Region-Based Connectivity - A New Paradigm for Design of Fault-tolerant Networks

Arunabha Sen, Sudheendra Murthy, Sujogya Banerjee
Department of Computer Science and Engineering
Arizona State University
Tempe, Arizona 85281
Email: {asen, sudhi, sujogya}@asu.edu

***Abstract*—The studies in fault-tolerance in networks mostly focus on the *connectivity* of the graph as the metric of fault-tolerance. If the underlying graph is *k-connected,* it can tolerate up to $k-1$ failures. In measuring the fault tolerance in terms of connectivity, no assumption regarding the locations of the faulty nodes are made - the failed nodes may be close to each other or far from each other. In other words, the connectivity metric has no way of capturing the notion of *locality* of faults. However in many networks, faults may be highly *localized*. This is particularly true in military networks, where an enemy bomb may inflict *massive* but *localized* damage to the network. To capture the notion of locality of faults in a network, a new metric *region-based connectivity* (RBC) was introduced in [1]. It was shown that RBC can achieve the same level of fault-tolerance as the metric connectivity, with much lower networking resources. The study in [1] was restricted to *single region fault model* (SRFM), where faults are confined to one region only. In this paper, we extend the notion of RBC to *multiple region fault model* (MRFM), where faults are no longer confined to a single region. As faults in MRFM are still confined to regions, albeit multiple of them, it is different from unconstrained fault model where no constraint on locality of faults is imposed. The MRFM leads to several new concepts, such as *region-disjoint paths* and *region cuts*. We show that the classical result, the maximum number of node-disjoint paths between a pair of nodes is equal to the minimum number of nodes whose removal disconnects the pair, is no longer valid when region-disjoint paths and region cuts are considered. We show that the problems of finding (i) the maximum number of region-disjoint paths between a pair of nodes, and (ii) minimum number of regions whose removal disconnect a pair of nodes, are both NP-complete. We provide heuristic solution to these two problems and evaluate their efficacy by comparing the results with optimal solutions. We also discuss the network design problem with minimum resources that guarantees a specified value of the RBC in MRFM.**

I. Introduction

Fault-tolerant design is important for any kind of networks - *wired* or *wireless*, *static* or *mobile*. Accordingly, over the years a multitude of papers have appeared in the literature detailing different aspects of fault-tolerance in networks [2]–[7]. Most of the studies on fault-tolerance in networks accept *connectivity* of the graph as the metric of fault-tolerance. If the underlying graph is *k-connected*, it can tolerate up to $k-1$ failures. In measuring fault tolerance in terms of connectivity, no assumption regarding the locations of the faulty nodes is made - the failed nodes may be close to each other or far from each other. In fact, the metric connectivity has no way of capturing the notion of *locality* of faults. However, in many networks, faults may be highly localized. This is particularly true in a military situation, where an enemy bomb can destroy a large number of nodes (may be deployed sensors, armored vehicles or dismounted soldiers). An example of localized failure is shown in Figure 1, where the faults are confined to the shaded regions. The localized fault scenario is applicable not only in military situation but also in civilian environment. A prime example of such localized communication network failure was encountered on September 11, 2001 when attacks were launched against the Twin Towers of the World Trade Center. Although the resulting failure was *massive* in nature, the faults were *localized*, i.e., confined to New York and Washington D.C. area only.

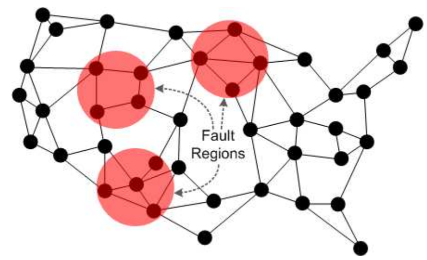


Fig. 1. A network with multiple fault regions

To capture the notion of locality in fault-tolerance capability of a network, a new metric *region-based connectivity* (RBC) was introduced in [1]. Obviously, the notion of RBC is tied to the notion of a *region*. A region may be defined in several different ways and they are discussed in detail in section III of the paper. Region based connectivity may be considered under two fault models - (i) Single Region Fault Model (SRFM) where faults are confined to one region only and (ii) Multiple Region Fault Model (MRFM) where faults are confined to k regions for some specified integer k. The results presented in [1] are limited to RBC in SFRM only. It was shown in [1] that the connectivity and the region-based connectivity of a graph in SRFM can be widely different. Moreover, RBC as a metric for fault-tolerance can achieve the same level of robustness as

This research was supported in part by ARO grant W911NF-06-1-0354. The information reported here does not reflect the position or the policy of the federal government.

the metric connectivity with much less networking resources. In this paper we extend results of the same vein to MRFM. In case it is known that the faults will be localized to at most k regions and each region can have at most p nodes, we argue that the focus should be on the design of networks that can tolerate failure of k regions rather than failure of $k \times p$ nodes. Since the number of nodes in each region is unlikely to be uniformly distributed, many regions will have much fewer than p nodes. Accordingly, a design to tolerate $k \times p$ node failures results in a network that is equipped to handle events that will never take place.

The contribution of this paper may be listed as follows: (i) extension of the notion of RBC from SRFM to MRFM, (ii) introduction of the notion of *region-disjoint paths* (RDP) and *region cuts* (RC), (iii) demonstration of invalidity of the Menger's theorem with respect to maximum RDP and minimum RC, (iv) presentation of NP-completeness proof of the minimum RC problem and the maximum RDP problem, (v) development of heuristics for minimum RC and maximum RDP problems, (vi) demonstration of effectiveness of the heuristics through extensive simulation, (vii) demonstration of cost effectiveness (in terms of power consumption) of RBC than the conventional metric in achieving the same level of fault-tolerance in the network.

II. Related Work

Fault-tolerance and connectivity of communication networks formed by the multihop packet radio networks have been studied by many researchers [2]–[7]. The authors in [3] were among the early researchers who studied the relationship between the density of the radio transceivers in the geographic region, their radius of transmission and the connectivity of the network formed with these two parameters. Research along this line was further refined in [6], [7], where the authors established the conditions for the radio network being connected (with high probability) for a given transmission range and distribution density of the transceivers. In [8], the authors studied the minimum transmit power required to maintain connectivity of the network when each node can independently choose a transmission power level. Utilizing some fundamental results presented by Penrose in [5], Bettstetter [2] developed an analytical expression for computing the range of the transceivers (on the assumption that they are uniformly distributed in a plane) so that the resulting network is k-connected with high probability.

In addition to the efforts of the networking research community, the issues related to the connectivity of graphs have been extensively studied by mathematicians with an interest in graph theory. Many different variations of connectivity have been studied, e.g. *distance connectivity* [9], *average connectivity*, *line connectivity*, *l-connectivity*, *supper connectivity*, *path connectivity*, *algebraic connectivity* [10], *cyclic connectivity* [11] and *conditional connectivity* [12]. The concept of *local connectivity*, studied in [13], at a first glance may appear to be similar to the notion of *region-based connectivity* introduced in [1]. However, on closer examination one can find out that the concepts are totally different. To the best of our knowledge, neither the networking research community nor the graph theory research community have studied any notion comparable to the notion of *region-based connectivity* in MRFM being proposed in this paper.

III. Region-Based Connectivity

In addition to the topology (graph) of the network, we may also have a *layout* of the network. We refer to the *layout* of the network as the *network geometry*. A *region* may be defined either with reference to the network topology or with reference to the network geometry.

- Diameter-based region: A diameter-based region in a network graph $G = (V, E)$ for a given d is defined as a *maximal* induced subgraph of G with diameter[1] d.
- Radius-based region: A radius-based region in a network graph $G = (V, E)$ for a given r is defined as a *maximal* induced subgraph of G containing node v and its r-hop neighborhood. We call node v as the center of such a region.
- Geometry-based region: With reference to the network geometry, a geometry-based region for a given r is defined as a collection of nodes and links covered by a circular area of radius r in the network layout[2].

As discussed earlier, RBC may be considered under two fault models - (i) Single Region Fault Model (SRFM) where faults are confined to one region only and (ii) Multiple Region Fault Model (MRFM) where faults are confined to k regions for some specified integer k. As results related to SRFM are available in [1], we focus on the MRFM after a brief introduction to RBC in SRFM.

A. Region-based Connectivity in Single Region Fault Model

Formally, in SRFM, the *single-region-based (node) connectivity* of graph G with a specified definition of region R, $s\kappa_R(G)$, is defined as follows: Suppose that $\{R_1, \ldots, R_k\}$ is the set of all possible regions of the graph G. Consider a k-dimensional vector T whose i-th entry, $T[i]$, indicates the number of nodes in region R_i whose failure will disconnect the graph G. If the graph G remains connected even after the failure of all nodes of the region R_i, then $T[i]$ is set equal to ∞. Then the *single-region-based connectivity* of a graph G with region R is $s\kappa_R(G) = \min_{1 \le i \le k} T[i]$.

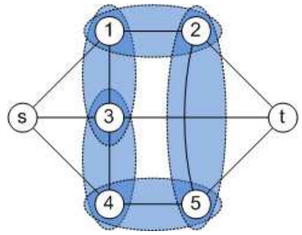
B. Region-based Connectivity in Multiple Region Fault Model

In MRFM, the *multi-region-based (node[3]) connectivity* of graph G with a specified definition of region R, $m\kappa_R(G)$, is defined as minimum number of regions whose removal (i.e., removal of all nodes in the regions and edges incident on them) will disconnect the graph. Analogous to the notion of

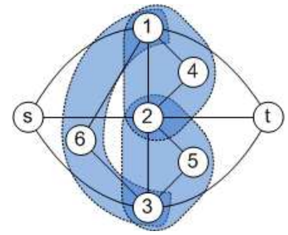
[1]The diameter of a graph is the longest shortest path between any two graph nodes

[2]We assume the locations of the nodes are given in a 2-dimensional Euclidean space.

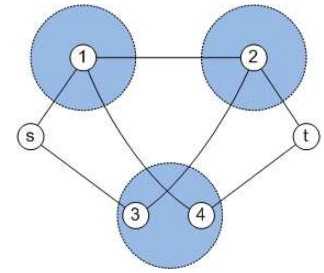
[3]The terms *node* and *vertex* are used interchangeably in this paper.



(a) Diameter-based region definition (Maximal subgraphs of diameter $d = 1$): Set of regions excluding regions containing the source and destination are R_1={1, 2}, R_2={1, 3}, R_3={3, 4}, R_4={4, 5}, R_5={2, 5}. Here, $m\kappa_R(G)$={R_2, R_3}, but $|\mathcal{P}| = 1$



(b) Radius-based region definition (Maximal subgraphs containing a vertex and its r=1-hop neighborhood): Set of regions excluding regions containing the source and destination are $R_1 = \{1, 2, 4\}$, $R_2 = \{2, 3, 5\}$, $R_3 = \{1, 3, 6\}$. Here, $m\kappa_R(G)$={R_1, R_2}, but $|\mathcal{P}| = 1$



(c) Geometry-based region definition (Circular area representing geometric regions): Set of regions excluding regions containing the source and destination are $R_1 = \{1\}$, $R_2 = \{2\}$, $R_3 = \{3, 4\}$. Here, $m\kappa_R(G)$={R_1, R_2}, but $|\mathcal{P}| = 1$

Fig. 2. Example graphs in which size of minimum region cut $m\kappa_R(G)$ may not be equal to maximum number of region-disjoint paths $\mathcal{P}$ for different definitions of region

vertex cut[4] in graphs, we define a *region cut* of a graph as a set of regions whose removal disconnects the graph. It is to be noted that the different regions in a graph may overlap with each other. In particular, a node can be part of multiple regions. We define the *region set of a node* v denoted by $R(v)$, as the set of regions that node v belongs to. The *region set of a path* P denoted by $R(P)$ is defined as the union of the region sets of the nodes of P. Given two nodes s, t in $G = (V, E)$, two paths P_1 and P_2 between s and t are called as *region-disjoint paths* if the union of regions associated with the internal nodes of P_1 and the union of regions associated with the internal nodes of P_2 are disjoint. (internal nodes of P_1 and P_2 are the nodes of P_1 and P_2 excluding the start node s and the termination node t). In other words, P_1 and P_2 are *region-disjoint* if sets $R(P_1) \setminus \{R(s) \cup R(t)\}$ and $R(P_2) \setminus \{R(s) \cup R(t)\}$ are disjoint.

One classical result of Graph Theory is *Menger's Theorem (1927)* which states that if u and v are two nodes in a graph $G = (V, E)$, then the *minimum* number of nodes whose removal disconnects u and v in G is equal to the *maximum* number of node-disjoint paths between u and v [14]. With the introduction of the notion of region-disjoint paths, a fundamental research question arises: *Does Menger's Theorem hold for region-based connectivity?* More specifically the question is: *Is the* minimum *number of* regions *whose removal disconnects* u *and* v *in* G*, equal to the* maximum *number of* region *disjoint paths between* u *and* v? We show that the answer to this question is negative through a few illustrative examples shown in Figure 2.

Claim : For a variety of definitions of a "region", the minimum number of regions whose removal disconnects the nodes s and t in G (i.e., the region cut RC between s and t), is not equal to the maximum number of region-disjoint paths (RDP) between s and t. Moreover, the difference between minimum RC and maximum RDP between s and t can be arbitrarily large.

[4]A vertex cut of a graph $G = (V, E)$ is a subset of nodes whose removal disconnects G.

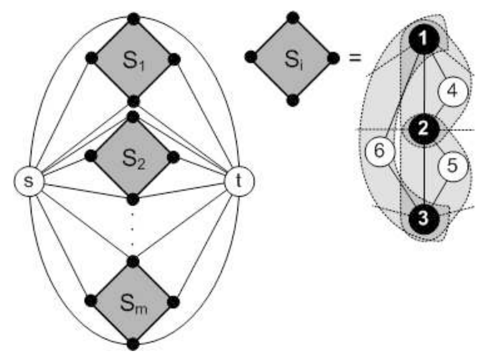


Fig. 3. A network with RDP and RC equal to m and $2m$ respectively (c) An example of a cost effective design using RBC in MRFM

Proof: Let $\mathcal{P}$ denote the largest set of region-disjoint paths between a node pair s and t in the graph and let $m\kappa_R(G)$ denotes the size of the minimum region cut. It is trivially true that $m\kappa_R(G) \geq |\mathcal{P}|$ since otherwise, the graph will be connected even after the removal of all the regions in the region cut. In the example graphs of Figure 2, the size of minimum region cut is greater than the maximum number of region-disjoint paths between nodes s and t even with three different definitions of a region. In Figures 2 (a), (b) and (c), minimum RC is 2 while maximum RDP is 1 between the nodes s and t with three different definitions of a "region". The difference between the minimum RC and maximum RDP in the Figure 2(b) is 1. However, by repeating the graph structure of Figure 2(b) m times (as shown in Figure 3), the difference between minimum RC and maximum RDP can be made as high as m. Accordingly, the difference between minimum RC and maximum RDP between a node pair s and t can be arbitrarily large.

IV. Computation of Minimum Region Cut in MRFM

In this section, we formally state the Minimum Region Cut (MRC) Problem in MRFM, prove that the problem is NP-complete and provide a heuristic for the solution of the problem.

Given a graph $G = (V, E)$, two vertices $s, t \in V$ and a set of regions[5] $\mathcal{R} = \{R_1, R_2, \ldots, R_m\}$, the MRC problem is to compute the subset of regions $\mathcal{R}_s \subseteq \mathcal{R}$ of smallest size whose removal disconnects s and t.

We show the NP-completeness of a restricted version of MRC problem when the regions are computed using the radius-based region definition with radius value $r = 1$ in the following section.

A. Complexity Analysis of the MRC Problem

The decision version of the restricted MRC problem is as follows.

INSTANCE: Graph $G = (V, E)$, vertices $s, t \in V$, a set of regions $\mathcal{R} = \{R_1, R_2, \ldots, R_m\}$ computed with the radius-based region definition for radius value $r = 1$ and integer K

QUESTION: Is there a subset $\mathcal{R}_s \subseteq \mathcal{R}$ whose removal disconnects s and t such that $|\mathcal{R}_s| \leq K$?

To prove the NP-completeness of the MRC problem, we transform Set Cover problem [15] into MRC problem.

Theorem 1: MRC problem is NP-complete.

Proof: Clearly MRC problem is in NP since a nondeterministic algorithm need only guess the subset $\mathcal{R}_s \subseteq \mathcal{R}$ and verify in polynomial time whether $\mathcal{R}_s$ is a region cut with $|\mathcal{R}_s| \leq K$.

Suppose a finite set $X = \{x_1, x_2, \ldots, x_n\}$ of n elements and a family $\mathcal{F} = \{S_1, S_2, \ldots, S_m\}$ of m subsets of X make up the instance of Set Cover problem. From this instance of Set Cover, we will construct an instance of MRC problem such that minimum region cut exists with size $\leq K$ if and only if $\mathcal{F}$ contains a set cover whose size $\leq K$. We use local replacement technique for the NP-completeness proof. Corresponding to the instance of the Set Cover problem, we will construct a graph $G = (V, E)$ in which there is a vertex v_{x_i} corresponding to each element in $x_i \in X$ and a vertex pair v_{S_j} and v'_{S_j} corresponding to each subset $S_j \in \mathcal{F}$. Additionally, there are two vertices s and t in V. Thus, $V = \{v_{x_1}, v_{x_2}, \ldots, v_{x_n}\} \cup \{v_{S_1}, v'_{S_1}, v_{S_2}, v'_{S_2}, \ldots, v_{S_m}, v'_{S_m}\} \cup \{s, t\}$. The edge set includes the following four types of edges: (i) edges (s, v_{x_i}) and (t, v_{x_i}), $\forall 1 \leq i \leq n$, (ii) edges (v_{S_j}, v'_{S_j}), $\forall 1 \leq j \leq m$, (iii) edges between t and v'_{S_j}, $\forall 1 \leq j \leq m$ and (iv) for each subset $S_j = \{x_a, x_b, x_c, \ldots\}$, graph G will have edges connecting vertex v_{S_j} and vertices $\{v_{x_a}, v_{x_b}, v_{x_c}, \ldots\}$ corresponding to the elements of S_j. Consider the set of regions (not containing s and t) formed with the radius-based region definition with radius value $r = 1$. It can be easily verified that for $r = 1$, the regions in the graph constructed above are precisely the regions with vertex v_{S_j} as the center[6] for all $1 \leq j \leq m$. The graph formed for an example Set Cover instance is shown in Figure 4. It is clear that this construction procedure can be performed in polynomial time.

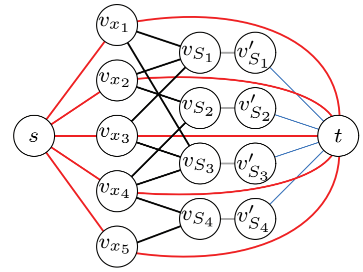


Fig. 4. MRC Graph for the Set Cover instance: $X = \{x_1, x_2, x_3, x_4, x_5\}$, $\mathcal{F} = \{S_1, S_2, S_3, S_4\}$ with $S_1 = \{x_1, x_2, x_3\}$, $S_2 = \{x_2, x_4\}$, $S_3 = \{x_1, x_3, x_4\}$, $S_4 = \{x_4, x_5\}$. The set of regions (with radius definition, r=1) excluding regions containing the source and destination are $R_1 = \{v_{S_1}, v'_{S_1}, v_{x_1}, v_{x_2}, v_{x_3}\}$, $R_2 = \{v_{S_2}, v'_{S_2}, v_{x_2}, v_{x_4}\}$, $R_3 = \{v_{S_3}, v'_{S_3}, v_{x_1}, v_{x_3}, v_{x_4}\}$, $R_4 = \{v_{S_4}, v'_{S_4}, v_{x_4}, v_{x_5}\}$. Here, minimum region-cut = $\{R_1, R_4\}$ and the optimal set cover = $\{S_1, S_4\}$.

Claim: The instance of MRC problem has a region cut of size at most K if and only if the set cover instance has a cover of size at most K.

Suppose that $\{S_1, S_2, \ldots, S_k\}$ are the subsets from $\mathcal{F}$ that make up a cover of X. Consider the set of regions $\mathcal{R}_s = \{R_{i_1}, R_{i_2}, \ldots, R_{i_k}\}$, where R_{i_l}, $1 \leq l \leq k$ is the region having vertex v_{S_l} as the center. Because of the construction procedure, removal of set of regions in $\mathcal{R}_s$ will result in removal of all the vertices $v_{x_1}, v_{x_2}, \ldots, v_{x_n}$, thereby isolating vertex s from the rest of the graph. Thus, $\mathcal{R}_s$ is a region cut of the graph.

To prove the converse, suppose that $\mathcal{R}_s = \{R_{i_1}, R_{i_2}, \ldots, R_{i_k}\}$ is a region cut of the graph and let v_{S_l} be the center of region R_{i_l}, $\forall 1 \leq l \leq k$. From the construction procedure, there exists a subset S_j in the set cover instance corresponding to each v_{S_j}, $\forall 1 \leq j \leq m$. Also, note that the graph has paths of the form $s \rightarrow v_{x_i} \rightarrow t$, $\forall 1 \leq i \leq n$. The sets $\{S_{i_1}, S_{i_2}, \ldots, S_{i_l}\}$ of subsets corresponding to the centers v_{S_l} of the regions of $\mathcal{R}_s$ form a set cover, since otherwise there would exist a vertex v_{x_a} through which a $s \rightarrow v_{x_a} \rightarrow t$ path can be found contradicting the region cut $\mathcal{R}_s$. ■

B. Solution to the MRC Problem

Since computing minimum region cut in a graph is NP-complete, we resort to efficient heuristics. An input to the MRC problem is the set of regions computed following the radius-based, diameter-based or geometry-based region definitions. We next describe the region enumeration process for the different definitions of region.

Computation of Regions in a graph

1) *Geometry-based Regions:* Given the locations of n nodes and a specific radius r in the geometry-based region

[5] None of the regions in this set contain the vertices s or t, since region cut for s and t will be meaningless if s or t itself is removed.

[6] 1-hop neighborhood around any other vertex will contain the source or the destination.

definition, there may potentially be an infinite number of regions that can cover part of the network layout. This is because if one slides the circular region slightly on the plane, it may be considered as a different region. However, we do not need to distinguish two regions if they cover the same set of nodes. In our earlier work [1], it was shown that the set of regions that contain distinct subset of nodes is finite in number. These regions can be found in $O(n^2)$ time using the procedure given below. Interested readers are referred to [1] for correctness proof of the algorithm.

Algorithm 1 compute_geometric_regions(P, r)

Input: Locations of n nodes $P = \{p_1, p_2, \ldots, p_n\}$, region radius r

Output: $\mathcal{R} = \{R_1, R_2, \ldots\}$ representing the geometric regions

Draw n circles with each circle centered at point $p_i \in P$ and radius r.

For every pair of circles whose centers are at most $2r$ distance apart, compute the points of intersection. These points are referred to as *I-points*. In case a circle does not intersect with any other circle, the center of the circle itself will be an *I-point*.

The circles drawn with centers at *I-points* and radius r constitute the different regions. Compute the set of circles that contain distinct subset of nodes. These are the required set of regions $\mathcal{R}$.

return $\mathcal{R}$

2) *Radius-based Regions:* The radius-based regions for a given graph $G = (V, E)$ and radius value r can be computed by considering the r-hop neighborhood around each node. This can be performed in $O(|E|)$ time [16].

3) *Diameter-based Regions:* For a given graph $G = (V, E)$ and a diameter value d, the maximal subgraph of G having diameter d corresponds to the notion of *n-club* often used in the context of Social Network analysis [17]. It has been shown in [17] that the number of *n-clubs* in a graph can be exponential. [18] provides an Integer Linear Program technique to compute the *n-club* of maximum size which can be extended to enumerate all *n-clubs* in a graph. We do not provide these results due to space considerations.

Heuristic for computing Region Cut

Given a graph $G = (V, E)$, a set of regions $\mathcal{R} = \{R_1, R_2, \ldots, R_m\}$ and two vertices $s, t \in V$, we propose a *simple* but *efficient* heuristic to compute the minimum region cut for s and t. The intuition behind the heuristic is based on the fact that the set of vertices belonging to a region cut of s and t is also vertex cut for s and t. Each vertex in the vertex cut may belong to multiple regions and a region may contain one or more vertices of the vertex cut. For each vertex cut, we can find the set of fewest regions that contains all the vertices of the vertex cut (using a set cover construction procedure explained below). The minimum region cut for s and t is then the smallest among all such sets of region covers. It is to be noted that it is sufficient to consider only the *minimal* vertex cuts, instead of considering *all* vertex cuts in the graph. A *minimal* vertex cut for vertices s, t in a graph G is a vertex cut V_c of s, t such that no proper subset of V_c is a vertex cut of s, t in G.

However, there may be an exponential number of minimal vertex cuts in a graph. Thus, we generate only k unique minimal vertex cuts in the heuristic, where k is a configurable parameter of the heuristic. We use the algorithm given in [19] to generate k unique minimal vertex cuts. The algorithm in [19] employs a recursive procedure to generate minimal vertex cuts at $O(|V||E|)$ computational effort per vertex cut.

The heuristic then constructs a set cover instance, in which there exists an element for each vertex of the vertex cut and a subset corresponding to each region in the graph. If a region R_i contains vertices V' of the vertex cut V_c, then the subset corresponding to R_i will contain elements corresponding to vertices V'. Since optimal set cover is NP-complete, we apply a well-known greedy set cover approximation algorithm to solve the set cover instance. The greedy set cover selects in each iteration, the subset S that covers the largest number of remaining elements that are uncovered. This polynomial-time algorithm provides an approximation ratio of $\ln |X|$, where $|X|$ is the number of vertices in the vertex cut [20].

Algorithm 2 compute_region_cut(G, $\mathcal{R}$, s, t, k)

Input: Graph $G = (V, E)$, set of regions $\mathcal{R} = \{R_1, R_2, \ldots\}$, vertices s and t, heuristic parameter k

Output: region cut $\mathcal{R}_s \subseteq \mathcal{R}$

Initialize $\mathcal{R}_s = \emptyset$

Generate k *minimal* vertex cuts $\mathcal{V}_c = \{V_1, V_2, \ldots, V_k\}$ in G using the algorithm in [19]

for all $V_i \in \mathcal{V}_c$ with $V_i = \{v_{i_1}, v_{i_2}, \ldots, v_{i_n}\}$ **do**

  $\forall 1 \leq l \leq n$, let R_{i_l} be the set of regions containing vertex v_{i_l}. Let the number of unique regions containing all the vertices of V_i be m.

  Construct set cover instance with $X = \{x_{i_1}, x_{i_2}, \ldots, x_{i_n}\}$ and family of subsets $\mathcal{F} = \{S_1, S_2, \ldots, S_m\}$. For each vertex $v_{i_l} \in V_i$, if $v_{i_l} \in R_{i_j}$ then $x_{i_l} \in S_j$.

  Solve the set cover instance using the greedy set-cover algorithm. Suppose the solution returned by the set-cover algorithm is $\{S_{i_1}, S_{i_2}, \ldots, S_{i_p}\}$ and the regions corresponding to these subsets is $\{R_{i_1}, R_{i_2}, \ldots, R_{i_p}\}$

  if $|\mathcal{R}_s| < p$ **then**

    $\mathcal{R}_s \leftarrow \{R_{i_1}, R_{i_2}, \ldots, R_{i_p}\}$

  end if

end for

return $\mathcal{R}_s$

Optimal Solution for computing Region Cut

In the heuristic for the MRC problem, there are two places for loss of accuracy of the heuristic. The first is because of the value of k chosen. If the value of k is such that all minimal vertex cuts of the graph are not being generated, then the vertex cut corresponding to the best region cut may not be found. The second place of inaccuracy is the approximate solution to the set cover for each minimal vertex cut. In order to compute the optimal solution for the MRC problem, we set k to be a sufficiently large value so that all minimal vertex cuts in the graph are generated and solve the set cover instance optimally using an Integer Linear Program formulation[7].

It may be noted that the multi-region-based connectivity $m\kappa_R(G)$ of the graph G can be obtained by invoking region cut algorithm on all vertex pairs in the graph and taking the minimum of the region cut values. Accordingly, depending on whether exact $m\kappa_R(G)$ is required or approximate $m\kappa_R(G)$ is sufficient, the optimal or heuristic region cut algorithms can be applied respectively.

We conducted extensive experiments to evaluate the efficacy of the heuristic by comparing it with the optimal solution. The results of the experiments are presented in Section VII.

V. Computation of Maximum Number of Region-Disjoint Paths in MFRM

In this section, we formally state the Maximum Number of Region-Disjoint Paths (MRDP) Problem in MRFM, prove that the problem is NP-complete and provide a heuristic for the solution of the problem.

Given a graph $G = (V, E)$, two vertices $s, t \in V$ and a set of regions $\mathcal{R} = \{R_1, R_2, \ldots, R_m\}$, the MRDP problem is to compute maximum number of region-disjoint paths between s and t. We provide the NP-completeness of MRDP problem in the next section.

A. Complexity Analysis of the MRDP Problem

We show that a restricted version of MRDP problem is NP-complete.

INSTANCE: Graph $G = (V, E)$, vertices $s, t \in V$ and a set of regions $\mathcal{R} = \{R_1, R_2, \ldots, R_m\}$ computed with diameter definition of region with $d = 1$

QUESTION: Are there *two* region-disjoint paths between s and t?

Theorem 2: MRDP problem is NP-complete.

Proof: A *hole* is a set of vertices in a graph G that induces a chordless cycle in G. The *hole recognition* problem is to determine if a hole passing through two vertices in a graph exists. This problem was proved to be NP-complete in [21]. In the diameter-based definition of region with $d = 1$, two paths will be region-disjoint if and only if they induce a chordless cycle. Thus, for $d = 1$, the restricted MRDP problem is equivalent to the *hole recognition* problem in a graph. ■

B. Solution to the MRDP Problem

Given the graph $G = (V, E)$, vertices $s, t \in V$ and set of regions $\mathcal{R} = \{R_1, R_2, \ldots, R_m\}$, the MRDP heuristic computes many region-disjoint paths between s and t. The heuristic works in two phases. In the first phase, it computes maximum number of node-disjoint paths between s and t in the graph using max-flow-min-cut techniques [20]. If the computed node-disjoint paths are also pairwise region-disjoint then we are done. If not, the heuristic enters the second phase in which it finds a large subset of the node-disjoint paths that is region-disjoint. This is accomplished by constructing a ***path intersection graph***. Suppose $\mathcal{P}_{ND} = \{P_1, P_2, \ldots, P_n\}$ are the n node-disjoint paths computed in the first phase. In the second phase, the heuristic constructs the *path intersection graph* $G' = (V', E')$, in which a vertex $v_i \in V'$ corresponding to a path $P_i \in \mathcal{P}_{ND}$. If two paths $P_i, P_j \in \mathcal{P}_{ND}$ are not region-disjoint, then we have an edge $(v_i, v_j) \in E'$. In the path intersection graph G', an independent set of vertices corresponds to a set of region-disjoint paths in graph G and hence, the heuristic finds a large independent set of vertices in G'. Since the problem of finding maximum independent set in a graph is NP-complete [15], the heuristic uses a greedy approach to find an independent set in G'. The greedy algorithm in each iteration selects the vertex with the minimum degree into the independent set and removes the vertex and its neighbors from the graph. The procedure is repeated until there are no vertices left in the graph. The heuristic returns the region-disjoint paths corresponding to the vertices in the independent set.

Algorithm 3 compute_region_disjoint_paths(G, $\mathcal{R}$, s, t)

Input: Graph $G = (V, E)$, set of regions $\mathcal{R} = \{R_1, R_2, \ldots\}$, vertices s and t
Output: set $\mathcal{P}$ of region-disjoint paths

1: $\mathcal{P} \leftarrow \emptyset$
2: Compute $\mathcal{P}_{ND} = \{P_1, P_2, \ldots, P_n\}$, the set of node-disjoint paths between s and t in G.
3: Construct *path intersection graph* $G' = (V', E')$, where $\forall P_i \in \mathcal{P}_{ND}, v_i \in V'$ and edge $(v_i, v_j) \in E'$ if paths P_i, P_j are not region-disjoint.
4: **while** $V' \neq \emptyset$ **do**
5: Find vertex $v_i \in V'$ such that v_i has the smallest degree.
6: $\mathcal{P} \leftarrow \mathcal{P} \cup \{P_i\}$
7: $V' \leftarrow V' \setminus \{v_i \cup N(v_i)\}$, where $N(v_i)$ is the set of neighbors of v_i.
8: **end while**
9: **return** $\mathcal{P}$

Optimal Solution for computing Region-Disjoint Paths

In order to obtain the optimal solution to the MRDP problem, we first enumerate all paths between s, t. There exists a number of algorithms to compute k shortest paths between a source-destination node pair in a graph [22] that can be used for this purpose. Next, the path intersection graph is constructed on the set of all the paths. We can find optimal independent set of the path intersection graph using Integer Linear Program (ILP) formulation[8].

[7] We do not provide the ILP formulation due to space constraints.

[8] We do not provide the ILP formulation due to space constraints.

VI. Fault-Tolerant Network Design in MRFM

In this section, we focus on fault-tolerant design of a wireless network. The goal of the design is to attain a specified value of multi-region-based connectivity with the smallest transmission range possible. The inputs to the problem are the locations of the nodes in a two-dimensional plane, radius r of the circular region, required multi-region-based connectivity value $m\kappa_R(G)$ and the network design problem is to compute the minimum transmission range for each node, that will result in a network with multi-region-based connectivity $m\kappa_R(G)$.

The heuristic compute_transmission_range finds a small transmission range for each sensor node that ensures a specified multi-region-based connectivity. The heuristic uses *binary search* on an ordered range of possible transmission ranges. For each value of transmission range, it checks if the graph provides the required multi-region-based connectivity. The procedure for computing region cut can be used for this purpose. If the optimal transmission range is sought, then the optimal technique of computing multi-region-based connectivity can be applied. The steps of the heuristic are provided in Algorithm 4.

Algorithm 4 compute_transmission_range(P, r, L, K)

Input: Locations of n nodes $P = \{p_1, p_2, \ldots, p_n\}$, region radius r, list L of possible transmission ranges sorted in nondecreasing order and required multi-region-based connectivity K

Output: Minimum transmission range that ensures multi-region-based connectivity K.

$low \leftarrow 0$, $high \leftarrow |L|$.
while $low \leq high$ **do**
  $mid \leftarrow \frac{low+high}{2}$
  Use $L[mid]$ as the transmission range to derive network $G = (V, E)$ from sensor layout.
  if $G = (V, E)$ is connected **then**
    Find $K_r \leftarrow$ multi-region-based connectivity of graph G using the compute_region_cut heuristic.
  else
    Set $K_r \leftarrow 0$
  end if
  if $K_r > K$ **then**
    $high \leftarrow mid - 1$
  else
    $low \leftarrow mid + 1$
  end if
end while
return Tr_{mid}

VII. Experimental Results and Discussion

In order to evaluate the effectiveness of the proposed heuristics, we conducted extensive simulations. Since all our experiments were conducted on the network layout, the geometric-based region definition was applied to compute the different regions in the graph. The experiments conducted can be categorized into two types.

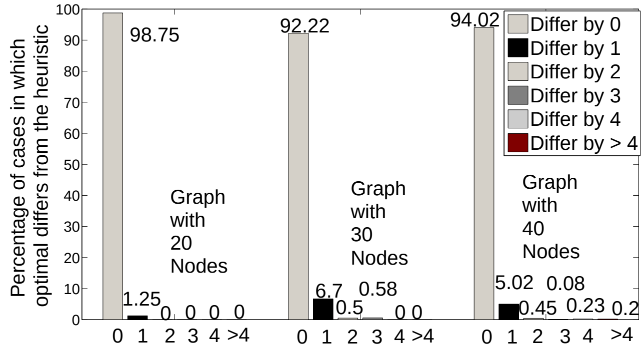


Fig. 5. Percentage of cases in which Region Cut heuristic differs from optimal by value i for three different random graphs of size 20, 30 and 40

Type I experiments were conducted to measure the accuracy of the solution produced by the region cut and region-disjoint paths heuristics. In these experiments, the results obtained from the heuristic were compared with the optimal solutions for both region cut and region-disjoint paths. The optimal solutions were obtained by solving the Integer Linear Programs using CPLEX Optimizer 10.0. In all these experiments, the location coordinates for the n nodes of the graphs are uniformly generated in a layout of 1000×1000 unit area. We consider three parameters that influence the comparison results: (i) the number of nodes n, (ii) the region radius r and (iii) the transmission range T_r of the nodes. A number of experiments were conducted to measure the effect of these parameters on the region cut and number of region-disjoint paths in the graph.

A. Results for the MRC Problem

When computing region cut using the heuristic, the number of minimal vertex cuts, k was taken to be 50. In the first set of experiments under Type I, we set $r = 100$ and vary the transmission range T_r of the sensor nodes from 250 to 500 in steps of 50. For each transmission range, the difference between the region cut produced by the heuristic and the optimal for *all source-destination pairs* in the graph is measured. Experiments were performed for graphs having different number of nodes (20, 30 and 40 nodes). For each value of n, 10 random graphs were generated each having n nodes. The results were averaged over the 10 graphs. Figure 5 shows the percentage of cases where heuristic cut value differed from the optimal cut value by i for $0 \leq i \leq 4$ and $i > 4$. It may be observed that in all the three types of graphs, the heuristic computed the optimal region cut for more than 92% of the source-destination pairs.

The results of experiments showing the effect of the transmission range on region connectivity is shown in Figure 6. This set of experiments was performed on graph of 30 nodes and region radius 100. It may be observed that the region connectivity increases with the increase in transmission range.

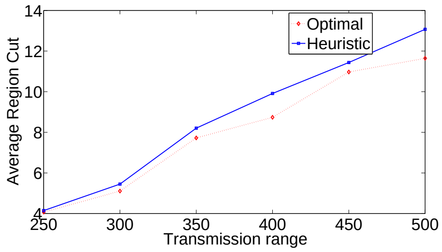


Fig. 6. Average Region Cut vs Region Radius of a graph with 30 nodes and transmission range of 150 unit

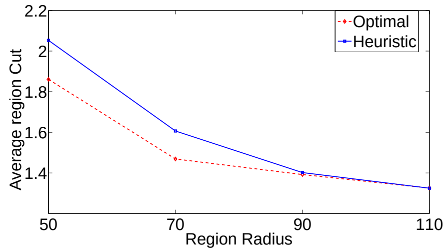


Fig. 7. Average region Cut vs Transmission Range for a graph with 30 nodes and region radius of 100 unit

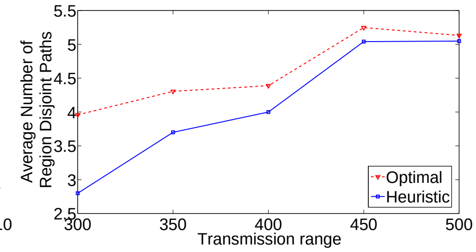


Fig. 8. Average Number of Disjoint Paths vs Transmission Range in a graph of 30 nodes and region radius of 100 units

This is natural since increase in transmission range induces more edges in the graph, resulting in increased connectivity. Thus, more regions have to be removed on an average to disconnect two nodes in the graph. As in the previous case, the heuristic gives near-optimal results. In the third set of experiments on region connectivity, we vary the region radius r from 50 to 110 in steps of 20 on a randomly generated graphs of size 30 while keeping the transmission range T_r constant at 150. The results of the experiment are shown in Figure 7. As seen in the Figure, the region connectivity decreases with the increase in region radius. This is because when region radius is increased, a region may contain more nodes than before and thus, the region cut may contain lesser number of regions. Here again, the heuristic gives near-optimal results.

B. Results for the MRDP Problem

In the other part of the Type `I` experiments, we evaluated the efficacy of the region-disjoint paths heuristic. In this experiment, we measured the average number of cases for which the heuristic solution differs from the optimal solution. The results for this set of experiment are given in Figures 9 and 8. Figure 9 shows the percentage of cases where the solution produced by heuristic differed from the optimal region-disjoint paths by i for $0 \leq i \leq 4$ and $i > 4$. It can be seen that in more than 87% of the cases, the number of region-disjoint paths produced by the heuristic differed from the optimal by at most 1. Figure 8 shows the variation of number of region-disjoint paths (averaged over all source-destination pairs) with the transmission range. It can be seen from the figure that increase in transmission range increases the number of region-disjoint paths. This is because the increase in transmission range induces more edges in the graph and thus more region-disjoint paths may be available.

C. Results for the Network Design Problem

The Type `II` experiments were conducted for the network design problem. In these experiments, the nodes were randomly distributed in a field of 500×500 sq-units. In these experiments, we measure the change of transmission radius T_r with the region based connectivity $m\kappa_R$ for different varying number of nodes and the region radius r. The goal is to compute the minimum transmission range of the nodes to ensure region-connectivity $m\kappa_R$. It may be noted that

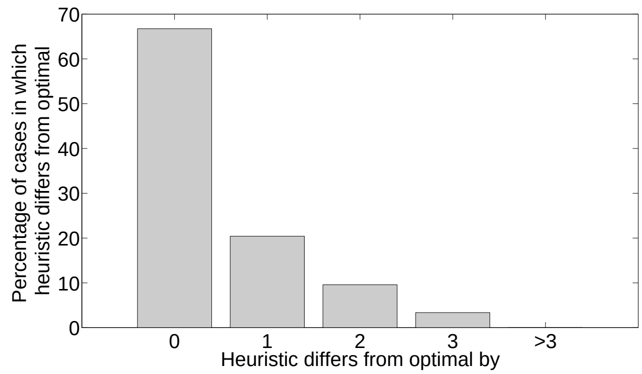


Fig. 9. Percentage of cases in which region-disjoint path heuristic differs from optimal by value i for random graphs of size 30

region based connectivity $\kappa^{'}(G)$ can be provided for a graph G if we design the graph with general node connectivity $\kappa^{'}(G) = m\kappa_R(G) \times p$, where p is the maximum number of node that contains in a region of the graph G. We compared the transmission ranges required to obtain a k region connected graph by the proposed network design algorithm with the transmission range needed to get same region connected graph by the above method. We compare these two sets of results for region radius varying from 50 to 90 at steps of 20. The region connectivity $m\kappa_R$ is varied from 2 to 10. We conducted experiments for graphs containing different number of nodes n, 20, 30 and 40. For each value of n, the average transmission range was computed over 10 random graphs each containing n nodes. The results are given Figure 10, 11 and 12. It is clear that the proposed network design technique results in smaller transmission range to achieve multi-region fault tolerance $m\kappa_R$ compared with the transmission range required to design $m\kappa_R(G) \times p$ fault tolerant network.

Few interesting observations can be made in these cases. First, for all three types of graph of different node-density the transmission range increases with the increase in region radius for the same region connectivity. This result is expected because with the increase in region radius removal of a region takes out more node and links, so transmission range needs to be higher to maintain the same connectivity. Secondly, with the increase in node density the retransmission range decreases.

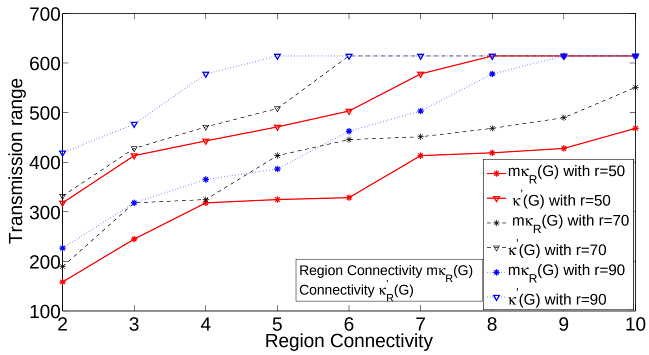


Fig. 11. Transmission Range vs Region Connectivity $m\kappa_R$ with $n = 30$ and region radius $r = 50$, 70 and 90

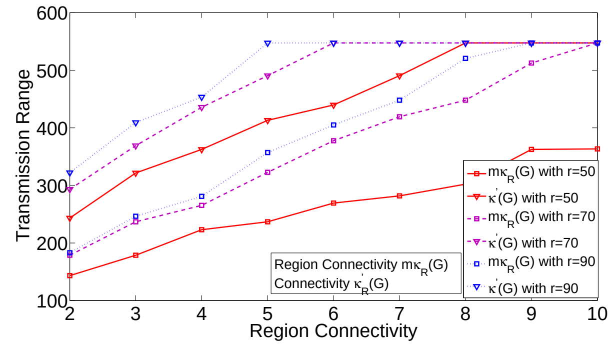


Fig. 12. Transmission Range vs Region Connectivity $m\kappa_R$ with $n = 40$ and region radius $r = 50$, 70 and 90

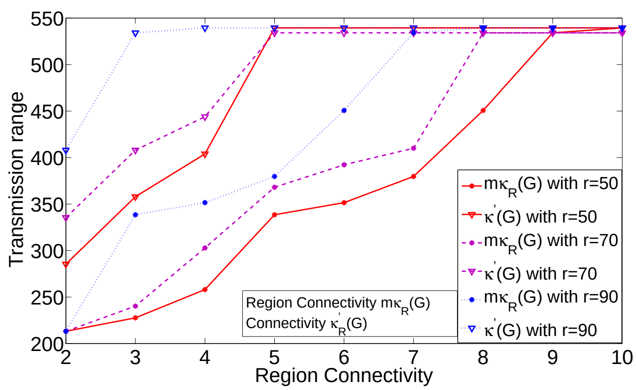


Fig. 10. Transmission Range vs Region Connectivity $m\kappa_R$ with $n = 20$ and region radius $r = 50$, 70 and 90

This can be explained as increasing node density means increasing proximity of the nodes, so low transmission range will suffice to maintain the desired region connectivity. However with the increase in connectivity both the methods reach some high value of transmission range and further increase is not possible as the network becomes fully connected by then.

VIII. Conclusion

In this paper, we proposed a new *multiple region fault model* to capture the notion of *massive* but *localized* faults in the network. We showed that the notion of *region-disjoint paths* and *region cuts* do not necessarily obey Menger's theorem with respect to maximum number of region-disjoint paths and minimum size of region cut. We provided the NP-completeness results and efficient heuristics for computing minimum region cut and maximum region-disjoint paths in a graph. The extensive experiments conducted demonstrate the efficacy of the heuristics in computing close to optimal results in very less time. The experimental results also showed cost effectiveness (in terms of power consumption) of multi-region-based connectivity than the conventional metric in achieving the same level of fault-tolerance in the network.

References

[1] A. Sen, B. Shen, L. Zhou, and B. Hao, "Fault-tolerance in sensor networks: A new evaluation metric," in *INFOCOM*, 2006.

[2] C. Bettstetter and C. Hartmann, "Connectivity of wireless multihop networks in a shadow fading environment," in *MSWiM*, 2003.

[3] Y.-C. Cheng and T. Robertazzi, "Critical connectivity phenomena in multihop radio models," *IEEE Trans. on Communications*, 1989.

[4] P. Gupta and P. Kumar, "Critical power for asymptotic connectivity in wireless networks," *Stochastic Analysis, Control, Optimization and Applications*, 1998.

[5] M. D. Penrose, "On k-connectivity for a geometric random graph," *Wiley*, vol. 15, no. 2, pp. 145–164, 1999.

[6] T. J. Philips, S. S. Panwar, and A. N. Tantawi, "Connectivity properties of packet radio network model," *IEEE Transactions on Information Theory*, vol. 35, no. 5, pp. 1044–1047, Sept. 1989.

[7] P. Pirlet, "On the connectivity of radio networks," *IEEE Trans. on Information Theory*, vol. 37, no. 5, pp. 1490–1492, Sep. 1991.

[8] B. Rangarajan, J. K. Chen, S. Shakkottai, and T. S. Rappaport, "Connectivity of sensor networks with power control," in *The Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 2, Nov. 2003, pp. 1691–1693.

[9] M. C. Balbuena, A. Carmona, and M. A. Fiol, "Distance Connectivity in Graphs and Digraphs," *Journal of Graph Theory*, vol. 22, no. 4, pp. 281–292, 1998.

[10] F. Juhasz, "The asymptotic behaviour of fiedler's algebraic connectivity for random graphs," *Discrete Math.*, vol. 96, no. 1, pp. 59–63, 1991.

[11] B. Peroche, "On several sorts of connectivity," *Discrete Math.*, vol. 46, pp. 267–277, 1983.

[12] M. Wang and Q. Li, "Conditinal edge connectivity properties reliability comparisons and transitivity of graphs," *Discrete Math.*, vol. 258, pp. 205–214, 2002.

[13] D. W. Vanderjagt, "Graphs with prescribed local connectivity, discrete mathematics," vol. 10, pp. 391–395, 1974.

[14] R. Diestel, *Graph Theory*. Springer, 2003.

[15] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman Press, 1979.

[16] T. H. Cormen, C. E. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*, second edition ed. McGraw Hill, 2001.

[17] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994.

[18] B. Balasundaram, S. Butenko, and S. Trukhanov, "Novel approaches for analyzing biological networks," *Journal of Combinatorial Optimization*, 2005.

[19] C. Patvardhan, V. Prasad, and V. P. Pyara, "Vertex cutsets of undirected graphs," *IEEE Transactions on Reliability*, 1995.

[20] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*. Prentice Hall, 2004.

[21] D. Bienstock, "On the complexity of testing for odd holes and induced odd paths," *Discrete Mathematics*, 1991.

[22] D. Eppstein, "Finding the k shortest paths," Tech. Rep., 1994. [Online]. Available: http://www.ics.uci.edu/~eppstein/pubs/Epp-TR-94-26.pdf