Finding Compensatory Pathways in Yeast Genome

Olga Ohrimenko

Abstract

Pathways of genes found in protein interaction networks are used to establish a functional linkage between genes. A challenging problem in analysis of gene networks is to find pairs of compensatory pathways which can substitute for each other in case of a defective gene. All previous approaches on finding between-pathway models (BPMs) use the genetic network, while some of them also require each pathway to be a connected component in the physical network. In this thesis, we show that physical interactions can be used to induce missing genetic information and assist in finding BPMs. We extend the definition of a pair of compensatory pathways and propose a new method for finding BPMs which takes into account both networks. Experimental results on yeast interactome data set show that our method reveals more functionally enriched BPMs according to Gene Ontology database than previous work.

1 Introduction

Yeast genome is estimated to consist of 18.7% of essential genes, deletion of any of them leads to the death of a cell. Why can a cell survive a defect or deletion of a non-essential gene? This phenomenon is explained by the ability of a cell to buffer some of its functionality among several groups of genes. Such sets of genes involved in the same process that can compensate for each other, are called redundant pathways. Consider example in Figure 1, where blue arrows represent the information flow inside the cell. In the presence of redundant pathways 1 and 2, when a gene in pathway 1 is defected, pathway 2 is used instead, and vice versa. The natural question that arises is how to find pathways redundant in their functionality.

A synthetic lethal (SL) interaction between a pair of genes is present when deletion of both of the genes leads to a death of a cell, while a cell can survive a deletion of one of them. Furthermore, SL interaction usually exists between genes involved in the same or similar processes [9]. While this provides an inside look at buffering between genes, we are interested in whole pathways that can be removed and not affect cell viability. Kelley

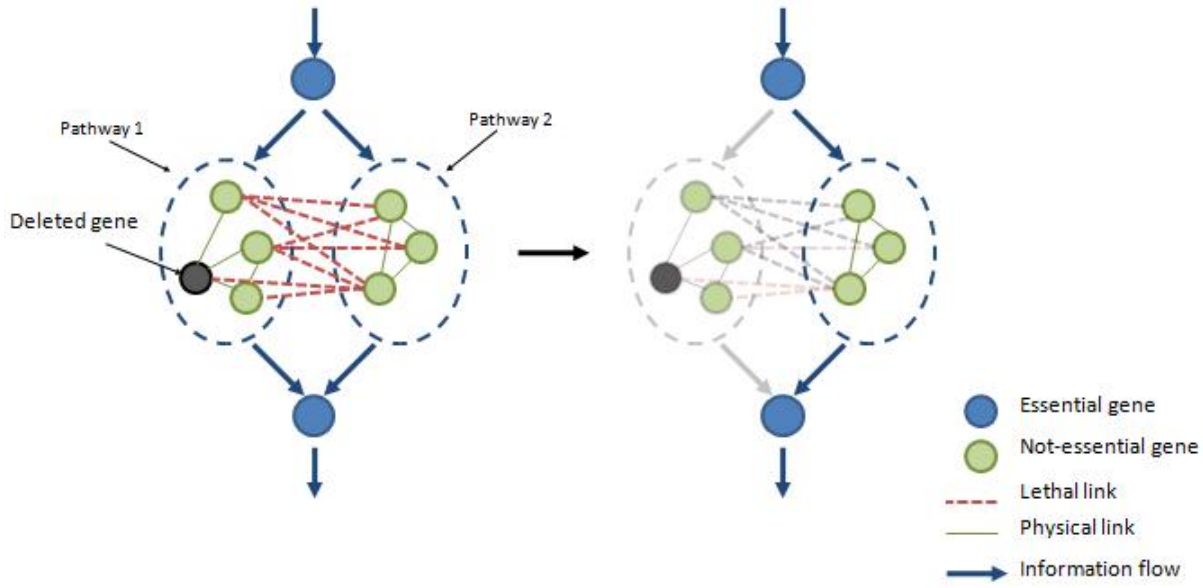


Figure 1: Between Pathway Model.

and Ideker [4] proposed to represent redundant pathways using a Between Pathway Model (BPM), a pair of pathways connected by SL interactions (red dashed links in Figure 1).

SL network comes from laboratory experiments with estimated 17–41% [8] false negatives. Since BPM model strongly relies on SL interactions inferring missing links from other types of data could improve the findings. Previous studies show that network of physical interactions (PI) among genes correlates with synthetic lethal network [6, 8]. For example a gene with many physical interactions has many SL interactions. In this paper, we show that physical interactions can be used to infer missing SL links. We then propose methods for finding BPMs based on this inference.

The layout of the paper is as follows. We first present the past work, followed by the notation in Section 3. We explain our methods in Section 4. Comparison of our methods with all known algorithms for finding BPMs is presented in Section 5.

2 Prior Work

We refer to synthetic lethal interactions as genetic interactions or lethal links, interchangeably.

Past work can be split into two parts: methods that use both SL and PI networks,

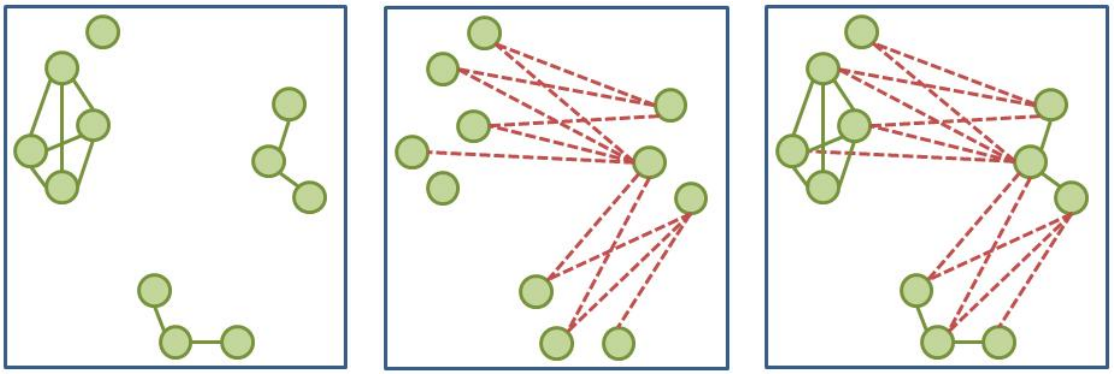
and methods that solely rely on genetic interactions, where each defines a BPM using assumptions about biological data.

Kelley and Ideker [4] were first to propose the Between Pathway Model and an algorithm for finding it. They defined a BPM to be a pair of physical pathways densely connected by lethal interactions, where each pathway is densely connected by physical interactions. They used a probabilistic method assigning each pair of pathways a score proportional to density of physical links within each pathway and density of genetic links between the pathways.

Shamir and Ulitsky [10] then extended this model requiring each pathway to be a connected component in the physical network. They explained more SL edges than Kelley and Ideker, since they relaxed the required connectivity of a BPM. There are two details in the methods by Kelley et al. and Shamir et al. that restrict the BPMs that can be found. First, both methods are initialized from small connected components which impose the physical and lethal interactions considered during search. Second, requiring BPMs to be fully or highly connected in a physical network relies on the fact that all physical interactions have been discovered.

The later work concentrated on SL network only, arguing that for some organisms physical network might not be available or contains many false positives and false negatives. Based on these assumptions Ma et al. [5] defined a BPM to be an approximately complete bipartite graph within the synthetic lethal interaction network. A similar work was done by Brady et al. [3] where they used the same definition of a BPM but proposed a greedy randomized algorithm which looks for stable bipartite subgraphs. Since neither of the methods required physical connectivity within pathways they found larger number of BPMs. However, the lack of knowledge about physical interactions resulted in lower functional homogeneity within the pathways.

The results from past work indicate that synthetic lethal interactions are a good source of data when looking for redundant pathways, while adding physical connections helps to find functional pathways. In this paper, we try to address the question of how to use physical links to find BPMs and not restrict the set of lethal links we consider.



(a) (b) (c)

Figure 2: (a) Physical interactions and (b) lethal interactions networks, where green solid edges represent physical links, and red dashed edges are synthetic lethal links. (c) Both networks combined.

3 Preliminaries

We can view each network as a graph, where a node represents a gene and an edge represents an interaction between corresponding genes. Let V be a set of yeast genes, E_L and E_P set of synthetic lethal and physical edges, correspondingly. Then, $G_L = (V, E_L)$ is a synthetic lethal network and $G_P = (V, E_P)$ is a network of physical interactions (Figure 2). Then a BPM $B = (P_1, P_2)$ consists of two sets of genes (pathways), s.t. $P_1 \cap P_2 = \emptyset$ and $P_1 \cup P_2 \subseteq V$. Figure 3 shows an example of a Between Pathway Model, with each pathway containing physical interactions, and 8 lethal links connecting them.

Using this notation we can summarize the properties of the BPMs found by past methods:

- Kelley and Ideker [4]: $B = (P_1, P_2)$ is a BPM if the number of lethal links between P_1 and P_2 is higher than expected on average, and each P_1 and P_2 are densely connected in V_P.
- Ulitsky and Shamir [10]: Same as above except P_1 and P_2 are required to be connected components in V_P.
- Ma et al. [5] try to maximize the size of each BPM ($|P_1| + |P_2|$) while maintaining many lethal edges between $P1$ and $P2$, and only a few inside of each pathway.

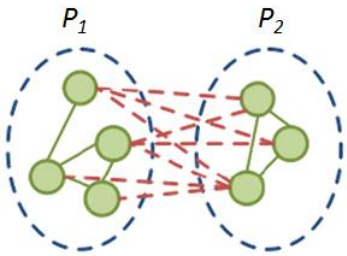


Figure 3: An example of redundant pathways P_1 and P_2.

3.1 Data

We analyze the network used by Kelley and Ideker [4] and Brady et al. [3][1]. We remove essential genes since we expect them to not appear inside of redundant pathways (see [10] for the discussion on essential genes inside of a BPM). After filtering the network contains 682 genes with 1858 synthetic lethal and 583 physical interactions (protein binding interactions).

3.2 Validation

We use FuncAssociate [2] web tool to analyze if pathways in a found BPM have a biological meaning. For each pathway FuncAssociate returns a set of Gene Ontology (GO) attributes with associated p-values. The lower the p-value the less likely it is that a pathway is annotated with this GO attribute by chance (see [2] for more details). We say that a BPM is *validated*, or *functionally homogeneous*, if it is annotated with at least one GO attribute with significance value is at least 0.01. We refer to GO attribute, as an attribute or function, interchangeably.

4 Methodology

We relax the definition of a BPM given by Kelley and Ideker [4]. We do not require pathways in a BPM to be connected components in G_P, due to the limitations it poses (see Section 2). We still use physical links in our algorithms but for a different purpose. All our methods are based on this definition.

We first discuss the similarities between the correlation clustering problem and finding

[1] `http://www.cellcircuits.org/Kelley2005/`

BPMs, and explain why the methods for solving the former cannot be used in the latter case. We then propose two novel approaches for building BPMs both centered on the idea of using the PI network to induce missing SL edges.

4.1 Correlation clustering

The problem of finding BPMs slightly resembles *correlation clustering* [1] (CC): given a set of data objects and information about which pairs of these objects can appear in the same cluster and which cannot, the objective is to cluster these objects satisfying as many of pair-wise restrictions as possible. We can view this problem as a graph with objects being the nodes, and two types of edges, a must-link edge for pairs of objects that have to appear in the same cluster, and cannot-link edged for pairs that have to be in different clusters. If we consider each pathway to be a cluster, lethal and physical links to be cannot-link and must-link edges, respectively, then our problem can be presented as correlation clustering. However, there are several distinctions.

A pathway in a BPM is a functional unit, and since a gene can be annotated with several functions it can participate in more than one pathway. Hence, pathways are not necessarily disjoint, whereas clustering procedure builds disjoint sets of points. Second, the data we work with is not complete, i.e. we have 1858 out of possible 232,221 edges in G_L, while most CC methods work with almost complete pair-wise information graphs. Another significant difference is the objective. We not only want to satisfy all must-link and cannot-link edges but also find pairs of clusters that are densely connected by cannot-link edges (each such pair is a BPM).

A greedy local search approach for solving the CC problem is to move a point to a cluster that would improve the objective the most, until no possible move results in a better objective value. We extend this greedy heuristic to overcome the aforementioned limitations of correlation clustering. We initialize each cluster using connected components of G_P, since we expect pathways to have physical interactions. This procedure resulted in only 30 clusters, one containing 1/3 of all genes. To increase the reliability on physical network we split clusters by removing articulation points and bridges. We are now left with 127 clusters, while the large cluster decreased in size it still remained significantly larger

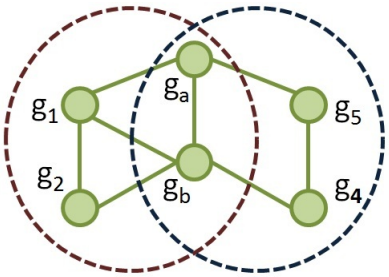


Figure 4: Genes g_a and g_b participate in two different biological processes (red and blue). However, it does not imply that g_1 and g_4 are from the same process.

than the rest of the clusters. We then extend the set of moves in our local search procedure to move, remove and replicate a gene to a cluster, forbidding moves which would violate must-link (physical) and cannot-link (lethal) edges. Note that a gene can be moved to a cluster with which it has no physical interactions (which is in line with our BPM definition). Our objective was to maximize the sum of edges between pairs of highly connected clusters.

The main shortcoming of the above approach is the presence of big clusters that biases the search since all other clusters tend to connect to it. We think that each of these highly connected components is an overlap of several pathways, where the physical interactions connecting each cluster are of different types and are not transitive (see Figure 4).

4.2 Ma et al. on extended G_L network

Synthetic lethal network contains many false negatives, which means that information about many lethal interactions is not present in the data either due to the nature or complexity of the conducted experiments. Due to existence of high correlation between physical and synthetic lethal networks [6, 8], we propose to use physical interactions to infer missing links in G_L.

We now explain how we can increase network G_L with edges inferred from graph G_P. Consider example in Figure 5, where genes g_1, g_2, g_3 and g_4 are all physically connected to each other, but only g_2, g_3 and g_4 are lethally connected to gene g_5. It is interesting that a group of first 4 genes has such a strong binding while only three of them are lethally connected to g_5. The intuition suggests that there must be a synthetic lethal edge present between g_1 ad g_5.

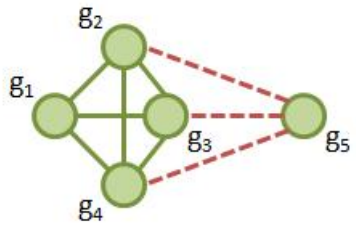


Figure 5: Inducing missing lethal link from physical links.

We create a new lethal network $G_{L,\beta} = (V, E_{L,\beta})$ as follows. For every $g \in V$ we find a set $P(g)$ of genes physically connected to g. Initially, $E_{L,\beta} = E_L$. We then add an edge (g, g') to $E_{L,\beta}$ if β fraction of genes in $P(g)$ have lethal interactions with g':

$$E_{L,\beta} \cup \{(g, g')\} \Leftrightarrow |\{p \mid \forall p \in P(g),\ (l', p) \in E_L\}| \geq \beta |P(g)| \wedge (g, g') \notin E_L.$$

Consider again example in Figure 5. Since all genes in $P(g_1) = \{g_2, g_3, g_4\}$ have lethal interactions with g_5, we add SL edge (g_1, g_5).

Using the above approach we created two networks $G_{L,1}$ and $G_{L,0.5}$ containing 146 and 264 new edges, respectively. To verify the edges that we added we consider a more recent network from BioGRID 2.0.31 [7]. We find that 46 out of 146 new edges in $G_{L,1}$ have been discovered in the recent version. This is a promising result, since it shows that links induced from a physical network were indeed missing.

This new combination of physical and lethal networks can now be used by algorithms that are based on lethal links only, i.e. [5] and [3], and **olya: provides** better comparison between different methods.

4.3 Inducing lethal interactions from PI network

We propose an algorithm for finding Between Pathway Models which is based on a finding from previous method. It is closely related to the algorithm by Ma et al. [5] but instead of using only lethal interactions to measure the connectivity between pathways we use both: physical and lethal interactions. The algorithm is presented in Figure 6. For a given lethal edge, it tries to grow a BPM (P_1, P_2) around this edge while the score between P_1 and P_2 is above some threshold $\mathcal{T}$. The score of each BPM is based on the number of lethal links

Algorithm *FindBPMs*(V, E_L, E_P)
Input: Set of yeast genes V;
Lethal interactions E_L;
Physical Interactions E_P;
Output: Set of Redundant Pathways; R

```
1.  R ←{}
2.  for (g, g') ∈ E_L
3.          P_1 ←{g}, P_2 ←{g'}
4.          while (true)
5.                  // S_1 and S_2 sets of genes that can be added to P_1 and P_2, respectively.
6.                  S_1 ←{v | ∀v ∈ V : ∃p ∈ P_2 ∧ (v,p) ∈ E_L ∨ ∃p ∈ P_1 ∧ (v,p) ∈ E_P}
7.                  S_2 ←... // Similar to S_1
8.                  if S_1 and S_2 are empty
9.                    then break
10.                 for v ∈ S_1
11.                         score[v] = Score(P_1 ∪ {v}, P2)
12.                 for v ∈ S_2
13.                         score[v] = Score(P_1, P_2 ∪ {v})
14.                 v ←argmax_v' score[v']
15.                 if score[v] < T break
16.                 if v ∈ S_1
17.                         P_1 ←P_1 ∪ {v}
18.                   else
19.                         P_2 ←P_2 ∪ {v}
20.         R ←R ∪ {(P_1, P_2)}
21. return R
```

Figure 6: Algorithm for finding redundant pathways.

between P_1 and P_2 out of total possible. Additionally, we consider the physical links within each pathway to derive the number of missing lethal links between P_1 and P_2. We derive $Score(P_1, P_2)$ as follows:

$$Score(P_1, P_2) = \frac{\sum_{i \in P_1, j \in P_2} conn(i,j)}{|P_1| \times |P_2|}$$

where

$$conn(i,j) = \begin{cases} 1 & \text{if } (i,j) \in E_L, \\ |\{g \mid \forall g \in P(i) : \ (g,j) \in E_L\}| \ / \ |P(i)| & \text{otherwise.} \end{cases}$$

Notice that the physical links contribute to the score, but only as support for missing links.

Example: Let the subnetwork in Figure 5 be a BPM, where $P_1 = \{g_1, g_2, g_3, g_4\}$ and

$P_2 = \{g_5\}$. Then $conn(g_i, g_5) = 1$, for $1 \leq i \leq 4$, and:

$$Score(P_1, P_2) = \frac{1+1+1+1}{4 \times 1} = 1$$

We set $\mathcal{T}$ to 0.75, since Ma et al. [5] used the same level. We omit the check for level of violation of lethal links within each pathway from Figure 6.

5 Results

We compared our method with 4 previously known algorithms for finding redundant pathways. For each method we remove overlapping BPMs as done in [4, 5, 10]: if two BPMs share more than 50% synthetic lethal interactions, we remove the smaller one. The results are presented in Table 1. The BPM algorithms are split into groups depending on what network they use (please find specifics of each method in Sections 2 and 3). We are interested in finding BPMs that contain functionally homogeneous pathways. The columns 4 and 5 show the number of BPMs with one or both pathways validated for at least one attribute in GO database, while the sixth column presents how many contained pathways enriched for the same function.

We first run Brady et al. [3] and Ma et al. [5] which are developed to use only lethal interactions, G_L (row 1 in Table 1). We see that Ma et al. finds a slightly larger number of BPMs, while a much higher percentage of them is validated. We then analyze the result of Ma et al. on the extended networks that we built in Section 4.2, $G_{L,1}$ and $G_{L,0.5}$ (rows 2–3 in Table 1). We observe that the number of found BPMs and the number of validated pathways within them increases as we extend the original network with induced missing links. We attribute this result to the addition of missing links to the semi-dense parts of the G_L network which improves the connectivity of the subnetworks that were previously below the threshold level.

We now analyze results on the network containing both physical and lethal interactions, G_P+G_L. Kelley and Ideker [4] and Ulitsky and Shamir [10] require high connectivity in G_P and, hence, produce a small number of BPMs but most of them are validated. Both [3]

Network	Method	BPMs found (%validated)	Validated Pathways			Ave BPM size
			P_1 or P_2	P_1 and P_2	Same func.	
G_L	Brady et al.	139 [2](66%)	44	48	31	29.15
	Ma et al.	147 (96%)	29	112	44	10.24
$G_{L,1}$	Ma et al.	171 (96%)	33	132	46	10.78
$G_{L,0.5}$	Ma et al.	217 (**97%**)	51	159	56	11.38
G_P+G_L	Kelley et al.	20 (95%)	1	18	4	8.4
	Shamir et al.	16 (100%)	0	16	8	25.43
	Ours	267 (**97%**)	**58**	**201**	**75**	11.14

Table 1: Numerical Results on Finding BPMs.

and [5] find more redundant pathways as only SL network is considered. Although our method is similar to [5], connectivity of our BPMs and the scoring function we use rely on both networks without being as strict on connectivity in G_P as [4] and [10]. Overall, our method finds the most number of validated BPMs with 75 of them validated with the same GO attribute. The improvement produced from our BPM scoring model is related to the result on $G_{L,\beta}$ network, since both take into consideration the missing lethal edges inferred from the physical interactions.

6 Conclusion

We present two methods for finding Between Pathway Models. The first method extends the synthetic lethal interaction network with missing links, which are inferred from physical interactions network. We find that 1/3 of predicted missing links are present in recent version of SL database. Moreover, SL-based method finds more validated pathways on extended network than on the original one, since now it indirectly uses PI network. Based on this finding, we propose an algorithm with a novel BPM scoring model that takes into account both physical and lethal interactions. Our approach finds more validated BPMs than previously known methods. Experimental results from both of our methods show that physical interactions are indeed useful when looking for BPMs. Since the data we used contains many false positives and false negatives, the analysis of another type of data on yeast, e.g. gene expression data, is left for our future work.

[2]Brady et al. [3] do not remove overlapping pathways, hence they report 602 pathways, instead of 139.

References

[1] A. Ben-Dor, R. Shamir, and Z. Yakhini. Clustering gene expression patterns. *Journal of Computational Biology*, 6(3/4):281–297, 1999.

[2] G. F. Berriz, O. D. King, B. Bryant, et al. Characterizing gene sets with FuncAssociate. *Bioinformatics*, 19:2502–2504, 2003.

[3] A. Brady, K. Maxwell, N. Daniels, and L. J. Cowen. Fault tolerance in protein interaction networks: Stable bipartite subgraphs and redundant pathways. *PLoS ONE*, 4 (4):e5364, April 2009.

[4] R. Kelley and T. Ideker. Systematic interpretation of genetic interactions using protein networks. *Nat Biotech*, 23(5):561–566, May 2005.

[5] X. Ma, A. M. Tarone, and W. Li. Mapping genetically compensatory pathways from synthetic lethal interactions in yeast. *PLoS ONE*, 3(4):e1922, April 2008.

[6] O. Ozier, N. Amin, and T. Ideker. Global architecture of genetic interactions on the protein network. *Nat Biotech*, 21(5):490–491, 05 2003.

[7] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, et al. BioGRID: a general repository for interaction datasets . *Nucleic Acids Research*, 34:D535–539, 2006.

[8] A. H. Y. Tong, G. Lesage, G. D. Bader, et al. Global mapping of the yeast genetic interaction network. *Science*, 303(5659):808–813, 2004.

[9] C. L. Tucker and S. Fields. Lethal combinations. *Nature Genetics*, 35:204–205, 2003.

[10] I. Ulitsky and R. Shamir. Pathway redundancy and protein essentiality revealed in the Saccharomyces cerevisiae interaction networks. *Mol Syst Biol*, 3(104):561–566, April 2007.