Overcoming Drug Resistance by Co-Targeting

Marzieh Ayati[1], Golnaz Taheri[1#], Shahriar Arab[2], Limsoon Wong[3], Changiz Eslahchi[4,5*]

1. Department of Computer Science, Sharif University of Technology, Iran
2. School of Computer Science, Institute for Research in Fundamental Sciences (IPM), Iran
3. Department of Computer Science, National University of Singapore, Singapore
4. Department of Mathematics, Shahid Beheshti University, Iran
5. Center of Excellence in Biomathematics, School of Mathematics, Statistics and Computer Science, University of Tehran, Tehran, Iran

Abstract

Removal or suppression of key proteins in an essential pathway of a pathogen is expected to disrupt the pathway and prohibit the pathogen from performing a vital function. Thus disconnecting multiple essential pathways should disrupt the survival of a pathogen even when it has multiple pathways to drug resistance. We consider a scenario where the drug-resistance pathways are unknown. To disrupt these pathways, we consider a cut set S of G, where G is a connected simple graph representing the protein interaction network of the pathogen, so that G-S splits to two partitions such that the endpoints of each pathway are in different partitions. If the difference between the sizes of the two partitions is high, the probability of existence of a functioning pathway in one partition is increased. Thus, we need to partition the graph into two balanced partitions. We approximate the balanced bipartitioning problem with spectral bipartitioning since finding (2,1)-separator is NP-complete. We test our technique on E. coli and C. jejuni. We show that over 50% of genes in the cut sets are essential. Moreover, all proteins in the cut sets have fundamental roles in cell and inhibition of each of them is harmful for cell survival. Also, 20% and 17% of known targets are in the vertex cut of E.coli and C.jejuni. Hence our approach has produced plausible "co-targets" whose inhibition should counter a pathogen's drug resistance.

1. Introduction[1]

There is a critical need to address the emergence of drug resistant pathogens for several infectious diseases. For example, drug-resistant tuberculosis has continued to spread internationally and is now approaching critical proportions. Approaches to counter drug resistance have so far achieved limited success [1]. It has been proposed that this lack of success is due to a lack of understanding of how resistance emerges in bacteria upon drug treatment and that a systems-level analysis of the proteins and interactions involved is essential to gaining insights into the routes required for drug resistance.

A recent idea for a systems-level analysis is the so-called "co-targets" of a drug [2]. Typically, the intended target of a drug, referred to as the "primary target", is inhibited and the pathogen is killed or its growth is arrested. However, the pathogen sometimes develops resistance to the drug. Communication to the resistance machinery---e.g., efflux pumps and drug-modifying enzymes---is established via pathways in the protein interaction and gene regulation networks. Proteins mediating such communication are termed "co-targets". The inhibition of these co-targets, and thus partitioning the implicated pathways into disconnected pieces, should effectively disrupt the survival of the pathogen [3].

We consider the basic scenario where we do not know the likely drug-escape mechanisms of the pathogen. So we try to disrupt the maximum number of pathways, including all target and back-up (drug-escape) paths, by identifying as small a set of proteins (the "co-targets") to inhibit as many pathways as possible. The partitioning of multiple essential pathways into disconnected pieces should effectively disrupt the survival of a bacterium even when it has multiple pathways to drug resistance[3]. This problem was considered in a similar way in 2010 by Hormozdiari et al[4]. We explore here the computational problem of efficiently identifying a minimum set of co-targets to disable drug-resistance in a pathogen as described above. We test our technique on *E. coli* and *C. jejuni*.

2. The Problem

The input is a protein interaction network (maybe augmented with gene regulation interactions) of the pathogen expressed as a simple connected graph $G = \langle V, E \rangle$. The nodes V correspond to proteins and edges E correspond to interactions of proteins. In our scenario, targets and drug-escape pathways are not known. To disrupt these pathways, we consider a cut set S of G, such that G-S splits into two partitions and the endpoints of each pathway are in different partitions. If the difference between the sizes of the

[1] # The first and second author contributed equally
* Corresponding author ch-eslahchi@sbu.ac.ir

two partitions is high, the probability of existence of a pathway in one partition is increased. Therefore, we partition the graph into two balanced partitions by removing the minimum number of vertices and disrupt the maximum number of pathways including unknown back-up pathways. The problem is thus to pick a subset of nodes S such that (i) S is as small as possible and (ii) the removal of S, partitions G into sub graphs G1 and G2 and (iii) |G1|/|G2| is as close to 1 as possible.

Definition. Let G = ⟨V, E⟩ be a simple connected graph. A partition P is called (k, 1+ε)-separator if P partitions V into k disjoint subsets each containing at most (1+ε) n/k vertices. We know that finding (k, α)-separator is NP-Complete [5].

It is also well known that the (2,1)-separator problem, in other words the Minimum Bisection problem, is NP-complete[6]. Hence, in the next section, we introduce an approximation algorithm to find a (2, 1)-separator in a graph G.

4. Algorithm

We approximate the balanced bipartitioning problem with spectral bipartitioning. Spectral partitioning is based on eigenvectors of Laplace eigenvalues of graphs [7]. It is one of the most successful heuristic approaches in the design of partition algorithms. This spectral method recursively bisects a graph by considering an eigenvector of a Laplacian matrix to gain an understanding of global properties of the graph.

We set up the problem as follows. Define the graph G by vertex set V and edge set E. Let V= $\{v_1, v_2,..,v_n\}$ be the vertex set of graph G and e_{ij} in E denotes the edge between v_i and v_j . Now we want to assign a variable x_i to each v_i such that the following conditions are satisfied:

$$x_i = \pm 1 \text{ and } |\sum_{Vi} x_i| \leq 1 \qquad (*)$$

We call a vector $x=(x_1,...,x_n)$ whose elements satisfy (*) conditions an indicator vector. Let M and N denot the two sets of vertices with $x_i = +1$ and $x_j = -1$ assigned to them respectively. In the next step, we try to minimize $f(x) = 1/4 \sum_{eij} (x_i - x_j)^2$ by using laplacian matrix L(G) of G by the method mentioned by Simon [8]. If u_1, u_2 ,…, u_n are the normalized eigenvectors of L(G) with corresponding eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n$, it is known that $x = \sqrt{n}\, u_2$ is a good approximation to minimize f(x). Bispectral partitioning is based on this property of λ_2.

In the case of bisection, we find the median of the u_2 values and then map vertices with corresponding u_2 above the median to one set, and those below to the other. This solution is the nearest discrete point to the continuous optimum.

An immediate corollary of the reasoning used to solve the continuous minimization problem is that $n\lambda_2$ /4 is a lower bound on the number of cuts produced by any balanced partitioning of the graph, because the solution space of the continuous problem contains the solution space of the discrete problem. This can be slightly improved using insight from higher dimensional partitioning schemes.

Our algorithm is O(nlogn) to sort the elements of the eigen vector and sort the eigenvalues of the matrix to find minimum eigenvalue.

The rest of the algorithm is, using the set of resulting cut edges, to identify a subset S of the vertex set of graph G satisfying the following conditions:

1- For each cut edge e=ab, a ∈ S or b ∈ S.
2- G-S has two component G_1 and G_2 such that $|G_1|-|G_2|<2$.
3- |S| is as minimal as possible

To find the set S, let A= $\{e_1=a_1b_2, \dots, e_k=a_kb_k\}$ be the set of cut edges found by the above algorithm. We form a bipartite graph H induced by A. (i.e. the graph consisting of these edges and their end point).

Any vertex cover of this induced graph (i.e. a set of vertices such that every edge has at least one endpoint in the set) will serve as a S. Here, we sort the vertices of H corresponding to their degrees. We choose vertices with respect to their degree alternatively till all of the edges in cut are covered. Since we choose alternatively, two partitions are still balanced. It is obvious that the set of S which is obtained from graph H is vertex cut of graph G

The pseudo codes of our algorithm are presented in the following table.

Algorithm

Step 0. Compute Laplacian matrix of the graph.

Step 1. Compute eigenvalues of Laplacian graph.

Step 2. Compute eigenvector of Laplacian accor-ding to second smallest eigenvalue and sort its elements.

Step 3. Insert half of the vertices in partition A and remainder in partition B. The edges between these two partitions are in edge cut.

Step 4. Find vertex cut: (i) Sort vertices according to their degrees. (ii) Insert vertices as vertex cut alternatively until all edge cuts are covered.

---

5. Experiments

We validate our scenario and algorithm using the protein interaction network of *E. coli* and *C. jejuni*. *C.jejuni* is a Gram-negative food-borne pathogen that is a major cause of gastroenteritis in humans[9]. In our method we use the *C. jejuni* in file 7 from [10]. The interaction map in this data set includes a large subnetwork of putative essential genes that may be used to identify potential new antimicrobial drug targets for *C. jejuni* and related organisms. On the other hand, *E. coli* is the only bacterium for which protein complex purifications have been applied at the proteome scale. So, we used a set of high-confidence *E. coli* protein interactions from literature-cited low-throughput experiments compiled in the Database of Interacting Proteins (DIP) [11].
The dataset of *C. jejuni* has 139 vertices and 379 edges. The size of the vertex cut and partitions are 13, 64 and 61, respectively. The dataset of *E. coli* has 165 vertices and 333 edges ,and a cut vertex set of size 17 partitions the graph in two subsets of size 72 and 76. ~ 60% of the vertices have degree less than 5 in *C.jejuni* and 3 in *E.coli*. While the vertices in the cut sets are in the top 40% of high-degree vertices in both data sets. According to the distribution of the degree of the vertices and the large number of vertices with small degree, the selected vertex cut in both dataset seems to be a good approximation for minimum cut. Jeong *et al.* [12] demonstrated that high-degree nodes or hubs in a protein interaction network of *S. cerevisiae* contain more essential proteins than would be expected by chance. Since then the correlation between degree and essentiality was confirmed by other studies [13-16], but until recently there was no systematic attempt to examine the reasons for this correlation. Jeong *et al.* [12] suggested that overrepresentation of essential proteins among high-degree nodes can be attributed to the central role of hubs in mediating interactions among numerous less-connected proteins. However, Zotenko *et al* found that node degree is a better predictor of essentiality than other measures tested. Since then Zotenko *et al* [17] checked the relationship between degree and essentiality in six networks and they showed that essential genes do not always have a high degree. Note that our algorithm is not concentrated on high-degree nodes. Hence, it is consistent with their result. In Table 1, according to relation between degree and essentiality, our algorithm can identify essential vertices with low degree in the network. Zotenko *et al* [17] found that while in some networks high-degree nodes are as important in maintaining the overall network connectivity as nodes having high betweenness centrality values, this property is not due to essential proteins. In particular, even though removing nodes with high betweenness centrality indices is much more effective in shattering some of our protein interaction networks, their correlation with essentiality is reduced to statistically insignificant levels by subtracting their correlation with degree centrality. Table 1 also matches well- with the results of Zotenko *et al*. This table also describes some biological properties of the elements of vertex cut. The first property is essentiality. Essential genes are those indispensable for the survival of an organism. Therefore, the essential genes are good targets. The next property is functional role and biological process which shows the function of proteins in cell. Over 50% of genes in the cut vertex set are essential. The essentiality information of each gene was retrieved from DEG (Database of Essential Genes – http://tubic.tju.edu.cn/deg) [18]. Some proteins in the cut vertex set are located in cell membrane. Due to their accessibility, they are good candidate targets. It is noticeable that all proteins in the cut vertex sets have fundamental roles in cell and inhibition of each of them is harmful for cell survival. So corresponding to the diseases, we can choose specific targets from the cut. Also, according to the antibacterial targets for enteric bacteria retrieved from www.drugbank.ca, 20% and 17% of known targets are in the vertex cut in *E. coli* and *C. jejuni* respectively [19-20]. Thus the selected vertices have good properties in biological and graph theory aspects.

6. Discussion

By removing the proteins that exist in a cut vertex set, the connections between some proteins are disconnected as the graph is divided in two separate parts. A pathway will be disrupted if their endpoints are in different parts. However, partitioning graph into a big part vs. a small part is likely to cause the two endpoints of a pathway to lie in the big one. We want to find targets that disrupt more pathways. So to decrease the probability of two endpoints of a pathway landing in one partition, we partition the graph into two balanced parts. And to decrease the minimum number of proteins to be targeted, we find a good approximation of minimum cut. This problem was also studied by Hormozdiari et al [4]. Their algorithm is based on β-balance partitioning in weighted graph. So they found a cut set and components which have different size. In this work we introduced an algorithm which returns balance partition with similar cut size.
Suppressing an essential gene in a pathogen can induce an irreparable damage to the pathogen, provided there is no alternative "back-up" pathway for the pathogen to counter or escape this effect. If a drug fails, and there is no mutation in the drug targets in the pathogen, then it is likely that some back-up pathways exist. So we need to cut off these escape

routes. This requires additional proteins to be targeted (i.e. the co-targets). Co-targets are protein(s) that need to be simultaneously inhibited along with the intended target(s), to check emergence of resistance to a given drug [2].

If back-up pathways are known, suppressing them disrupts the viability of pathogen. On the other hand, as in the scenario of our paper, the target and drug-escape pathways are often not known. So our algorithm tries to disrupt the maximum number of pathways. Consequently, the number of genes introduced as co-targets are many.

Our algorithm can be easily adapted for the scenario where back-up pathways are known. The cut vertices produced by our algorithm destroy the maximum number of pathways, if we simultaneously inhibit them. The known target pathway and its backup pathways should be among those pathways affected by the cut vertices produced by our algorithm. So we can choose what genes of these known back-up pathways to co-target from the cut set. Obviously, the upper bound on the number of co-targets is reduced to the number of known backup pathways in this scenario. Moreover, we can filter the PPI network according to type of drugs (such as different kind of antibiotics) as follows:

1. We select all the proteins which could be affected directly by one of the specific drugs.
2. We also keep the proteins which are in the middle of the shortest paths between the selected proteins on the step 1 (as done in [2]).
3. We run our algorithm on the resulting network.
4. The vertices in the cut set are (co-)target for the considered drugs.

Also, in this scenario one can choose a subset of vertex cut as drug target genes with respect to essentiality, cellular position and functional role of the genes. Besides, we can compute the betweenness of the known drug target in our PPI network and substitute that drug target for the vertex in the cut with a lower betweenness. Then the vertex cut would destroy more paths (back-up pathways). For example, we can substitute the proteins with low betweenness in cut with rplD having betweenness 20 and 12 in *E. coli* and *C. jejuni* (shown in Tables 6 and 7). According to the algorithm, most of the hub vertices are in the cut set. So the chance of these proteins working well as target or co-target is higher than the others. Furthermore, most of the cut vertices are essential, which is a good reason to conclude that these vertices would be good candidate (co-)targets.

Acknowledgments. This work was supported in part by the IPM, Iran; and the Singapore Ministry of Education tier-2 grant MOE2009-T2-2-004.

References

1. Johnson R, Streicher E.M, et al.: **Drug resistance in mycobacterium tuberculosis**. *Curr Issues Mol Biol*, 8:97-111, 2006.

2. Raman K, Chandra N: **Mycobacterium tuberculosis interactome analysis unravels potential pathways to drug resistance**. *BMC Microbiology*, 8:234, 2008.

3. Strong M, Eisenberg D: **The protein network as a tool for finding novel drug targets**. *Progress in Drug Research*, 64:191-215, 2007.

4. Hormozdiari F, Salari R, et al: **Protein-protein interaction network evaluation for identifying potential drug targets**. *J Comput Biol.,* 17(5):669-684, 2010.

5. Konstantin A, Räcke H.: **Balanced graph partitions**. *Proc 16th SPAA*, pp. 120-124, 2004.

6. Garey M.R, Johnson D.S: **Computers and intractability: a guide to the theory of NP-completeness**. W. H. Freeman and Co., San Francisco, Calif., 1979.

7. Gleich D, et al: **Specteral graph partitioning,** http://www.users.cselabs.umn.edu/classes/Fall2008/csci8363/TALKS/Yun-graphpart2.pdf.

8. Simon H.D: **Partitioning of unstructured problems for parallel processing.** *Proc .Conf Parallel Methods on Large Scale Structural Analysis and Physics Applications*, 1991.

9. Blaser M.J: **Epidemiologic and clinical features of Campylobacter jejuni infections**. *J Infect Dis*, 176**:**S103-S105, 1997.

10. Parrish J.R, Yu J, et al.: **A proteome-wide protein interaction map for campylobacter jejuni**. *Genome Biol*, 8:130, 2007.

11. Xenarios I, Rice D.W, et al: **DIP: the database of interacting proteins**. *Nucleic Acids Res*, 28:289-291, 2000.

12. Jeong H, Mason S.P, et al: **Lethality and centrality in protein networks**. *Nature*, 411:41-42, 2001.

13. Batada N.N, Hurst L.D, Tyers M: **Evolutionary and physiological importance of hub proteins**. *PLoS Comput Biol*, 2:E88, 2006.

14. Hahn M.W, Kern A.D: **Comparative genomics of centrality and essentiality in three eukaryotic protein-interaction networks**. *Mol Biol Evol*, 22:803-806, 2005.

15. Yu H, Kim P.M, et al: **The importance of bottlenecks in protein networks: correlation with gene essentiality and expression dynamics**. *PLoS Comput Biol*, 3:E59, 2007.

16. Yu H, Greenbaum D, Xin L.H, et al: **Genomic analysis of essentiality within protein networks**. *Trends Genet*, 20:227-231, 2004.

17. Zotenko E, Mestre J, et al: **Why do hubs in the yeast protein interaction network tend to be essential: reexamining the connection between the network topology and essentiality?**. *PLoS Comput Biol*, 4:e1000140, 2008.

18. Zhang R, Zhang C.T: **DEG: a database of essential genes**. *Nucleic Acids Res*, 32:D271-D272, 2004.

19. Wishart D.S, Knox C, et al: **a knowledgebase for drugs, drug actions and drug targets**. *Nucleic Acids Res*., 36:D901-D906, 2008.

20. Wishart D.S, Knox C, et al: **a comprehensive resource for in silico drug discovery and exploration**. *Nucleic Acids Res*., 34:D668-D672, 2006.

Table 1. Cut genes and their essentiality and functional role in *C. jejuni* and *E.coli.*

C. jejuni				E .coli			
Locus name	**Gene name**	**Essential Gene /Degree /Betweenness**	**Functional role**	**Locus name**	**Gene name**	**Essential Gene /Degree /Betweenness**	**Functional role**
Cj1182c	rpsB	**E/26/40**	30S ribosomal protein S2	**P36979**	yfgB	**N/19/49**	Hypothetical protein yfgB
Cj1708c	rpsJ	**E/25/72**	30S ribosomal protein S10	**P08330**	prsA	**E/17/48**	Ribose-phosphate pyrophosphokinase
Cj0576	lpxD	**E/16/27**	UDP-3-O-[3-hydroxymyristoyl] glucosamine N-acyltransferase	**P02364**	rpsJ	**E/13/15**	30S ribosomal protein S10
Cj1713	-	**N/23/45**	putative radical SAM domain protein	**P14825**	lysU	**N/12/5**	Lysyl-tRNA synthetase, heat inducible
Cj0918c	prsA	**E/16/27**	ribose-phosphate pyrophosphokinase	**P02351**	rpsB	**E/13/15**	30S ribosomal protein S2
Cj1594	rpsD	**E/13/8**	30S ribosomal protein S4	**P32168**	hslU	**N/8/4**	ATP-dependent hsl protease ATP-binding subunit hslU
Cj0596	peb4-cbf2	**N/9/6**	major antigenic peptide PEB-cell binding factor	**P02423**	rplV	**N/6/3**	50S ribosomal protein L22
Cj1206c	ftsY	**E/8/8**	putative signal recognition particle protein	**P06961**	cca	**E/11/12**	tRNA nucleotidyltransferase
Cj1037c	pycA	**N/6/5**	pyruvate carboxylase A subunit	**P10440**	lpxA	**E/10/24**	Acyl-[acyl-carrier-protein]--UDP-N-acetylglucosamine O-acyltransferase
Cj0509c	clpB	**N/3/2**	ATP-dependent Clp protease ATP-binding subunit	**P04475**	dnaK	**E/5/2**	Chaperone protein dnaK
Cj0893c	rpsA	**E/5/3**	30S ribosomal protein S1	**P21645**	lpxD	**E/4/2**	UDP-3-O-[3-hydroxymyristoyl] glucosamine N-acyltransferase
Cj0493	fusA	**E/5/1**	elongation factor G	**P02392**	rplL	**E/4/2**	50S ribosomal protein L7/L12
				P03017	recA	**N/3/1**	RecA protein
				P37181	hybC	**N/2/1**	Hydrogenase-2 large chain precursor
				P13519	mreB	**E/3/2**	Rod shape-determining protein mreB
				P02905	accB	**E/4/2**	Biotin carboxyl carrier protein of acetyl-CoA carboxylase

Table2. The properties of drug targets in *C.Jejuni* and *E.coli*

E.coli			***C.jejuni***		
Gene	**Degree**	**Betweenness**	**Gene**	**Degree**	**Betweenness**
rplV(in cut)	3	1	**rpsD(in cut)**	5	1
murA, rplJ, rplD, rpsD	5,5,12,5	0,2,20,0	**gyrA, ileS, rplD, murA, rplJ**	1,1,12,5,5	0,0,8,0,0