noting that the above definition of differential interaction w.r.t DNA damage-induced `dE-MAP` network is consistent with the one in [2]. Specifically, a positive differential interaction indicate DNA damage-induced lethality, while a negative differential interaction indicate inducible epistasis or suppression. Importantly, the differential response does not distinguish, for example, one that goes from negative to positive from one that goes from positive to more positive. Although the former is arguably more interesting, the latter still is biologically significant because it indicates a significant response due to treatment.

Although we now have a model of individual gene-gene interaction responses due to condition change, it remains unclear how one *automatically* infers broader, systemic functional responses from these detailed interactions. This issue is pertinent in high-throughput experiments, which often generate thousands, even millions, of interacting genes within a single experiment. Hence we present our approach to model responses due to condition change from a functional perspective.

3.2 Functional Subgraphs in a Differential Network

We begin by modeling a systemic functional response by a subgraph of functionally-similar gene interactions (*i.e.,* a set of genes of a specific function and their interactions). Let $\Delta = \{T_1, T_2, \ldots\}$ be a set of GO terms in the Gene Ontology. This represents the set of biological functions relevant to our study. Every gene $v \in V$ is annotated with zero or more biological functions in Δ. Then a *functional subgraph*, denoted by $C_T = (V_T, E_T)$, is a subgraph of G_d such that: (a) C_T is a subgraph of G_d induced by V_T, and (b) every gene $v \in V_T$ shares a function $T \in \Delta$. For instance, the subgraph $C1$ in Figure 3(a) is a functional subgraph of genes sharing the `DNA repair` function. One can see that a functional subgraph models the interaction responses of genes with a specific function as a whole.

We evaluate each functional subgraph C_T with the *skewness* and *coherence* measures. We say that a functional subgraph is *skewed* if its interactions significantly respond to condition change (*i.e.,* the interactions in the subgraph are significantly positive or negative differential). Analogous to individual gene interactions, we call a subgraph $C_T = (V_T, E_T)$ *positively skewed* if the sum of its edge weights, defined as $skew(C_T) = \sum_{e \in E_T} w_d(e)$, is greater than 0; it is *negatively skewed* if the sum of its edge weights is less than 0, *i.e.*, $skew(C_T) < 0$. The greater the value of $skew(C_T)$, the more the interactions of C_T respond to condition change.

We say that a functional subgraph is *coherent* if its interactions are largely skewed in one direction (either positive or negative differential). Figure 3(a) depicts the coherence of subgraphs of the toy network in Figure 4. Consider the subgraph representing `DNA repair` function. It is coherent because it consists of interactions that are skewed towards positive differential in tandem. Intuitively, this would mean that the `DNA repair` function, as a whole, has increased alleviating response due to the condition change. Meanwhile, the subgraph representing `transport` has a mix of positive and negative differential interactions. There is no clear indication whether the `transport` function is positively or negatively affected by the condition change. We now formally define the notion of *subgraph coherence*. Given a subgraph C_T, $coherence(C_T) \in [0, 1]$ is given by:

$$coherence(C_T) = \frac{max(|\{e : w_d(e) > 0\}|, |\{e : w_d(e) < 0\}|)}{|E_T|} \tag{2}$$

The greater the value of $coherence(C_T)$, the more coherent is the subgraph. If $coherence(C_T) = 1$ then it indicates that all interactions are exclusively positive differential or exclusively negative differential.

Figure 3(a) depicts the skewness and coherence of several functional subgraphs. Each bar graph associated with a functional subgraph depicts the differential weight w_d values of the interactions in the subgraph. A high coherence and high skew subgraph has interactions with large w_d values in one direction. On the other hand, a low coherence and low skew subgraph has low w_d values in diverging directions. Consider the following two functional subgraphs: the subgraph of genes sharing the `DNA repair` function (`RAD5, RAD52, SIN3, ASH1`), and subgraph of genes sharing the `transport` function (`MSN1, ASH1, MRC1`, `PPH3`, `PSY4`, `PSY2`). Observe that interactions in the former are positive differential and skewed in one coherent direction, while the latter is not. We are more interested in the former type of subgraphs because it represents a concerted and significant functional response due to the condition change. Generally, functional subgraphs that are high skew and high coherence are informative and represent significant functional responses due to condition change. On the other hand, a

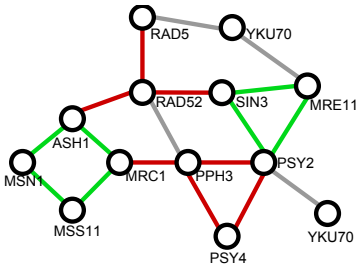


Figure 4. A toy differential network of gene interactions.

subgraph with both low coherence and skew represent function that remain relatively unchanged.

From a statistical point of view, a module constructed from interactions that are unaffected by condition change will have similar interaction distributions, resulting in a coherence score centered around 0 (zero coherence). A high coherence module represents a module with significant change in interaction distribution profile, thus representing a statistically significant module. Biologically, analogous to functional enrichment in gene lists, the statistical significance of high entropy modules means that the function associated with such module exhibit statistically significant interaction response patterns compared to a random function.

Based on the above observation, if one can decompose G_d into a set of highly coherent and skewed functional subgraphs, denoted by $\mathcal{S} = \{C_{T^1}, C_{T^2}, \ldots C_{T^k}\}$, then one can meaningfully obtain a summary representing positive and negative functional responses of G_d due to condition change. We shall later describe how one quantifies the decomposition of G_d based on the coherence and skewness of its functional subgraphs. Consider the decomposition depicted in Figures 3(b)-(c). The network of differential interactions is summarized into a set of functional subgraphs representing the following functional responses – `DNA repair` (positive), `response to radiation` (positive), `DNA integrity checkpoint` (negative) and `pseudohyphal growth` (negative). Each subgraph is coherent and skewed towards either positive or negative differential response.

At this point, it remains unclear how to optimally decompose G_d into a set of coherent and skewed functional subgraphs. To contrast with the previous example, suppose we decompose G_d into $\mathcal{S}$ = {`transport` (`MSN1, ASH1, MRC1, PPH3, PSY4, PSY2`), `response to radiation` (`MRC1`, `PPH3`, `PSY4`, `PSY2`)}. This decomposition poorly summarizes the network in Figure 4 because a significant portion of differential interactions are not captured by the subgraphs in $\mathcal{S}$. The `transport` subgraph also has low coherence.

3.3 Modeling the Differential Summarization Problem

Given the existence of potentially many possible decompositions of G_d, the problem of *differential summarization* is to identify the *best* decomposition that represents the functional responses in G_d. Suppose we have a set containing all possible functional subgraphs of G_d. Let us denote this set by the universe $\mathcal{E}$. Clearly, some subgraphs will represent meaningful functional responses, while others will be unaffected by the condition change. One would like to choose a subset of $\mathcal{E}$ representing functional responses in G_d that are significantly affected by the condition change. To do this, we

must first identify *summarization objectives* that assess the quality of a decomposition of G_d. We argue that a good decomposition of G_d should have the following desirable summary objectives:

- **Subgraph Coherence and Skewness.** A decomposition $\mathcal{S}$ should comprise of functional subgraphs that are significantly coherent and skewed. Recall that our goal is to identify functional regions that significantly respond, either positively or negatively, to condition change. This directly correlates to having coherent and skewed functional subgraphs, and finding $\mathcal{S}$ that maximizes coherence and skewness of its functional subgraphs is desirable. The *differential score* of C_T combines the skewness and coherence of the subgraph as follows:

$$differential(C_T) = coherence(C_T)^{\alpha} \times skew(C_T) \tag{3}$$

  where $\alpha \geq 0$ is a parameter controlling the influence of coherence on the differential score. Note that $0 \leq coherence(C_T)^{\alpha} \leq 1$.

- **Edge Coverage.** A good decomposition of G_d should convey key information regarding functional regions affected by condition change. It is natural to prefer a decomposition that covers as much differential interactions in G_d as possible. We introduce the *edge coverage* measure that reflects how well $\mathcal{S}$ represents the differential interactions of G_d. Formally, the *edge coverage* of $\mathcal{S}$ can be expressed as:

$$coverage(\mathcal{S}) = \frac{|\bigcup_{C_i \in \mathcal{S}} E_i|}{|E|} \tag{4}$$

  Intuitively, it indicates the percentage of interactions in G_d that is represented by the subgraphs in $\mathcal{S}$. The wider the coverage, the more representative is the decomposition of the interactions in G_d.

- **Distinctiveness.** Intuitively, two functional subgraphs having disjoint differential interactions is more informative than two redundant subgraphs with identical interactions. Thus, one prefers a decomposition which cleanly partitions G_d into distinctive sets of interactions. We quantify this objective with the *distinctiveness* measure. It quantifies redundancy of functional subgraphs, such that the greater the redundancy, the lower the distinctiveness value. Hence, distinctiveness of $\mathcal{S}$ is 1 if its subgraphs are mutually disjoint. Formally, it is defined as:

$$distinctiveness(\mathcal{S}) = \frac{|\bigcup_{C_i \in \mathcal{S}} E_i|}{\sum_{C_i \in \mathcal{S}} |E_i|} \tag{5}$$

We introduce an optimization model that selects functional subgraphs to maximally cover the set of differential interactions of G_d to maximize the above objective scores. Because the set of possible functional subgraphs can be large, a naive ranking approach of selecting the most significantly coherent and skewed subgraphs can be suboptimal. There is no control on coverage and distinctivenes, leading to significant redundancy in the results. Thus, we propose an optimization model to construct a summary that satisfies all three desireable objective scores. This optimization model can be posed as a *weighted k-set cover* problem [4] of choosing a subset $\mathcal{S} \subseteq \mathcal{E}$ and a set of *remainder subgraphs* $\mathcal{R}$ with cardinality constraint k that minimizes the reciprocal of $differential(\mathcal{S})$. A remainder subgraph $R = (V_R, E_R) \in \mathcal{R}$ is a subgraph of G that is not part of the summary (*i.e.,* $R \cap C_T = \emptyset$ for all $C_T \in \mathcal{S}$). We shall later introduce a penalty for having remainder subgraphs.

Definition 1 ***[Differential summarization problem]****. Let G_d be the differential network of two gene interaction networks, G_c and G_t, under different conditions. Let $U = \bigcup_{C_T \in \mathcal{E}} E_T$ be the universe of differential interactions in G_d where $\mathcal{E}$ is a set of all possible functional subgraphs C_T. The* ***differential summarization problem*** *is to identify the* differential decomposition $\mathcal{S}$ *of functional subgraphs and $\mathcal{R}$ of* remainder subgraphs *(representing unselected interactions) by solving the following optimization problem:*

$$\begin{aligned} \arg\min_{\mathcal{S}\cup\mathcal{R}} f(\mathcal{S}\cup\mathcal{R}) = \quad & \arg\min_{\mathcal{S}} \sum_{C_T\in\mathcal{S}} differential^{-1}(C_T) + \sum_{R\in\mathcal{R}} r(R) \\ \textit{subject to} \qquad E \quad = \quad & \bigcup_{C_T\in\mathcal{S}} E_T \cup \bigcup_{R\in\mathcal{R}} E_R \\ |\mathcal{S}| + |\mathcal{R}| \quad \leq \quad & k \end{aligned}$$

where the $differential^{-1}(C_T)$ – the reciprocal of the coherence and skewness of C_T – is the cost associated with each functional subgraph $C_T \in \mathcal{S}$, and $r(R) = (|E_R| + 1)\max_{C_T\in\mathcal{E}} differential^{-1}(C_T)$ captures the penalty for not covering the edges of the network.

It can be proven that there is at most one remainder subgraph that can be selected, which is disjoint from all functional subgraphs in $\mathcal{S}$. Because of $r(R)$, the formulation penalizes a summary that provides low interaction coverage. Also, observe that in principle the above cost function penalizes functional subgraphs with low coherence or skewness scores. The decomposition $\mathcal{S}$ summarizes the key functional responses representing the differences between G_c and G_t. The cardinality constraint k controls the distinctiveness and coverage of the decomposition.

3.4 Solving the Differential Summarization Problem

Unfortunately, the differential summarization problem defined in the preceding section is NP-hard because it is posed as a weighted k-set cover problem [13]. Hence, we describe an algorithm called `DiffNet` that solves this problem heuristically. Here, we adopt a greedy algorithm that admits a H_k-approximation algorithm for the weighted minimum k-set cover problem [4], where $H_k = \sum_{i=1}^{k} \frac{1}{i}$. First, the differential network G_d is computed. Following that, `DiffNet` finds the universe of candidate functional subgraphs of G_d. The basic principle of `DiffNet` is to select, at each iteration, the functional subgraph that gives the best differential score contribution to the existing $\mathcal{S}$. At each iteration, we choose a functional subgraph that maximizes the total differential score. To achieve this, the algorithm maintains a map of interactions of G_d that is represented by currently selected functional subgraphs. For every candidate functional subgraph evaluated for selection, we evaluate its contribution to the remaining unselected interactions. The greedy algorithm then chooses the candidate subgraph that adds the highest differential score to the current summary. This process is iterated until k subgraphs have been selected. Because the penalty of choosing a remainder subgraph is always higher than any functional subgraphs, we let the remainder subgraph, if any, be the last subgraph. Algorithm 1 outlines the pseudocode of the above procedure. Given k passes and the worst case of evaluating $|E|$ edges per subgraph, the proposed algorithm has a worst case complexity of $O(k|\Delta||E|)$.

Algorithm 1 `DiffNet`.

Input: $G_t = (V, E, w_t)$, $G_c = (V, E, w_c)$, Δ, k
Output: $\mathcal{S}$

Let $p_{\max} = 0$
for $e \in E$ **do**
  $w_d(e) = (1 + e^{-\frac{w_t(e) - w_c(e)}{|w_c(e)|}}) - 0.5$
end for
Let $G_d = (V, E, w_d)$
Let $\mathcal{E} = \emptyset$
for $T \in \Delta$ **do**
  $\mathcal{E} \leftarrow \mathcal{E} \cup \{C_T\}$
end for
Let $\mathcal{S} = \emptyset$
repeat
  $mincost \leftarrow \infty$
  $best \leftarrow \emptyset$
  for all $C_T = (V_T, E_T) \in \mathcal{E} \setminus \mathcal{S}$ **do**
    $SelectedEdges \leftarrow \bigcup_{C \in \mathcal{S}} E$
    $n \leftarrow \left| E_T \setminus SelectedEdges \right|$
    $f \leftarrow differential^{-1}(C_T)/n$
    if $f < mincost$ and $n > 0$ **then**
      $mincost \leftarrow f$
      $best \leftarrow \{C_T\}$
    end if
  end for
  $\mathcal{S} \leftarrow \mathcal{S} \cup best$
until $|\mathcal{S}| > k$
return $\mathcal{S}$

4 Results

The `DiffNet` algorithm is implemented in Scala. We now present experimental results of the performance of `DiffNet`. The experiments were conducted on a 1.66GHz Intel Core 2 Duo T5450 machine with 3GB memory. Unless specified otherwise, we set $k = 45$ and $\alpha = 5.0$.

4.1 Functional analysis of `MMS`-treated/untreated dE-MAP Network

Using the two `E-MAP` networks in [2], we constructed the differential functional summary associated with `MMS` treated/untreated genetic interactions. Figure 2 shows the differential functional summary of the yeast genetic interactome. We observe significant positive differential functional subgraphs associated with DNA damage and DNA integrity checkpoint. The chronological cell aging genes responsible for stress-resistance – `MSN2,MSN4,RIM15` [7] – also undergo significant genetic interaction remodeling following DNA damage. This important and top-scoring functional response is not identified using manual analysis in [2]. The reason why this module could not be detected in [2] is due to their approach of performing cluster analysis on protein-protein

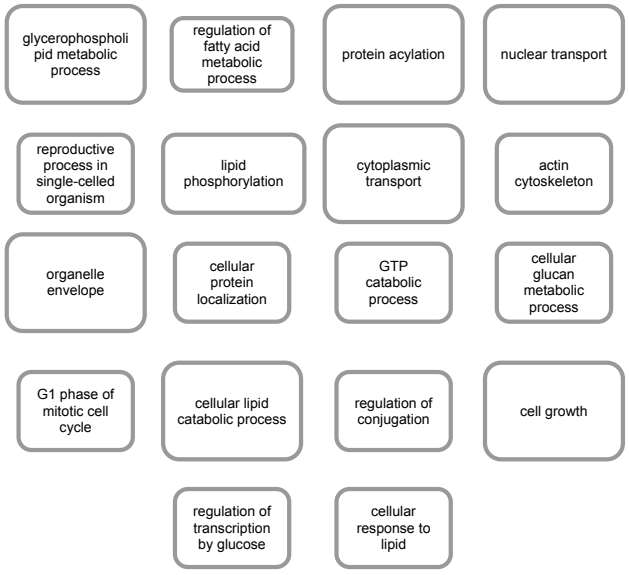


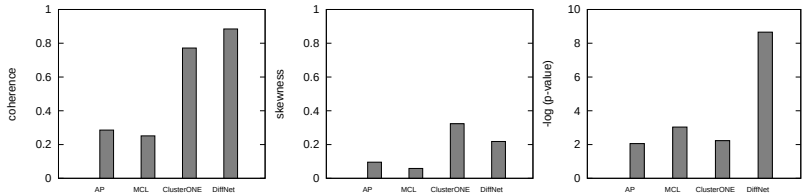
Figure 5. Functional summary of stable modules.

---

interaction (PPI) network rather than the differential interaction network itself. Thus, the set of genes, which has less PPI interaction density compared to protein complexes in the PPI network, was missed via convential cluster analysis. Another type of functional modules that demonstrate significant differential following `MMS` treatment are pathways related to apoptotis and cell cycle, such as the `G1` phase of mitotic cell cycle and cell aging modules. More interestingly, we observe significant negative differential responses in cell projection and cell wall biogenesis functions. The manual functional enrichment study conducted in [2] did not uncover the negative shift of these less obvious groups of genes. The autophagy module, which is a cellular catabolic process, is also seen to be positively activated [16]. Recently, DNA damage has been shown to induce autophagy [16], although the mechanism that triggers remains unclear. Apart from activating autophagy processes, DNA damage is also found to induce actin and septin rearrangement [12]. This is discovered by the differential functional summary, which finds positive activation of septin cytoskeleton organization module.

To contrast the differential functional summary, we also constructed a summary of functional subgraphs that shows subgraphs of genes whose genetic interactions remain largely unaltered after `MMS` treatment. To this end, instead of constructing the differential network G_d, we constructed an "inverse" differential network $G_s = (V, E, w_s)$, such that $w_s = min((w_d(e))^{-1}, \epsilon^{-1})$ where $e \in E$ and ϵ represents a pseudocount that prevents $w_s(e) \to \infty$. Observe that w_s represents the inverse of normalized *S-score* differences. We applied `DiffNet` on G_s to obtain a landscape of "stable" functional subgraphs, that is, functional subgraphs that are neither strongly positive differential nor strongly negative differential.

Figure 5 shows the functional summary of G_s following `MMS` treatment. The modules represented in this summary could be "housekeeping" processes and modules



(a) Coherence (b) Skewness (c) Func. Homogeneity

Figure 6. Comparison with general graph clustering algorithms.

whose genetic interaction strength remain unaltered regardless of the DNA-damage challenge [2]. For instance, the composition and interaction of the subunits of the RNA polymerase enzyme, a critical module of the cell regardless of cellular context, is unlikely to change. Thus, their genetic interactions should also remain stable. One can make the same argument for preribosome.

4.2 Comparison with Graph Clustering Algorithms

Since there is no existing technique that automatically generates differential functional summaries, we are confined to compare `DiffNet` with several representative graph clustering methods such as `MCL` [6], Affinity Propagation (`AP`) [8], and `ClusterONE` [14]. We used the dataset in [2] containing 418 genes (393 with annotations). In particular, we chose the `MCL` and `ClusterONE` approaches as a recent evaluation demonstrated that both these methods outperform other graph clustering algorithms on biological networks [14]. Because `MCL` and `ClusterONE` do not accept negative edge weights, they cannot be directly applied to differential networks. To this end, we constructed two separate networks from a differential network – (a) a *positive network* containing only positive differential edges and (b) a *negative network* containing only negative differential edges. We assessed whether individually clustering both networks using general graph clustering methods, and then aggregating the clusters into one list, could provide results similar to those generated by `DiffNet`. For all approaches, we discarded clusters with fewer than 3 genes and selected the 25 best scoring clusters for cluster quality evaluation.

To quantitatively evaluate the quality of the clusters, we introduce several evaluation measures. Given a set of cluster subgraphs $\mathcal{S}$, the *average coherence* and *average skewness* are given by:

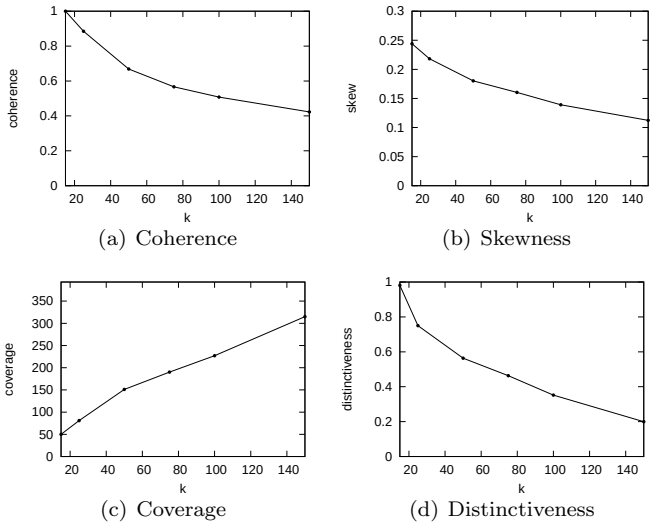
$$AvgCoherence(\mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{C_T \in \mathcal{S}} coherence(C_T) \quad (6)$$

$$AvgSkewness(\mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{C_T \in \mathcal{S}} skew(C_T) \quad (7)$$

To assess the functional relevance of each cluster, we used the annotation over-representation analysis of the clusters [21]. To this end, the *functional homogeneity* of $\mathcal{S}$ is given by :

$$FuncHomo(\mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{C_T=(E_T,V_T) \in \mathcal{S}} -log(p - value(V_T)) \quad (8)$$

where $p\text{-}value(V_T)$ is the most significant GO term enrichment $p\text{-}value$ score of the genes in V_T.



(a) Coherence (b) Skewness

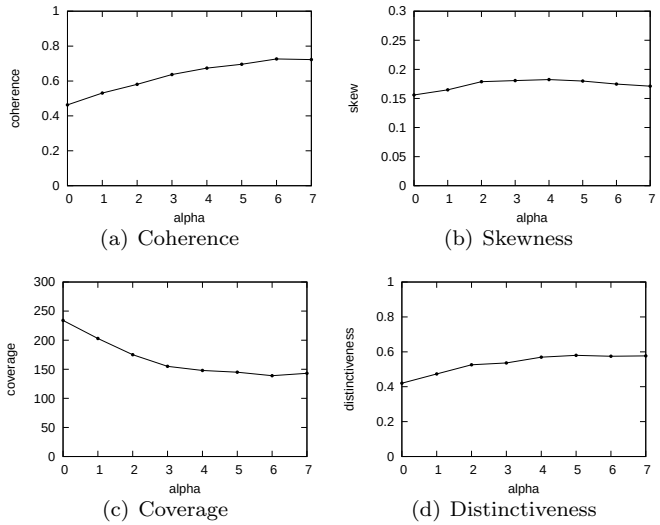
(c) Coverage (d) Distinctiveness

Figure 7. Effect of parameter k on `DiffNet`.

Figure 6 plots the results of different approaches. Observe that `DiffNet` is superior to the clustering techniques in the following ways. First, each subgraph in `DiffNet` has a direct association with a biological function. Recall that functional subgraphs have the constraint that every gene in a subgraph must share a specific function. With graph clustering algorithms such as `MCL`, each subgraph cluster may contain genes with diverging functions. In that case, it is unclear what biological function the cluster represents. This is quantified by the superior functional homogeneity score of `DiffNet`. Second, subgraphs in `DiffNet` have superior coherence compared to other methods. Traditional graph clustering methods are not designed to identify clusters of positive differential interactions and negative interactions. These methods must cluster negative and positive edges independently, and the information encoded in the mixture of positive and negative weights is lost. Third, our method is the second best performer for skewness score. This shows that despite fulfilling multiple summarization constraints, the clusters obtained have high skewness (*i.e.,* high edge weights) scores comparable to general graph clustering methods. Fourth, the 'node-based' decomposition in `MCL` do not admit overlapping genes. Consider for instance the subgraph $C3$ in Figure 3. If this subgraph is chosen as a cluster in `MCL`, then the subgraph $C4$ cannot be another cluster because of gene overlap. The 'edge-based' decomposition of `DiffNet`, which we argue is a more natural way of grouping interaction responses, does not suffer from this problem.

4.3 Effect of Parameter k

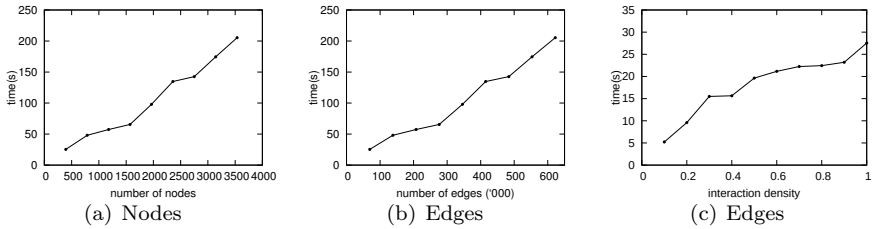
Figures 7(a)-(d) show the effect of k on summary coherence, skewness, coverage and distinctiveness. We observe that k controls the trade-off of summary coverage versus distinctiveness. The higher the value of k, the greater the coverage of functional subgraphs in the summary. However, the increase in coverage reduces the quality of the clusters (lower skewness, coherence and distinctiveness) due to the fact that one must now include lower quality clusters to satisfy the coverage requirement. Note that it is



(a) Coherence (b) Skewness

(c) Coverage (d) Distinctiveness

Figure 8. Effect of α on `DiffNet`.



(a) Nodes (b) Edges (c) Edges

Figure 9. Running time of `DiffNet`.

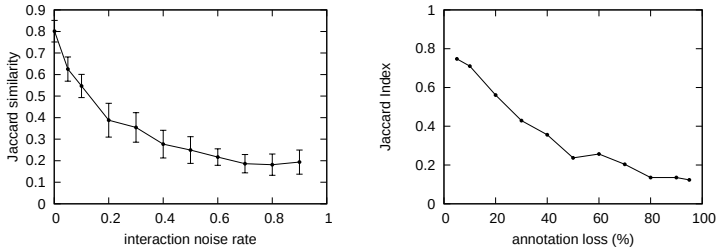
unrealistic to expect the majority of differential interactions to respond significantly to condition change. Thus, full coverage of all interaction responses, especially those that respond weakly, is typically not required in a differential summary.

4.4 Effect of Parameter α

Figures 8(a)-(d) show the effect of α on summary coherence, skewness, coverage and distinctiveness. We observe that α directly controls the influence of summary coherence. The higher the value of α, the greater the coherence of functional subgraphs in the summary. The increased coherence, however, comes at a cost. Coverage of the summary is reduced with greater α. This is because the increase penalty for choosing incoherent functional subgraphs reduces the exploration space during decomposition selection. Distinctiveness is also slightly increased with greater α.

4.5 Running Times

We generated synthetic networks by randomly adding nodes and edges to the [2] dataset network until the desired size is obtained. Figures 9(a)-(b) plot the running



(a) Interaction noise (b) Annotation loss

Figure 10. Effect of interaction noise and loss of annotations.

times of `DiffNet` of varying network sizes (viewed by number of nodes and edges, respectively). We observe that `DiffNet` scales almost linearly with the number of nodes and edges in the network. A differential network of 2500 nodes is summarized in less than 3 minutes. This shows that `DiffNet` constructs a summary within a reasonable time frame.

We further evaluated the running time of `DiffNet` at varying network density. Figure 9(c) shows the running time on [2] dataset network from 10% density (0.1) to full density (1.0). We artificially construct networks at varying density by randomly removing network edges until the desired density is achieved. From the figure, running time of `DiffNet` grows almost linearly with the network density.

4.6 Effect of Interaction Noise

Given that interaction profiles are likely to be noisy, we evaluate the effect of interaction noise on `DiffNet` summary construction. We assume the `DiffNet` summary generated from the differential network in [2] is without interaction noise and use it as the reference summary. We then simulate the effect of noise by perturbing the interactions of the network by random rearrangement of its interactions. The amount of perturbation is indicated by the *interaction noise rate*, which is the fraction of the original interactions that have been randomly rearranged. Figure 10(a) shows the stability of of the `DiffNet` summary after interaction noise perturbation. At each noise rate, we simulate 10 perturbed network samples. We compute the Jaccard similarity of the functional subgraphs of a perturbed summary ($\mathcal{S}_1$) against the reference summary ($\mathcal{S}_2$). Specifically $JaccardSimilarity(\mathcal{S}_1, \mathcal{S}_2) = 1$ if the gene set of each functional subgraph in $\mathcal{S}_1$ and $\mathcal{S}_2$ is identical. As expected, we observe a steady decrease in similarity against the reference summary with increasing interaction noise rate.

4.7 Effect of Annotation Loss

As current gene annotations are likely to be incomplete, here we study the effect of gradually removing gene annotations on `DiffNet` summary construction.

Suppose $\mathcal{S}_0$ is a reference `DiffNet` summary of the [2] differential network with complete gene annotations. We then constructed `DiffNet` summaries of differential networks with removed annotations and observed their similarities with the reference summary. Given two summaries $\mathcal{S}_1$ and $\mathcal{S}_2$, the similarity of the functional subgraphs

between the summaries can be measured using the following:

$$JaccardIndex(\mathcal{S}_1, \mathcal{S}_2) = \frac{1}{|\mathcal{S}_1|} \sum_{C_1 \in \mathcal{S}_1} \max_{C_2 \in \mathcal{S}_2} \frac{|V_1 \cap V_2|}{|V_1 \cup V_2|} \tag{9}$$

where $JaccardIndex(\mathcal{S}_1, \mathcal{S}_2) = 1$ if the gene set of each functional subgraph in $\mathcal{S}_1$ and $\mathcal{S}_2$ is identical. We removed $n\%$ of the gene annotations from the differential network and constructed a new summary $\mathcal{S}_n$. We call $\mathcal{S}_n$ a summary of the differential network with $n\%$ *annotation loss*. Figure 10(b) shows the $JaccardIndex$ similarities of summaries with varying annotation loss. We observe that annotation loss creates a summary that is increasingly different from the reference summary. The drop in $JaccardIndex$ similarity is gradual, suggesting that `DiffNet` summary construction is relatively robust to annotation noise. More importantly, as annotations of genes are likely to increase with time, it will only lead to more improved performance of `DiffNet`.

5 Conclusions

We propose `DiffNet`, a novel data-driven algorithm that automatically constructs summaries of differential functional responses of gene interaction networks under environment or condition change. Specifically, it leverages combination of GO annotation information and underlying interaction data to greedily identify a set of functional subgraphs that are highly skewed and coherent, representing significant functional responses due to condition change. Our empirical study with a real-world network revealed that `DiffNet` can automatically generate high quality differential functional summaries from the differential network including differential interactions that [2] failed to identify. Furthermore, we showed that state-of-the-art graph clustering algorithms cannot be adopted to generate such differential summaries. Currently, our approach primarily focuses on the `dE-MAP` data derived from `e-MAP`s. Thus, it cannot be applied to more than two treatments. As future work, we intend to explore the possibility of applying differential functional summarization on multiple conditions (> 2). Lastly, `DiffNet` is efficient and can generate differential summaries in acceptable time.

Observe that although `DiffNet` leverages GO terms for functional annotation, it does not exploit the hierarchical relationship between these terms. Hence, as part of future work we intend to extend `DiffNet` to incorporate such relationships and study its impact on the quality of the summaries. In summary, the results of this paper are an important first step in this regard.

References

[1] Bader, G. D. and Hogue, C. W. V. (2003). An automated method for finding molecular complexes in large protein interaction networks. *BMC bioinformatics*, **4**, 2.

[2] Bandyopadhyay, S., Mehta, M., Kuo, D., Sung, M.-K., Chuang, R., Jaehnig, E. J., Bodenmiller, B., Licon, K., Copeland, W., Shales, M., Fiedler, D., Dutkowski, J., Guénolé, A., van Attikum, H., Shokat, K. M., Kolodner, R. D., Huh, W.-K., Aebersold, R., Keogh, M.-C., Krogan, N. J., and Ideker, T. (2010). Rewiring of genetic networks in response to DNA damage. *Science (New York, N.Y.)*, **330**(6009), 1385–9.

[3] Boyle, E. I., Weng, S., Gollub, J., Jin, H., Botstein, D., Cherry, J. M., and Sherlock, G. (2004). GO::TermFinder–open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. *Bioinformatics (Oxford, England)*, **20**(18), 3710–5.

[4] Chvatal, V. (1979). A Greedy Heuristic for the Set-Covering Problem. *Mathematics of Operations Research*, **4**(3), 233–235.

[5] Collins, S. R., Schuldiner, M., Krogan, N. J., and Weissman, J. S. (2006). A strategy for extracting and analyzing large-scale quantitative epistatic interaction data. *Genome biology*, **7**(7), R63.

[6] 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**(7), 1575–84.

[7] Fabrizio, P., Pozza, F., Pletcher, S. D., Gendron, C. M., and Longo, V. D. (2001). Regulation of longevity and stress resistance by Sch9 in yeast. *Science (New York, N.Y.)*, **292**(5515), 288–90.

[8] Frey, B. J. and Dueck, D. (2007). Clustering by passing messages between data points. *Science (New York, N.Y.)*, **315**(5814), 972–6.

[9] Gillis, J., Mistry, M., and Pavlidis, P. (2010). Gene function analysis in complex data sets using ErmineJ. *Nature protocols*, **5**(6), 1148–59.

[10] Ideker, T. and Krogan, N. J. (2012). Differential network biology. *Molecular systems biology*, **8**, 565.

[11] Jain, A. and Mohiuddin, K. (1996). Artificial neural networks: a tutorial. *Computer*, **29**(3), 31–44.

[12] Kremer, B.E., Adang L.A., and Macara I.G. (2007). Septins regulate actin organization and cell-cycle arrest through nuclear accumulation of NCK mediated by SOCS7. *Cell*, **130**(5):837–50.

[13] Korte, B. and Vygen, J. (2012). Combinatorial Optimization: Theory and Algorithms (5 ed.) *Springer*, ISBN 978–3–642–24487–2.

[14] Nepusz, T., Yu, H., and Paccanaro, A. (2012). Detecting overlapping protein complexes in protein-protein interaction networks. *Nature methods*, **9**(5), 471–2.

[15] Przytycka, T. M., Singh, M., and Slonim, D. K. (2010). Toward the dynamic interactome: it's about time. *Briefings in bioinformatics*, **11**(1), 15–29.

[16] Rodriguez-Rocha, Garcia-Garcia A., Panayiotidis M.I., and Franco R (2011). DNA damage and autophagy. *Mutat Res.*, **711**(1-2):158–66.

[17] Schuldiner, M., Collins, S. R., Weissman, J. S., and Krogan, N. J. (2006). Quantitative genetic analysis in Saccharomyces cerevisiae using epistatic miniarray profiles (E-MAPs) and its application to chromatin functions. *Methods (San Diego, Calif.)*, **40**(4), 344–52.

[18] St Onge, R. P., Mani, R., Oh, J., Proctor, M., Fung, E., Davis, R. W., Nislow, C., Roth, F. P., and Giaever, G. (2007). Systematic pathway analysis using high-resolution fitness profiling of combinatorial gene deletions. *Nature genetics*, **39**(2), 199–206.

[19] Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, S. L., Golub, T. R., Lander, E. S., and Mesirov, J. P. (2005). Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *PNAS*, **102**(43), 15545–50.

[20] Subramanian, A., Kuehn, H., Gould, J., Tamayo, P., and Mesirov, J. P. (2007). GSEA-P: a desktop application for Gene Set Enrichment Analysis. *Bioinformatics (Oxford, England)*, **23**(23), 3251–3.

[21] Zhang, B., Park, B.-H., Karpinets, T., and Samatova, N. F. (2008). From pull-down data to protein interaction networks and complexes with biological relevance. *Bioinformatics (Oxford, England)*, **24**(7), 979–86.