The Most Reliable Subgraph Problem

Petteri Hintsanen

HIIT Basic Research Unit, Department of Computer Science,
PO Box 68, FI-00014 University of Helsinki, Finland
petteri.hintsanen@cs.helsinki.fi

Abstract. We introduce the problem of finding the most reliable subgraph: given a probabilistic graph G subject to random edge failures, a set of terminal vertices, and an integer K, find a subgraph $H \subset G$ having K fewer edges than G, such that the probability of connecting the terminals in H is maximized. The solution has applications in link analysis and visualization. We begin by formally defining the problem in a general form, after which we focus on a two-terminal, undirected case. Although the problem is most likely computationally intractable, we give a polynomial-time algorithm for a special case where G is series-parallel. For the general case, we propose a computationally efficient greedy heuristic. Our experiments on simulated graphs illustrate the usefulness of the concept of most reliable subgraph, and suggest that the heuristic for the general case is quite competitive.

1 Introduction

Many contemporary domains in data mining have heterogeneous objects linked together by various relations. Graphs are natural models for data arising from such domains; for example, social networks and the World Wide Web can be naturally described as graphs. In this article we consider probabilistic graphs, whose edges are unreliable and can fail with specified probabilities. Telecommunications and electrical networks are classical examples of real-world structures often modeled as probabilistic graphs.

Informally, given a probabilistic graph and a set of terminal vertices, the reliability of the graph is the probability that there exists at least one path between all pairs of terminals at the time of inspection. It is easy to see that some edges can be more important for the existence of a connection than others. For example, edges forming a cut between two terminals cannot fail simultaneously without breaking connections between those terminals. A natural question to ask is which edges contribute most to the reliability, or equivalently, which edges are safest to remove without a significant loss of reliability? We formulate this question as the problem of finding the most reliable subgraph: given a probabilistic graph, a set of terminal vertices, and an integer K, what is the optimal way to remove K edges from the graph, such that the remaining graph has maximum reliability?

A solution to this problem can be readily applied to a variety of network problems. Consider, for example, a telecommunications network, where edges

J.N. Kok et al. (Eds.): PKDD 2007, LNAI 4702, pp. 471–478, 2007.

represent links between communicating parties, and are subject to random malfunctions. The most reliable subgraph between two communicating terminals describes the most reliable channels for exchanging messages. In social networks, where relative mutual acquaintances could be represented as edge probabilities, one can discover most important relationships between specified individuals by finding a reliable subgraph connecting them. Reliable subgraphs are also useful when visualizing large graphs: they can be highlighted in a picture, or extracted altogether for visual inspection.

A closely related concept of *connection subgraph* was recently introduced by Faloutsos and others [1]. They formalized the *connection subgraph problem*: given a weighted graph, two vertices s and t, and an integer k, find a k-vertex subgraph containing s and t which maximizes a given goodness function. Our framework has a similar goal, but is based on probabilistic reasoning and is defined for multiple terminal vertices.

Overall, little has been published on the extraction and analysis of general connection subgraphs. Lin and Chalupsky use rarity of simple paths and cycles for evaluating the novelty and interestingness of links [2]. Following Faloutsos et al., Ramakrishnan and others propose a method for extracting informative connection subgraphs from RDF graphs [3]. There are a couple of methods utilizing network reliability. One is described by Asthana et al., who predict protein complex memberships in a network of protein interactions [4]. Sevon et al. use network reliability for evaluating the connection strength between entities in biological graphs [5]. Finally, De Raedt et al. consider compression of probabilistic first-order theories and their uses for link discovery in biological networks [6].

Network reliability, on the other hand, has been under extensive research. A canonical summary is given by Colbourn [7]. However, we have been unable to find any references to our problem from the vast literature on reliability theory. Closest effort into this direction seems to be Birnbaum's classical text on *reliability importance*, which measures the importance of a single edge for the reliability of a graph [8]. It has been extended for pairs of edges by Hong and Lie [9]. Finally, Page and Perry consider the reliability importance for ranking the edges of a given graph [10].

2 Problem Definition and Complexity

We use a standard probabilistic graph model. Let $G = (V, E)$ be a graph with a vertex set V and an edge set E. Edges are unreliable: each edge $e \in E$ has an associated probability p_e for functioning; conversely, each edge can fail with probability $1 - p_e$. Edge failures are assumed to be independent. On the other hand, vertices are expected to be fully reliable, that is, they do not fail.

Let G be an undirected probabilistic graph, and let $U \subset V$ be a set of k terminal vertices or nodes. We review the six classical reliability measures, following Colbourn [7]. First, *k-terminal reliability* Rel_k is defined as the probability that each of the k terminal nodes in U can communicate in G; equivalently, it is the probability that there exists a path between any pair of terminals.

When $k = 2$, this measure is referred to as *two-terminal network reliability*, while the case $k = |V|$ is known as *all-terminal network reliability*. We denote these measures by Rel_2 and Rel_A, respectively. (We omit explicit references to G and U whenever they are clear from the context.)

These measures have natural counterparts for directed probabilistic graphs. One vertex $s \in U$ is chosen as the source node, and the rest of the vertices of U are target nodes. The directed version of Rel_k, known as *s,T-connectedness* or Conn_k, is the probability that there exists a (directed) path from s to all target nodes. When $k = 2$, this measure is called *s,t-connectedness* or Conn_2. Finally, the directed analogue of Rel_A is known as *reachability* or Conn_A.

The objective in the *most reliable subgraph problem* (MRSP) is to find the most reliable subgraph obtained from G by removing exactly K edges:

Definition 1 (The Most Reliable Subgraph Problem). *Let $G = (V, E)$ be a probabilistic graph, and let $U \subset V$ be a set of k terminal vertices, where $2 \leq k \leq |V|$. Let $f \in \{\mathrm{Rel}_2, \mathrm{Rel}_k, \mathrm{Rel}_\mathrm{A}, \mathrm{Conn}_2, \mathrm{Conn}_k, \mathrm{Conn}_\mathrm{A}\}$ be the corresponding reliability measure with respect to U, and let $K \in \mathbb{N}$ with $0 \leq K \leq |E|$. The objective is to find a subgraph $H \subset G$ with $|E| - K$ edges, such that $f(H) \geq f(H')$ for all subgraphs $H' \subset G$ having $|E| - K$ edges.*

Given the fact that exact calculations of $\mathrm{Rel}_{\{2,k,A\}}$ and $\mathrm{Conn}_{\{2,k,A\}}$ are #P-complete problems [11], it is not surprising that the MRSP is likely to be computationally hard as well. The problem does not ask for the value of $f(H)$ for the chosen f and an optimal subgraph H, so the MRSP could be in that sense easier than computing the reliability. Despite this relaxation, it is easy to see that the k-terminal undirected MRSP is NP-hard:

Theorem 1. *MRSP with $f = \mathrm{Rel}_k$ is NP-hard.*

Proof. We give a polynomial time reduction from the NP-complete Steiner Tree problem [12] to the MRSP. Let (G, U, B) be an instance of Steiner Tree, where $G = (V, E)$ is a graph with positive edge weights, $U \subset V$ is a set of terminals, and $B \in \mathbb{N}$ is a bound for the size of the tree.

Without a loss of generality we assume that all edge weights are equal to 1. We transform G into a probabilistic graph $H = (V, E)$ by setting $p_e = 1/2$ for each $e \in E$. Next, we find the smallest (that is, having the least number of vertices and edges) optimal subgraph $H^* \subset H$ connecting the terminals, by solving the MRSP for $K = 0, \ldots, |E| - |U| + 1$ and checking the results in polynomial time. Obviously H^* is a tree; it is also a minimal Steiner tree. Assume to the contrary that there exists a minimal Steiner tree T such that $\|T\| < \|H^*\|$, where $\| \cdot \|$ denotes the number of edges. By construction, we have $\mathrm{Rel}_k(T) = 1/2^{\|T\|} > 1/2^{\|H^*\|} = \mathrm{Rel}_k(H^*)$, which contradicts the optimality of H^*, since T is also a subgraph of H connecting the vertices in U.

To complete the reduction, we simply check if $\|H^*\| \leq B$ holds. □

The complexity of cases where $f \in \{\mathrm{Rel}_2, \mathrm{Rel}_\mathrm{A}\}$ remains open, but we conjecture that they are also NP-hard. The directed variants of the problem are probably hard too, considering the fact that the directed reliability problems are as hard as the corresponding undirected problems [13].

3 Algorithms

It is most likely that there is no efficient algorithm for solving the MRSP in a general case. However, we next describe a polynomial-time algorithm for solving the two-terminal undirected MRSP in an important special case, where the graph is series-parallel. We then give a computationally efficient greedy heuristic for the MRSP in a general case.

3.1 Series-Parallel Graphs

The class of (edge) series-parallel graphs is usually defined using series and parallel composition rules [14]. For our purposes, the following equivalent definition is better: a probabilistic graph G with specified terminals s and t is *series-parallel*, if it can be reduced into a single edge (s, t) by repeatedly applying the following reductions:

- *Series reduction*: If G has a vertex $v \notin \{s, t\}$ of degree two, v and its adjacent edges $e = (u, v)$ and $f = (v, w)$ can be replaced with a single edge $g = (u, w)$ with $p_g = p_e p_f$.
- *Parallel reduction*: If G has two parallel edges $e = (u, v)$ and $f = (u, v)$, they can be replaced with a single edge $g = (u, v)$ with $p_g = 1 - (1 - p_e)(1 - p_f)$.

The specific sequence of reductions is irrelevant, that is, if reductions are applied in any order until no reduction is possible, the result is the single edge (s, t) [14].

Before we describe the algorithm, let us introduce some terminology and notation. For an arbitrary edge set $F \subset E$, let $G[F]$ be the subgraph edge-induced by F. We denote the set of edges reduced into an edge e by $S(e)$; i.e. $f \in S(e)$, if f occurs in the sequence of series-parallel reductions that produced e. Initially, we let $S(e) = \{e\}$ for each $e \in E$.

Let $e = (u, v) \in E$. An i-edge subset $S(e, i) \subset S(e)$ is said to be an *optimal solution for* $G[S(e)]$, if $G[S(e) - S(e, i)]$ is the most reliable subgraph of $G[S(e)]$ with $|S(e)| - i$ edges and terminals u and v. In other words, $G[S(e) - S(e, i)]$ is a solution to the MRSP for $G[S(e)]$ with $K = i$ and $U = \{u, v\}$. Let $S_{\mathrm{R}}(e, i)$ be the reliability of an optimal solution $S(e, i)$, i.e. $S_{\mathrm{R}}(e, i) = \mathrm{Rel}_2\big(G[S(e) - S(e, i)]\big)$.

The iterative definition of series-parallel graphs suggests an iterative, dynamic programming algorithm for solving the MRSP, given that an optimal solution can be constructed from optimal solutions to smaller subgraphs. The following lemma states that this is indeed the case.

Lemma 1. *Let e and f be two edges in series or parallel, and let $S(e, i)$, $S(f, i)$ be optimal solutions for $G[S(e)]$ and $G[S(f)]$, where $0 \le i \le K$. Optimal solutions $S(g, i)$, for all i, can be formed in $O(K^2)$ time, where g is the edge produced by the reduction of e and f.*

Proof. Let i be fixed. Since $S(g) = S(e) \cup S(f)$ and $S(e) \cap S(f) = \emptyset$, an optimal solution $S(g,i)$ has exactly j edges in $S(e)$ and $i-j$ edges in $S(f)$, where $0 \leq j \leq i$. If e and f are in series, we have $S_{\mathrm{R}}(g,i) = S_{\mathrm{R}}(e,j) \cdot S_{\mathrm{R}}(f,i-j)$. Otherwise e and f are parallel, and we have $S_{\mathrm{R}}(g,i) = 1 - \bigl(1 - S_{\mathrm{R}}(e,j)\bigr) \cdot \bigl(1 - S_{\mathrm{R}}(f,i-j)\bigr)$. An optimal solution can be found by simply enumerating all i possible combinations of edge assignments and choosing one which maximizes $S_{\mathrm{R}}(g,i)$:

$$k = \operatorname*{arg\,max}_{0 \leq j \leq i} \begin{cases} S_{\mathrm{R}}(e,j) \cdot S_{\mathrm{R}}(f,i-j) & \text{if } e \text{ and } f \text{ are in series} \\ 1 - \bigl(1 - S_{\mathrm{R}}(e,j)\bigr) \cdot \bigl(1 - S_{\mathrm{R}}(f,i-j)\bigr) & \text{if } e \text{ and } f \text{ are parallel} \end{cases}$$

$$S(g,i) = S(e,k) \cup S(f,i-k) \ .$$

The solution can be found in $O(i)$ time. By repeating the procedure for all i, $0 \leq i \leq K$, we obtain the solutions $S(g,i)$ in $O(K^2)$ time. □

To solve the MRSP for a series-parallel graph G, we repeatedly apply series and parallel reductions until the graph is reduced into a single edge. As initialization, let $S(e,0) = \emptyset$, $S_{\mathrm{R}}(e,0) = p_e$, $S(e,1) = \{e\}$, and $S_{\mathrm{R}}(e,1) = 0$ for each $e \in E$. This establishes an invariant: each e has optimal solutions $S(e,i)$ for $G[S(e)]$, where $i = 0, \ldots, \min\{|S(e)|, K\}$. We maintain the invariant by keeping track of optimal solutions $S(e,i)$ and their reliabilities, for each remaining edge e. The invariant, with the definition of series-parallel graphs, guarantees that in the end we have an optimal solution to the MRSP for G.

At the beginning of each iteration, we identify a pair $\{e,f\}$ of reducible edges. This can be done in constant time by suitably augmenting the graph data structure. These edges are replaced with a new edge g; by Lemma 1 it is straightforward to form optimal solutions $S(g,i)$, thus maintaining the invariant. (Note that some combinations stated in Lemma 1 are undefined when $S(e,i)$ or $S(f,i)$ are available only for small i, however, these special cases can be easily detected.) Since each reduction effectively removes one edge from G, after $|E|-1$ iterations only a single edge e remains, and $S(e,K)$ contains an optimal solution for G. Putting the pieces together, we have established the following theorem.

Theorem 2. *Let $G = (V,E)$ be a series-parallel probabilistic graph. The MRSP for G can be solved in $O(K^2|E|)$ time, where $1 \leq K \leq |E|$.*

3.2 General Graphs

The set of series-parallel graphs is a very restricted class of graphs; in general, graphs are lot more complex. Unfortunately, as suggested in Sect. 2, the computational effort required for an exact solution quickly becomes excessive. It is most likely that one must content with approximate or heuristic solutions.

We next describe a simple, greedy heuristic for solving the MRSP on general graphs. The heuristic is based on a well-known Monte-Carlo (MC) simulation procedure, which is in many cases sufficient for approximating the reliability of a probabilistic graph G: one just simulates random edge failures in G by flipping a suitably biased coin for each edge, and checks if the terminals are connected in the resulting graph. By counting the number of positive outcomes (i.e. there is a connection between the terminals in that particular outcome) over many repetitions, the reliability estimate is then the fraction of positive outcomes out of the total number of simulations. If the reliability is not very low, then with a reasonable number of simulations we have a good estimate with high probability [15].

The MC procedure is not directly suitable for the MRSP, due to the large number of possible solutions (subgraphs) to consider. However, we use it to estimate $\mathrm{Rel}_2(G-e)$ for each $e \in E$, where $G-e$ denotes G with an edge e removed. Intuitively, edges with large values of $\mathrm{Rel}_2(G-e)$ are less critical, so we iteratively remove K edges with the highest $\mathrm{Rel}_2(G-e)$ values. If, at the beginning of an iteration, there are edges with an endpoint of degree one, we remove those edges first. Such edges are irrelevant from the reliability's standpoint, since they do not occur on any acyclic path between terminals. This heuristic can be implemented in a straightforward manner to run in $O\big(N|E|^2 + K(|E| + \log|E|)\big)$ time. We emphasize the computational efficiency of the heuristic, suggesting that it is suitable for interactive use such as visualization.

4 Experiments

Series-parallel graphs are practicable for evaluating the usefulness of the concept of most reliable subgraph and the relative performance of the proposed heuristic, since they are easy to generate with controlled parameters (size and reliability). Furthermore, we can efficiently calculate optimal solutions with the algorithm described in Sect. 3.1.

We generated nine datasets of random series-parallel multigraphs by repeatedly applying the series-parallel composition rules. Each set consists of 50 graphs with the same number of edges and the same reliability. The sizes are 50, 100 and 200 edges, and the reliabilities are 0.25, 0.5 and 0.75. These parameter choices give nine possible combinations, one for each dataset. The given sizes are averages: there is a slight random variation in the number of edges, because we reduced the parallel edges from the generated multigraphs in order to obtain proper probabilistic graphs.

To evaluate the approximation quality of the heuristic, we used it to solve the MRSP on the generated graphs, with different values of K. After this, the reliabilities of the results were estimated. Optimal solutions were calculated using the algorithm of Sect. 3.1. In order to assess the effect of using $\mathrm{Rel}_2(G-e)$ values to control the heuristic, we implemented a baseline heuristic. This heuristic is identical to the proposed heuristic with the exception that instead of using $\mathrm{Rel}_2(G-e)$ values to decide which edges to remove, it simply considers edge probabilities in ascending order. All estimates were done with 1,000,000 MC simulations. To control random variation, we report the average performance of each method over the 50 graphs in each dataset.

The results for two datasets are depicted in Fig. 1; in the remaining cases the results were comparable. From Fig. 1, we see that the relative performance (the estimated reliability of the subgraph produced by the heuristic divided by the reliability of the optimal subgraph) is fairly stable over different values of K. This is in contrast to the baseline heuristic, whose performance gets significantly worse as K grows. The results suggest that the proposed heuristic is quite competitive; in most cases, the mean relative performance over the different values of K is close to 85%.

The usefulness of most reliable subgraph is also observable in Fig. 1. In both cases, over half of the edges could be removed without a significant loss of reliability. The resulting small, reliable subgraphs contain the most critical edges for the connection, and could be used as a starting point for further analysis or visualization. Our preliminary experiments with real biological graphs show similar or even more accentuated effect (results not shown).

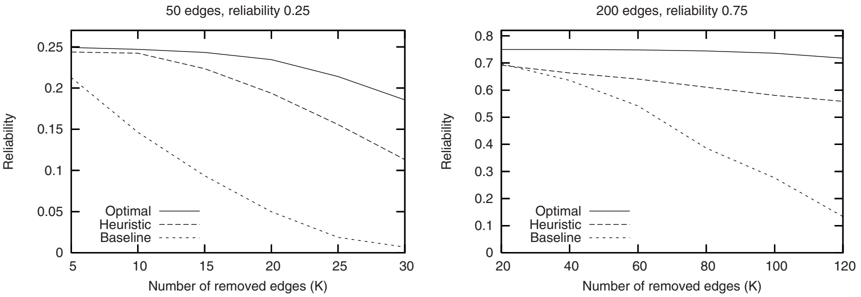

Fig. 1. Results for two generated datasets

5 Conclusions

As more and more domains of interest are best described as interlinked heterogeneous objects, we can expect graphs to become the data models of choice in many situations [16]. Applications may demonstrate a degree of randomness on links, e.g. technical unreliability, subjective uncertainty, or relevance with respect to a specific task. Probabilistic graphs are useful models in such cases.

In the light of these observations, novel graph mining concepts and methods are essential for coping with the increasing number of graph mining problems. The concept of most reliable subgraph, the associated most reliable subgraph problem, and the analysis of the problem are novel additions to this setting. We believe that the concept is useful in many data mining challenges on probabilistic graphs.

We described efficient methods for solving the MRSP, and demonstrated their usefulness with experimental results on synthetic probabilistic graphs. Future work will include improving the methods and assessing their performance with more varied and extensive experiments. There are also open questions on the complexity, and on the other variants of the MRSP.

Despite the apparent usefulness of the concept of most reliable subgraph, we were surprised to found out that (to the best of our knowledge) there is practically no previous research on the subject. Therefore, our proposed methods for solving the MRSP could be seen as the first steps toward more efficient and robust algorithms.

Acknowledgements. The author is grateful to Hannu Toivonen, Kimmo Kulovesi, Petteri Sevon, and Lauri Eronen for constructive discussions on the subject, and for numerous valuable improvements to the manuscript. This research has been supported by Tekes, Jurilab Ltd., Biocomputing Platforms Ltd., and GeneOS Ltd.

References

1. Faloutsos, C., McCurley, K.S., Tomkins, A.: Fast discovery of connection subgraphs. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 118–127. ACM Press, New York (2004)
2. Lin, S., Chalupsky, H.: Unsupervised link discovery in multi-relational data via rarity analysis. In: Proceedings of the Third IEEE International Conference on Data Mining, pp. 171–178. IEEE Computer Society Press, Los Alamitos (2003)
3. Ramakrishnan, C., Milnor, W.H., Perry, M., Sheth, A.P.: Discovering informative connection subgraphs in multi-relational graphs. SIGKDD Explorations 7, 56–63 (2005)
4. Asthana, S., King, O.D., Gibbons, F.D., Roth, F.P.: Predicting protein complex membership using probabilistic network reliability. Genome Research 14, 1170–1175 (2004)
5. Sevon, P., Eronen, L., Hintsanen, P., Kulovesi, K., Toivonen, H.: Link discovery in graphs derived from biological databases. In: Proceedings of Data Integration in the Life Sciences, Third International Workshop, pp. 35–49 (2006)
6. De Raedt, L., Kersting, K., Kimmig, A., Revoredo, K., Toivonen, H.: Compressing probabilistic Prolog programs (Submitted)
7. Colbourn, C.J.: The Combinatorics of Network Reliability. Oxford University Press, Oxford (1987)
8. Birnbaum, Z.W.: On the importance of different components in a multicomponent system. In: Multivariate Analysis - II, pp. 581–592 (1969)
9. Hong, J., Lie, C.: Joint reliability-importance of two edges in an undirected network. IEEE Transactions on Reliability 42, 17–33 (1993)
10. Page, L.B., Perry, J.E.: Reliability polynomials and link importance in networks. IEEE Transactions on Reliability 43, 51–58 (1994)
11. Valiant, L.G.: The complexity of enumeration and reliability problems. SIAM Journal on Computing 8, 410–421 (1979)
12. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman and Company (1979)
13. Ball, M.O.: Complexity of network reliability computations. Networks 10, 153–165 (1980)
14. Valdes, J., Tarjan, R.E., Lawler, E.L.: The recognition of series-parallel digraphs. SIAM Journal on Computing 11, 298–313 (1982)
15. Karp, R.M., Luby, M., Madras, N.: Monte-Carlo approximation algorithms for enumeration problems. Journal of Algorithms 10, 429–449 (1989)
16. Getoor, L., Diehl, C.P.: Link mining: A survey. SIGKDD Explorations 7, 3–12 (2005)