

Resilient Greedy Routing on GPS-free Surface Sensor Networks

Journal Title
XX(X):1-10
©The Author(s) 2016
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/ToBeAssigned
www.sagepub.com/

SAGE

Buri Ban¹, Xuan Li¹, and Miao Jin¹

Abstract

We design a greedy routing scheme specifically for GPS-free large-scale wireless sensor networks deployed on surfaces of complex-connected three-dimensional (3D) settings. Compared with other greedy embedding based surface network routing scheme, the proposed one is cut free such that no pair of nodes suffers a long detour to reach each other. The routing scheme is designed to be resilient to node or link failures especially under random node or link failure model where each node in a network has an equal and independent probability of failure during some time interval. The proposed algorithm is fully distributed and scalable to both the size and the topological complexity of a network. Each sensor node requires only limited and constant storage. Simulation results show the proposed routing scheme with a higher successful delivery ratio, a lower average stretch factor, and a lower normalized communication cost compared with other resilient routing methods.

Keywords

Greedy Routing, Surface Network, Resilient, GPS-free

Introduction

A wireless sensor network may be deployed on a 2D plane [Akyildiz et al. \(2002\)](#) (e.g., for crop sensing in fields or wildlife tracking on plains), or in a 3D volume [Allred et al. \(2007\)](#); [Xia et al. \(2013\)](#); [Cui et al. \(2006\)](#) (for underwater or space reconnaissance), or on a 3D surface [Zhao et al. \(2012, 2009\)](#) (e.g., with sensors mounted on ocean floor, mountain surface, or the surface of various man-made structures).

Routing has always been a fundamental challenge in large-scale wireless sensor networks. A diversity of sensor network routing algorithms have been developed, but they primarily target at 2D planar or 3D volumetric settings [Bose et al. \(1999\)](#); [Karp and Kung \(2001\)](#); [Kranakis et al. \(1999\)](#); [Kuhn et al. \(2003a,b\)](#); [Mitra and Liskov \(2005\)](#); [Frey and Stojmenovic \(2006\)](#); [Tan et al. \(2009\)](#); [Durocher et al. \(2008\)](#); [Wang et al. \(2008\)](#); [Liu and Wu \(2009\)](#); [Xia et al. \(2014\)](#). This paper focuses on designing a routing algorithm resilient to random node or link failures in a large-scale wireless sensor network deployed on the surface of a complex-connected three-dimensional (3D) setting. Such 3D setting generally has multiple handles and forms a high genus surface [Yu et al. \(2012\)](#). Applications for such wireless sensor networks on high genus surfaces have been demonstrated for Structural Health Monitoring (SHM), e.g., at the Golden Gate Bridge [Kim et al. \(2007\)](#) and National Stadium of China [Shen et al. \(2013\)](#), where sensors are deployed on megastructures to gauge changes in materials or geometric properties that could hinder the systems performance. Other applications can be found along the corridors of buildings for fire detection, and in water, sewer or gas system for monitoring underground pipelines as introduced in [Yu et al. \(2013\)](#). We consider densely-deployed sensors that operate on high radio frequency and extremely low transmission power with short communication range.

Only nearby sensors along a 3D surface can communicate with each other, whereas the wireless links connecting remote sensors across space are negligible and can be removed via a simple preprocessing. As a result, the dense sensors form a 3D surface network.

An Overview of Resilient Routing

Node and link failures are unavoidable in a large-scale distributed sensor system. For example, nodes may die out of battery and communication links can be temporarily or permanently disabled due to attacks or interference. We can model network failures with two failure models: the independent node failure model and the geographically correlated failure model. In the independent node failure model, each node in a network has an equal and independent probability of failure during some time interval [Ganesan et al. \(2001\)](#). Independent node failure can represent failure due to energy dissipation or localized environmental effects. In the geographically correlated failure model, all nodes within a certain range or hop count fail [Ganesan et al. \(2001\)](#). Geographically correlated failure may happen due to natural environmental effects (such as rain fades) or some human activity within a geographic region.

Resilient routing has been discussed in conventional 2D sensor networks. For example, Directed Diffusion [Intanagonwiwat et al. \(2000\)](#) is a well-known routing algorithm that establishes gradients and uses gradual

¹The Center for Advanced Computer Studies, University of Louisiana at Lafayette, U.S.A.

Corresponding author:

Miao Jin, The Center for Advanced Computer Studies, University of Louisiana at Lafayette, Lafayette, LA, 70504, U.S.A.

Email: miao.jin@louisiana.edu

reinforcement of better paths to allow path recovery, enabling a system to be robust to a local network and sink dynamics. Multipath routing represents an efficient routing strategy to increase network resilience to node and link failures. Algorithms have been proposed for Internet multipath routing [Shand and Bryant \(2009\)](#); [Atlas and Zinin \(2008\)](#); [Reichert et al. \(2005\)](#); [Ray et al. \(2010\)](#); [Ohara et al. \(2009\)](#) and sensor network multipath routing [Ganesan et al. \(2001\)](#); [Lou \(2005\)](#). In [Ganesan et al. \(2001\)](#), the authors consider two different approaches to construct multipaths between two nodes. One is node-disjoint multipath, where the alternate paths do not intersect with each other. The other approach builds many braided paths - partially disjoint alternate paths. In [Lou \(2005\)](#), a branch-aware flooding scheme is utilized to construct a spanning tree and discover a set of node-disjoint paths for each sensor node back to the base station. Later, a tree structure is applied for multipath routing [Motiwala et al. \(2008\)](#); [Gao and Zhou \(2011\)](#). One recent advance in multipath routing is homotopy-based routing [Zeng et al. \(2010\)](#). The authors apply the concept of homotopy to effectively evaluate the diversity of alternative paths of a network deployed on 2D plane with uncovered holes. The simplest case to illustrate the concept of paths with different homotopy types is a network deployed on a planar region with a lake inside. Two paths connecting a pair of source and destination are homotopic to each other if they are on the same side of the lake; otherwise, they belong to different homotopy types because one path cannot continuously deform to the other without crossing the lake. The most recent work in multipath routing is introduced in [Liu et al. \(2020\)](#) with a security disjoint routing-based verified message (SDRVM) scheme proposed to improve the network performance with comprehensive consideration of universal resistance to attacks, energy efficiency, and transmission delay. The main strategy is to divide nodes into several sets based on their remaining energy and assign different duties to each set.

Multipath routing schemes are effective for geographically correlated failures. Alternative paths, especially those disjoint with the failed one, has a high probability to avoid a network failure region. Homotopy-based multipath routing schemes choose alternative paths topologically different from the failed one. These homotopy different paths provide a high probability of routing success even under massive geographically correlated failures. However, under the independent node failure model, the performance of multipath routing schemes decrease dramatically with a slight increase of node failure probability. The reason is that an alternate path is always longer than the original one for any multipath method, so the failure probability of the alternate path is always higher than the original one considering each node has an equal probability of failure. It is obvious that multipath routing schemes, especially those choosing alternative paths based on disjoint or different homotopy types, have intrinsically a very low resilience to independent node failures. If the probability of node failure is not low, switching to an alternate path does not help.

Instead of choosing an alternative path, a local detour is more efficient under the independent node or link failure model. We can consider dead nodes or regions with broken links as holes. To avoid torrents of triggered global updates

of network topology, a practical strategy is to apply Face routing and its alternatives [Bose et al. \(1999\)](#); [Karp and Kung \(2001\)](#); [Kranakis et al. \(1999\)](#); [Kuhn et al. \(2003a,b\)](#); [Mitra and Liskov \(2005\)](#); [Frey and Stojmenovic \(2006\)](#); [Tan et al. \(2009\)](#) to locally detour packages around the boundary of each hole. Specifically, A node always forwards a packet to one of its neighbors closest to the destination of the packet. When greedy forwarding of a packet fails due to holes, the packet can be forwarded along the boundary of holes in either clockwise or counter-clockwise direction until greedy forwarding is achievable.

However, Face routing and its alternatives require all sensor nodes to know their own positions. For networks deployed on indoor, underground or underwater surfaces, GPS signal is forbidden. They also require networks to be planar. For networks deployed on surfaces of complex-connected 3D settings, when greedy forwarding of a packet is stuck at a local minimum including a hole, as shown in Fig.1 (a), there is no deterministic path to successfully forward the packet to the destination based on local information only. We propose a fully distributed algorithm to apply such local detour strategy to achieve efficient resilient routing for GPS-free sensor networks deployed on surfaces of complex-connected 3D structures. Routing is based on local information only. Network dynamic information only needs to be updated locally.

Our Approach:

A few greedy routing methods [Yu et al. \(2012, 2013\)](#) have been developed recently to address the GPS-free surface sensor network routing problem. In [Yu et al. \(2012, 2013\)](#), the authors propose to cut a closed high genus surface network open and then map the cut-open one to a planar rectangle with two pairs of boundaries. Furthermore, the planar rectangle can be mapped to a planar annulus with one pair of boundaries aligned and glued together. Each sensor node stores a set of planar coordinates and uses it to enable greedy routing. However, these methods suffer a large stretch factor in average. The averaged stretch factor is defined as an average of the ratios of the hop counts of greedy forwarding paths to the shortest ones. When the original surface network is cut open and mapped to either a planar disk, rectangle, or annulus, nodes located at the two sides of the cut-open boundaries need to take a long detour to reach each other instead of directly crossing the boundary. Fig.1 (b) shows an example of a pair of such nodes.

The problem of a large stretch factor is intrinsic to the methods proposed in [Yu et al. \(2012, 2013\)](#). They need to cut the originally closed surface network open and then map the network to plane. We propose a cut-free embedding of the original network in the plane. Locally, any region of the network is mapped to plane one-to-one and continuously. The key of the seamless embedding is to extract a triangular mesh structure from the connectivity graph of the original network and compute a set of special edge lengths for the mesh. Based on the computed edge lengths, the original surface network can be embedded in the plane seamlessly. A packet is greedily forwarded based on the planar coordinates stored at each node. Later, the network dynamically maintains the mesh structure. Each node periodically updates its neighboring nodes and local

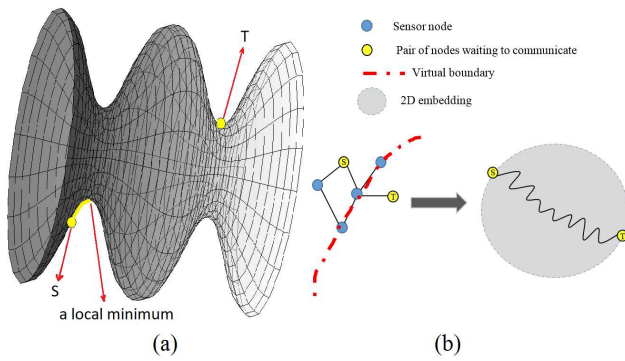


Figure 1. (a) Greedy forwarding of a packet is stuck at a local minimum on a surface of 3D setting. There is no deterministic path to successfully forward the packet to the destination based on local information only. (b) A pair of nodes with the shortest path crossing the virtual boundary in the original network needs to take a much longer detour to reach each other based on the computed virtual coordinates.

mesh structure due to a potential node or link failure. A face of the mesh is always an n -polygon with $n \geq 3$. We consider all faces with $n > 3$ holes. When a packet is routed to hit a hole, the packet is forwarded along the boundary of the hole in either counterclockwise or clockwise direction until greedy forwarding is achievable.

The simulation shows the proposed routing scheme with a higher successful delivery ratio, a lower average stretch factor, and a lower normalized communication cost compared with other resilient routing methods. Such results indicate the proposed routing scheme has a strong resilience to network failures, a low cost to detour, and long durability of the network system.

Our Contributions: Our contributions can be summarized as follows:

- We design a greedy routing scheme specifically for GPS-free wireless sensor networks deployed on surfaces of complex-connected 3D settings. Compared with other greedy embedding based surface network routing scheme, the proposed approach is cut free such that no pair of nodes suffers a long detour to reach each other.
- The routing scheme is designed to be resilient to node or link failures especially under random node or link failure model. Simulation results show a high successful delivery ratio, a low average stretch factor, and a low normalized communication cost.
- The proposed algorithm is fully distributed and scalable to both the size and the topological complexity of a network.

The rest of this paper is organized as follows: Section “Uniformization Metric” gives a brief introduction of uniformization metric - the special set of edge lengths we compute for a surface network. Section “Algorithms” elaborates the proposed resilient routing algorithms. Section “Simulations” presents simulation and comparison results. Section “Conclusion” concludes the paper.

Uniformization Metric

In this section, we introduce the uniformization metric, which is essentially a special set of edge lengths we compute to support the proposed resilient routing algorithm. Denote S a closed surface embedded in \mathbb{R}^3 and L a loop on S . L is surface separating if it can be expressed as the symmetric difference of boundaries of topological disks embedded in a surface as shown in Figure 2 with L_1 and L_2 ; otherwise, it is non-separating as shown in Figure 2 with L_3 .

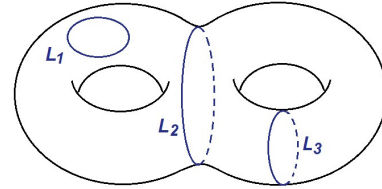


Figure 2. Loops on a genus two surface: L_1 and L_2 are surface separating loops; L_3 is non-separating loop.

The genus of S , equivalent to the number of handles of S , is the maximum number of disjoint non-separating loops L_1, L_2, \dots, L_g in M ; that is, any L_i and L_j have no topological intersection if $i \neq j$, and $M \setminus (L_1 \cup \dots \cup L_g)$ is connected. The genus number is the most basic topology information of a surface. For example, a torus is a genus one surface, and a double torus as shown in Figure 2 is a genus two surface.

A special set of edge lengths to embed S to plane without cutting is called uniformization metric. Uniformization metric exists for general closed surfaces according to Uniformization Theorem Schlag (2014).

Informally speaking, let S with Riemannian metric induced from the Euclidean metric of S . We denote the Riemannian metric of S as \mathbf{g} . There exists a unique metric represented as $\bar{\mathbf{g}} = e^{2u}\mathbf{g}$, where $u : S \rightarrow \mathbb{R}$ is a scalar function defined on S . We call $\bar{\mathbf{g}}$ a uniformization metric of S . It can be verified that $\bar{\mathbf{g}}$ is also a Riemannian metric on S and $\bar{\mathbf{g}}$ induces constant Gaussian curvature on S .

Specifically, based on surface uniformization metric, S with genus one can be periodically embedded into Euclidean plane and S with genus larger than one can be periodically embedded into the hyperbolic plane.

If S is a discrete surface, for example, a triangular mesh, We have the following discrete definitions of Riemannian metric and Gaussian curvature.

Given a triangular mesh, we denote a vertex set as V , an edge set as E , and a face set as F . e_{ij} represents the edge connecting vertices v_i and v_j , and f_{ijk} denotes the face formed by v_i , v_j , and v_k . The edge lengths of a triangular mesh are sufficient to define a discrete Riemannian metric on the discrete surface,

$$l : E \rightarrow \mathbb{R}^+,$$

as long as, for each face f_{ijk} , the edge lengths satisfy the triangle inequality: $l_{ij} + l_{jk} > l_{ki}$.

The discrete Gaussian curvature denoted as K_i on a vertex $v_i \in \Sigma$ can be computed from the angle deficit,

$$K_i = \begin{cases} 2\pi - \sum_{f_{ijk} \in F} \theta_i^{jk}, & v_i \notin \partial\Sigma \\ \pi - \sum_{f_{ijk} \in F} \theta_i^{jk}, & v_i \in \partial\Sigma \end{cases} \quad (1)$$

where θ_i^{jk} represents the corner angle attached to vertex v_i in the face f_{ijk} , and $\partial\Omega$ represents the boundary of the mesh. Since θ_i^{jk} can be computed directly from edge lengths of f_{ijk} , it is obvious that discrete Gaussian curvatures are determined by the discrete metrics.

Uniformization metric also exists on a discrete surface. It is a special set of edge lengths such that the discrete surface can be embedded in the plane based on the edge lengths with vertex Gaussian curvatures being zero everywhere. Discrete surface Ricci flow is an efficient tool to compute discrete surface uniformization metric. The algorithm of discrete surface Ricci flow is fully distributed, so it is well suited for wireless sensor network applications. We refer readers to [Jin et al. \(2008\)](#) for details of discrete surface Ricci flow.

Algorithms

This section describes the proposed routing scheme in three steps. Sec. gives the preprocessing step. Sec. introduces the algorithm to compute virtual planar coordinates for each sensor node of a giving network. Based on the computed virtual planar coordinates, resilient greedy routing is introduced in Sec. . Later, we analyze the time complexity and communication cost in Sec. .

Preprocessing

Given a wireless sensor network deployed on a high genus surface, we apply the algorithm proposed in [Zhou et al. \(2011\)](#) to extract a triangular mesh denoted as M from the connectivity graph of the network based on locally measured distances between nodes within one-hop communication range. Vertices of the triangular mesh are the set of sensor nodes. An edge between two neighboring vertices indicates the communication link between the two sensors. The process is local and has no constraint on communication models.

For densely deployed sensor nodes, we assume each face of the extracted mesh an n -polygon with $n = 3$. However, we allow the mesh with non-triangular faces, n -polygons with $n > 3$, due to the randomness of sensor deployment. We consider these non-triangular faces as holes. We will show later that these holes won't affect the computation of uniformization metric and resilient greedy routing.

The topology of the surface of a 3D setting, i.e., the genus number, could be available before the deployment of sensor nodes. Otherwise, we can still have the following simple algorithm to automatically detect the surface network topology.

The algorithm starts from one randomly chosen vertex v_i of M . v_i can be the one with the smallest node ID. v_i initiates a package with three counters v_n , e_n , and f_n and sets them as: 1,0,0 respectively. v_i marks itself and sends the packet to one of its direct neighbors, denoted as v_j . v_j marks itself and adds v_n by one. v_j checks its neighboring edges. For one with both ending vertices marked (e.g., e_{ij}), v_j adds e_n by one. v_j also checks its neighboring triangles. For one with three ending vertices marked, v_j adds f_n by one. v_j then sends the package to one of its unmarked neighbors. Once every vertex has been marked, v_n , e_n , and f_n in the packet have counted the total numbers of vertices, edges, and faces of M , respectively. The number of holes of M , denoted as

b , can be counted in a similar way. A non-triangular face initiates a package with a counter b set to one. The package is forwarded to neighboring faces until all faces in M have been visited. Once v_n , e_n , f_n , and b are flooded through the network, each node can easily compute the genus number of M based on Euler-Poincaré Theorem.

Theorem 1. Euler-Poincaré Theorem. *Denote v_n the number of vertices, e_n the number of edges, f_n the number of faces, g the genus number, and b the number of holes of a triangular mesh M . The following equation holds:*

$$v_n - e_n + f_n = 2 - 2g - b. \quad (2)$$

Computing Virtual Planar Coordinates

With the extracted triangular mesh and known topology of the surface, We apply discrete surface Ricci flow [Jin et al. \(2008\)](#) to compute the special set of edge lengths, the uniformization metric of M . As we discussed in Sec. , we allow the mesh with non-triangular faces, n -polygons with $n > 3$, due to the randomness of sensor deployment. For a face with $n > 3$, the algorithm simply adds one virtual vertex and virtual edges to connect the virtual vertex with all other vertices of the face. The face with $n > 3$ is then split into n triangle faces.

We refer readers to [Jin et al. \(2008\)](#) for theoretical background introduction of discrete surface Ricci flow. Here we only provide the algorithm of computing uniformization metric using discrete surface Ricci flow. The algorithm is fully distributed and each node only needs to exchange information with its one-range neighbors.

Specifically, each edge e_{ij} has its initial edge length set to unit one. Each vertex v_i is associated with a unit circle with radius denoted as $\gamma_i = 1$. The intersection angle of the two unit circles centered at v_i and v_j is set as the edge weight ϕ_{ij} at e_{ij} . In each iteration, each v_i exchanges its current γ_i with its direct neighbors and updates its adjacent edge lengths $\{l_{ij} | e_{ij} \in E\}$ according to the following Euclidean cosine law for genus one surface:

$$l_{ij}^2 = \gamma_i^2 + \gamma_j^2 + 2\gamma_i\gamma_j \cos \phi_{ij},$$

or the hyperbolic cosine law for genus larger than one surface:

$$\cosh l_{ij} = \cosh \gamma_i \cosh \gamma_j + \sinh \gamma_i \sinh \gamma_j \cos \phi_{ij}.$$

With the updated edge lengths, v_i updates its corner angles $\{\theta_i^{jk} | f_{ijk} \in F\}$ according to the following inverse Euclidean cosine law for genus one surface:

$$\theta_i^{jk} = \cos^{-1} \frac{l_{ki}^2 + l_{ij}^2 - l_{jk}^2}{2l_{ki}l_{ij}},$$

or the inverse hyperbolic cosine law for genus larger than one surface:

$$\theta_i^{jk} = \cos^{-1} \frac{\cosh l_{ki} \cosh l_{ij} - \cosh l_{jk}}{\sinh l_{ki} \sinh l_{ij}}.$$

Then v_i computes its current discrete Gaussian curvature K_i (Eqn. 1). If for every v_i , its current Gaussian curvature K_i is less than a threshold, discrete surface Ricci flow converges.

Otherwise, each v_i updates its u_i : $u_i = u_i + \delta(\bar{K}_i - K_i)$, where $u_i = \log \gamma_i$ and δ is the step length.

With the computed edge lengths when discrete surface Ricci flow converges, any node can initiate an embedding process. We assume the node with the smallest node ID, denoted as v_i . Its planar coordinates denoted as $uv(v_i)$ are set to $(0, 0)$. Then it chooses one of its direct neighbors, denoted as v_j . For genus one surface M , the planar coordinates of v_j is set to $uv(v_j) = (0, l_{ij})$. Otherwise, the planar coordinates of v_j is set to $uv(v_j) = (\tanh \frac{l_{ij}}{2}, 0)$. For vertex v_k , adjacent to both v_i and v_j , it calculates the intersection points of two circles centered at v_i and v_j with radii l_{ik} and l_{jk} , respectively. For genus one surface M , the two circles are Euclidean circles; otherwise, they are hyperbolic circles. A hyperbolic circle can be easily converted to a Euclidean circle according to Eqn. 3 given in Appendix, so the computation of the intersection points of two hyperbolic circles is equivalent to the computation of the intersection points of two Euclidean circles. v_k then chooses one intersection point as its planar coordinates such that v_i , v_j , and v_k satisfy right-hand rule in plane. With f_{ijk} embedded in plane, we continue to embed adjacent triangles with f_{ijk} into plane. Such breadth-first embedding continues until each vertex of M has at least two sets of planar virtual coordinates.

Resilient Greedy Routing

With the stored virtual planar coordinates at each node, the whole network is embedded in plane seamlessly. Locally, any region of the network is mapped to plane one-to-one and continuously. The algorithm then removes all the virtual vertices added in Sec. and considers faces with $n > 3$ holes. For edges shared by only one triangle face, they are labeled as boundary edges, otherwise, non-boundary edges. Vertices connecting with boundary edges are labeled as boundary vertices.

Given a pair of source and target nodes denoted as S and T respectively, considering both S and T having more than one set of planar coordinates, S calculates the Euclidean/Hyperbolic distances of all combinations and chooses the closest pair as the source and target coordinates. S attaches the target coordinates to a packet and then applies GPSR (Greedy Perimeter Stateless Routing) [Karp and Kung \(2001\)](#) to forward it. Specifically, in greedy forwarding mode, a packet is forwarded to next hop with shortest Euclidean/Hyperbolic distance to T . When greedy forwarding encounters holes or local minimums, the packet is changed to perimeter mode. In perimeter model, if the current stuck node is a boundary node, the packet is forwarded along the boundary in counter clockwise direction until greedy forwarding is achievable. Otherwise, the packet is stuck at a local minimum. It is forwarded to the next node of the current triangle face following counter clockwise direction.

At the same time, each node periodically updates its neighboring nodes and local mesh structure due to potential node or link failures. However, the resulting mesh may exist some dangling vertices and edges as shown in Fig. 3 that violate a triangular mesh structure. A dangling vertex is a vertex belonging to more than one boundaries. The existence of a dangling vertex confuses the orientation of

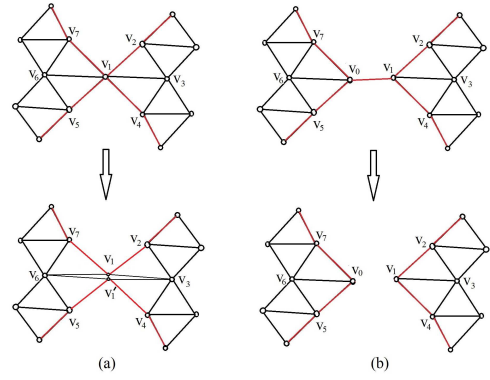


Figure 3. (a) Dangling vertex: Vertex v_1 belongs to two disconnected boundaries with boundary edges marked with red color. After vertex split, a boundary vertex belongs to one boundary only. (b) Dangling edge: Edge e_{01} with ending vertices v_0 and v_1 belongs to two disconnected boundaries with boundary edges marked with red color. After edge removal, the two boundaries are merged to one. Now a boundary edge belongs to one boundary and is adjacent with one triangle.

boundaries. We remove dangling vertices by vertex split operation. As shown in Fig. 3 (a), v_1 is connected with boundary vertices v_2 , v_4 , v_5 , and v_7 that belong to two disconnected boundaries with boundary edges marked with red color. We virtually split v_1 and denote the newly added node as v'_1 . v'_1 shares the same planar coordinates as v_1 . v_1 is then connected with v_7 and v_2 , and v'_1 is connected with v_4 and v_5 . Similarly, a dangling edge is an edge belonging to more than one boundary and not shared by any triangle face. We remove dangling edges by edge removal operation. As shown in Fig. 3 (b), e_{01} belongs to two disconnected boundaries. We remove this edge such that the previously disconnected boundaries are merged into one. The graph now is a triangular mesh with each face a triangle. A boundary vertex or edge belongs to exactly one boundary. Therefore, boundary edges and vertices along newly generated holes can be easily detected and labeled.

Time Complexity and Communication Cost

Given a network with n nodes, the time complexity to extract a triangular mesh M from the connectivity graph of the network is $O(n)$. The time complexity to detect the network topology is $O(n)$ too. The time complexity to compute uniformization metric of M using discrete surface Ricci flow is measured by the number of iterations. It is given by $-C \frac{\log \epsilon}{\lambda}$, where C is a constant, ϵ is the threshold of curvature error, and λ is the step length of each iteration [Jin et al. \(2008\)](#). We set ϵ and λ to $1e-5$ and 0.1 , respectively. The total number of iterations of each network model is no more than hundreds in our simulations. The time complexity to seamlessly embed M in the plane is also linear to n .

The majority of communication cost, measured by the number of messages, comes from computing uniformization metric using discrete surface Ricci flow. It is $O(-C \frac{\log \epsilon}{\lambda})nd$, where d is the average number of vertex degree since each vertex needs to exchange messages with its direct neighbors in each iteration.

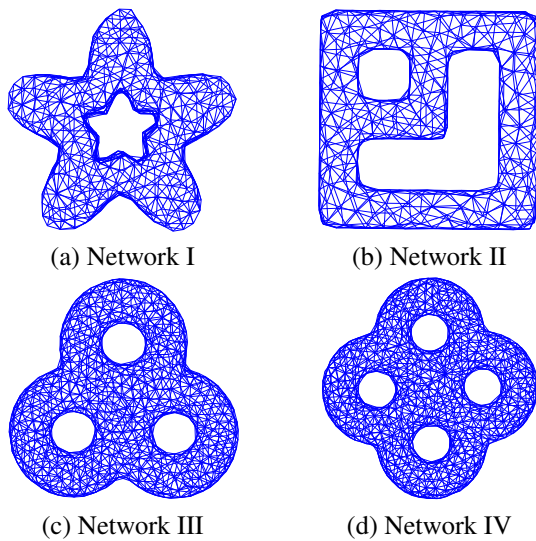


Figure 4. Network models with extracted triangular structure.

In Sec. , we apply the normalized communication cost to measure the consumed energy of a network to conduct a routing. We calculate the average number of messages exchanged in the network during one communication, whether it succeeds or not. The results show that the communication cost of our proposed resilient greedy routing algorithm is $O(n)$ for any pair of source and target nodes based on the precomputed virtual planar coordinates. It is well scalable to the size of a network.

Simulations

The proposed resilient routing algorithm can be applied for a wide range of large-scale sensor networks densely deployed on surfaces of coal mine tunnels for disaster prevention and rescue, or corridors of buildings for fire detection, or sewer or gas systems for monitoring underground pipelines. These surfaces generally have complex shapes and multiple handles, i.e., genus numbers. To this end, we create several representative network models as shown in Fig. 4 to carry out simulations. These network models have various genus numbers, ranging from one to four. Their sizes also vary from 500 to 1100 nodes.

In our simulations, we check two different failure models of a network: independent node failure, where each node in a network has an equal and independent probability of failure during some time interval; and geographically correlated failure, where all nodes within a fixed radius or hop count fail simultaneously and the center of failure region is along the primary path.

We assume the source and target nodes can locate anywhere in a network. We use three metrics to evaluate the performance of a resilient routing algorithm. The first one is the successful delivery ratio. It measures the probability of an algorithm successfully finding a path when the nodes or links of a network are not reliable. The successful delivery ratio indicates the resilience of a routing algorithm. The second metric is the average stretch factor. We divide the end-to-end hop count from source to target including the cost of forwarding messages and local detours, by the hop count of a

shortest path without failure triggered. The third metric is the normalized communication cost. We calculate the average number of messages exchanged in a network during one communication regardless of the communication succeeds or not. This metric reflects the consumed energy of the network to conduct one (successful/failed) communication and the durability of the system. For those broadcasting based methods, broadcasting dominates the communication cost of a newly established pair of source-target nodes. However, different source-target pairs sharing the same source node need only one-time broadcasting to build various paths from one source to different target nodes. We define source-to-pairs ratio as the reciprocal of the number of source-target pairs sharing the same source. To be fair, we set two different source-to-pairs ratios: 1 : 100 and 1 : 400 in our simulations.

Since the primary target of the proposed method is network resilience under independent node failure model, we compare our method with other well-known resilient routing methods that choose local detour instead of disjoint alternative paths, including Directed Diffusion [Intanagonwiwat et al. \(2000\)](#), Braided Multipath Routing [Ganesan et al. \(2001\)](#), and Robust Cooperative Routing [Huang et al. \(2008\)](#).

Besides, SINUS method [Yu et al. \(2013\)](#) is the most recent greedy routing algorithm specifically designed for networks on high genus surfaces. We implement it with GPSR module to enhance its resilience against node failures and consider it as another comparison method.

We randomly choose 20000 source-target pairs on each network model. Simulation results given in Secs. and show that our method achieves the best performance under both independent node and geographically correlated failure models.

Independent Node Failure

The first row of Fig. 5 compares the successful delivery ratio of different methods as a function of the probability of independent node failure for networks shown in Fig. 4. As the probability increases, the successful delivery ratios of Directed Diffusion, Braided, and Robust Cooperative Routing methods decrease dramatically. When the probability of independent node failure is 15%, more than half of the sampled source-target pairs of these methods fail. However, SINUS with GPSR and our method perform much better. As the probability of independent node failure continuously increases, our method shows a bigger advantage. The reason is that SINUS generates a planar embedding with uneven density. Failure nodes can easily cause isolated regions around the source or target nodes such that it is impossible to find a successful detour with GPSR in this situation.

For the same set of network models, the second row of Fig. 5 shows the average stretch factors of successful deliveries under different probabilities of individual node failure. As we can see, our method has a very low stretch factor, while SINUS with GPSR generates much longer paths in the network. We have discussed the reason in Sec. . SINUS cut the original surface network open along computed virtual boundaries to an annulus and then embed the annulus to plane. A pair of nodes with the shortest path crossing a virtual boundary in the original network needs to take a much longer detour to reach each other based on the computed

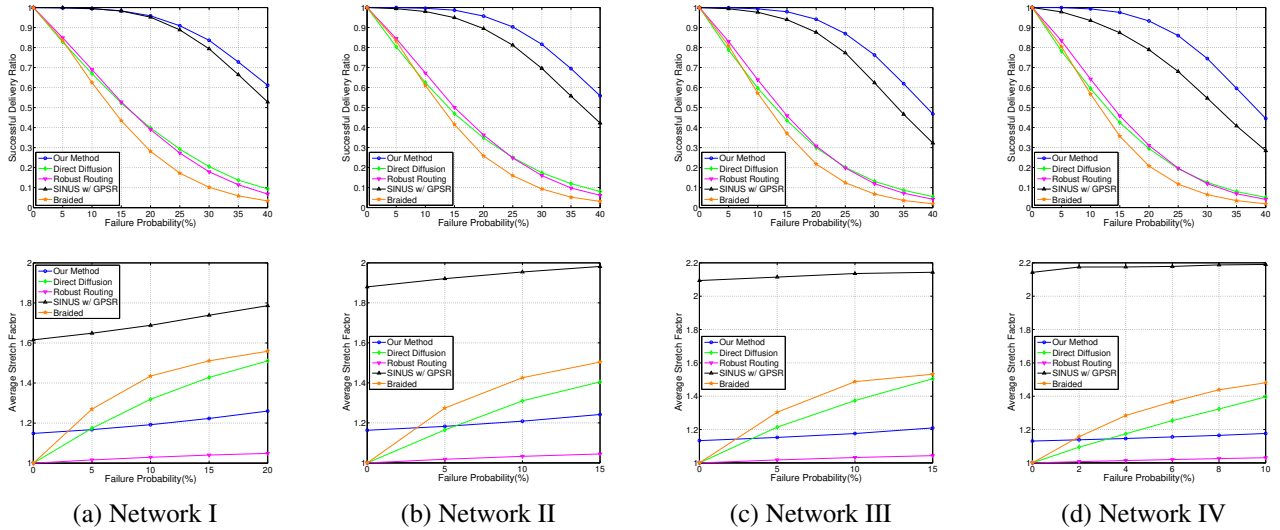


Figure 5. Independent node failure: the first row gives the successful delivery ratio as a function of the probability of independent node failure. The second row shows the average stretch factors under different probabilities of independent node failure.

virtual planar coordinates. We set the maximum probability of individual node failure for comparison of the average stretch factor as 10% – 20% only. If the failure probability is higher, the successfully delivered paths of comparison methods are quite short since longer paths are more likely to fail. To avoid such biased influence on the comparison of the average stretch factor, we compare different methods when there is still a significant number of successfully delivered paths.

We set the probability of independent node failure as 10% when measuring the communication cost. Fig. 6(a) and (b) show the normalized total/successful/failed communication costs under independent node failure model when the source-to-pairs ratio is 1 : 400 and 1 : 100, respectively. As we can see, even with a smaller source-to-pairs ratio, Directed Diffusion, Braided, and Robust Cooperative Routing spend much more energy than our method. When the ratio is larger, the energy consumption of these methods is more dramatic. Compared with SINUS with GPSR, our method consumes around 1/3 less energy for successful delivery in Network I&II. The difference is bigger when a network has more nodes and higher genus number. Besides, our method wastes much less for unsuccessful communications. Therefore, our method benefits the system a lot, especially for those time-sensitive applications that involve frequent communications in a network.

Geographically Correlated Failure

The first row of Fig. 7 compares the successful delivery ratios of different methods as a function of the radius of a failure region. As the radius of a failure region increases, the delivery ratios of Directed Diffusion, Braided, and Robust Cooperative Routing decrease dramatically. However, our method consistently performs the best with the highest successful delivery ratio, even when the failure region is huge. The only exception is the Network II model. When the radius of a failure region is larger than 3 in Network II model, both SINUS with GPSR and our method drop dramatically.

The reason is that Network II model has narrow handles with fewer nodes covered compared with other network models. When the primary path of a pair of source and target nodes passes along a handle, a large failure region can easily break the handle such that a local detour cannot find a successful alternative. It is obvious that when the radius of a failure region becomes huge in geographically correlated failure model, a routing method with high resilience needs to take the global topology information of a network into consideration and choose a path with different homotopy type, i.e., a path along different handles.

The second row of Fig. 7 shows the average stretch factors of successful deliveries under different radii of a failure region. Since the successful delivery ratio of Robust Cooperative Routing method drops to zero when the radius of a failure region is 1, we remove it from the comparison. As the radius of a failure region becomes larger, the successful delivery ratio of Directed Diffusion and Braided methods drop dramatically, and the successfully delivered paths are short ones since the longer paths all fail. To avoid a biased influence on the computation of average stretch factor, we set the maximum radius of a failure region as 2 only. With the increase of the radius of a failure region, the stretch factor of a successful routing path gets longer for all methods. However, our method still has a consistently lower stretch factor.

We set the radius of a failure region as 1 when measuring the communication cost. Fig. 8(a) and (b) show the normalized total/successful/failed communication cost under geographically correlated failure model when the source-to-pairs ratio is 1 : 400 and 1 : 100, respectively. Since the successful delivery ratio of Robust Cooperative Routing method drops to zero when the radius of a failure region is 1, we remove it from the comparison. The results are similar to those under the independent node failure model. Our method consistently performs the best among all.

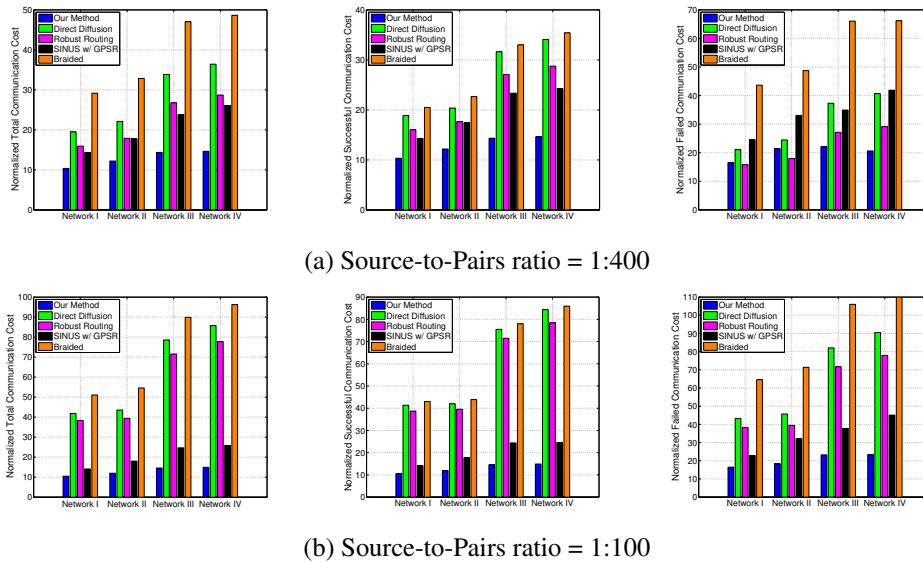


Figure 6. Independent node failure: the probability of independent node failure is set as 10%. The first row gives the normalized total/successful/failed communication costs when the source-to-pairs ratio is 1 : 400. The second row shows the costs when the source-to-pairs ratio is 1 : 100.

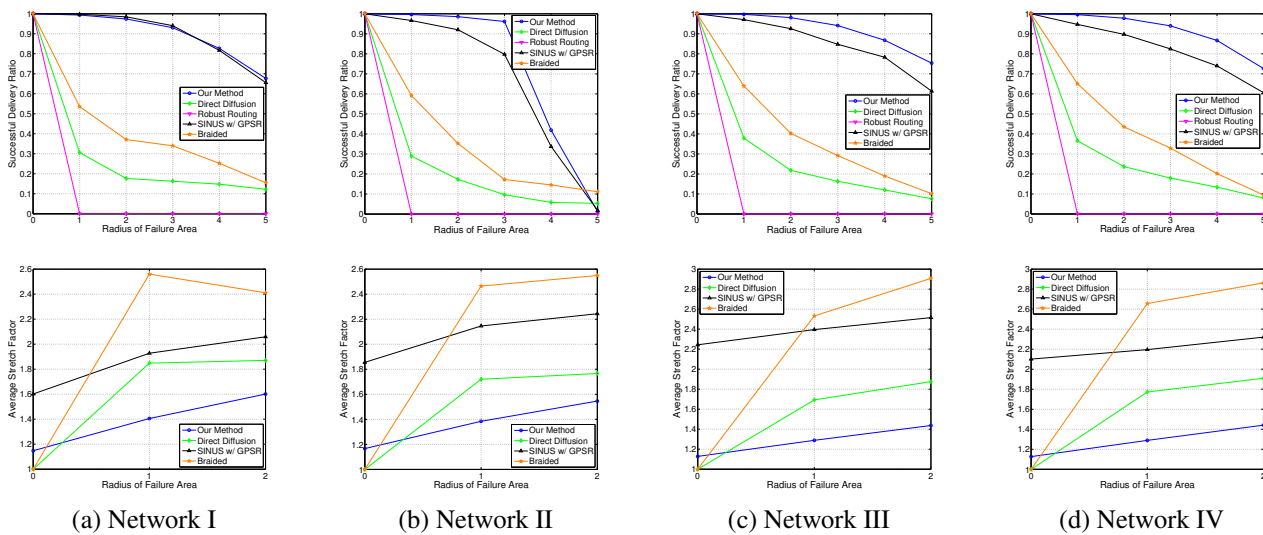


Figure 7. Geographically correlated failure: the first row gives the successful delivery ratios of different methods as a function of the radius of a failure region. The second row shows the average stretch factors under different failure radii.

Conclusion

We have proposed a cut-free resilient greedy routing scheme for GPS-free large-scale wireless sensor networks deployed on surfaces of complex-connected 3D settings. It is designed to be resilient to node or link failures especially under random node or link failure model. Simulation results have shown that the proposed scheme achieves a higher successful delivery ratio, a lower average stretch factor, and a lower normalized communication cost compared with other resilient routing methods.

References

Akyildiz IF, Su W, Sankarasubramanian Y and Cayirci E (2002) A survey on sensor networks. *IEEE Communications Magazine*

40(8): 102–114. DOI:10.1109/MCOM.2002.1024422.

Allred J, Hasan AB, Panichsakul S, Pisano W, Gray P, Huang J, Han R, Lawrence D and Mohseni K (2007) Sensorflock: an airborne wireless sensor network of micro-air vehicles. In: Jha S (ed.) *SenSys*. ACM. ISBN 978-1-59593-763-6, pp. 117–129. URL <http://dblp.uni-trier.de/db/conf/sensys/sensys2007.html#AllredHPPGHLM07>.

Atlas A and Zinin A (2008) Basic specification for ip fast reroute: Loop-free alternates.

Bose P, Morin P, Stojmenovic I and Urrutia J (1999) Routing with Guaranteed Delivery in Ad Hoc Wireless Networks. In: *Proc. of Third Workshop Discrete Algorithms and Methods for Mobile Computing and Communications*. pp. 48–55.

Cui JH, Kong J, Gerla M and Zhou S (2006) The challenges of building mobile underwater wireless networks for aquatic

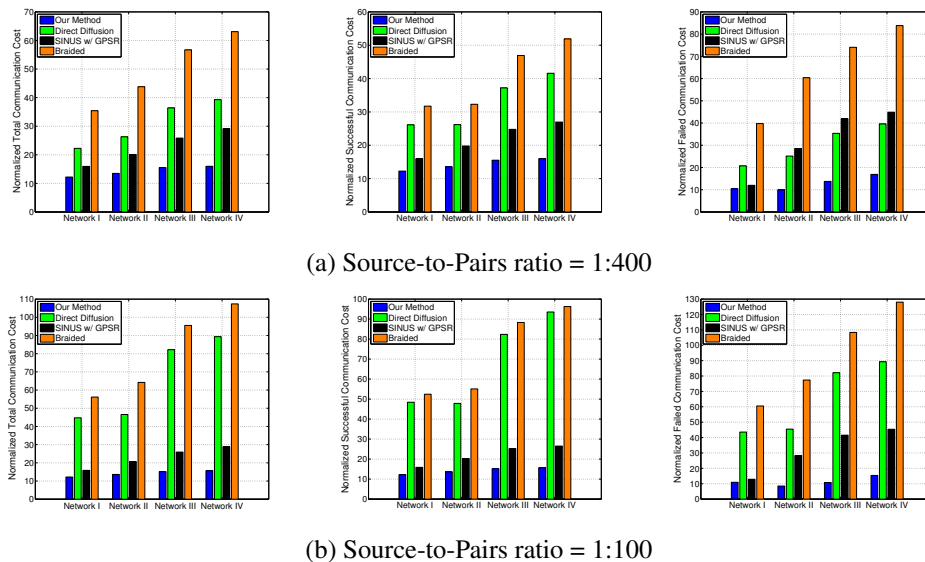


Figure 8. Geographically correlated failure: the failure radius is set as 1. The first row gives the normalized total/successful/failed communication costs when the source-to-pairs ratio is 1 : 400. The second row shows the costs when the source-to-pairs ratio is 1 : 100.

- applications. *IEEE Network* 20(3): 12–18. DOI:10.1109/MNET.2006.1637927.
- Durocher S, Kirkpatrick D and Narayanan L (2008) *On Routing with Guaranteed Delivery in Three-Dimensional Ad Hoc Wireless Networks*. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-540-77444-0, pp. 546–557. DOI:10.1007/978-3-540-77444-0_58. URL https://doi.org/10.1007/978-3-540-77444-0_58.
- Frey H and Stojmenovic I (2006) On Delivery Guarantees of Face and Combined Greedy-face Routing in Ad Hoc and Sensor Networks. In: *Proc. of MobiCom*. pp. 390–401.
- Ganesan D, Govindan R, Shenker S and Estrin D (2001) Highly-resilient, energy-efficient multipath routing in wireless sensor networks. *SIGMOBILE Mob. Comput. Commun. Rev.* 5(4): 11–25.
- Gao J and Zhou D (2011) Resilient and low stretch routing through embedding into tree metrics. In: *Proc. of WADS*. pp. 438–450.
- Huang X, Zhai H and Fang Y (2008) Robust cooperative routing protocol in mobile wireless sensor networks. *IEEE Transactions on Wireless Communications* 7(12): 5278–5285. DOI:10.1109/T-WC.2008.060680.
- Intanagonwiwat C, Govindan R and Estrin D (2000) Directed Diffusion: a Scalable and Robust Communication Paradigm for Sensor Networks. In: *Proc. of MobiCom*. pp. 56–67.
- Jin M, Kim J, Luo F and Gu X (2008) Discrete surface ricci flow. *IEEE Transactions on Visualization and Computer Graphics* 14(5): 1030 – 1043.
- Karp B and Kung H (2001) GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In: *Proc. of MobiCom*. pp. 1–12.
- Kim S, Pakzad S, Culler D, Demmel J, Fenves G, Glaser S and Turon M (2007) Health monitoring of civil infrastructures using wireless sensor networks. In: *2007 6th International Symposium on Information Processing in Sensor Networks*. pp. 254–263.
- Kranakis E, Singh H and Urrutia J (1999) Compass Routing on Geometric Networks. In: *Proc. of Canadian Conference on Computational Geometry (CCCG)*. pp. 51–54.
- Kuhn F, Wattenhofer R, Zhang Y and Zollinger A (2003a) Geometric Ad-hoc Routing: Theory and Practice. In: *Proc. of The 22nd ACM Symposium on the Principles of Distributed Computing*. pp. 63–72.
- Kuhn F, Wattenhofer R and Zollinger A (2003b) Worst-case Optimal and Average-case Efficient Geometric Ad-hoc Routing. In: *Proc. of MobiHOC*. pp. 267–278.
- Liu C and Wu J (2009) Efficient geometric routing in three dimensional ad hoc networks. In: *IEEE INFOCOM 2009*. pp. 2751–2755. DOI:10.1109/INFCOM.2009.5062225.
- Liu X, Liu A, Wang T, Ota K, Dong M, Liu Y and Cai Z (2020) Adaptive data and verified message disjoint security routing for gathering big data in energy harvesting networks. *Journal of Parallel and Distributed Computing* 135: 140–155.
- Lou W (2005) An efficient n-to-1 multipath routing protocol in wireless sensor networks. In: *Proc. of MASS*. pp. 664–672.
- Mitra BLS and Liskov B (2005) Path Vector Face Routing: Geographic Routing with Local Face Information. In: *Proc. of ICNP*. pp. 147–158.
- Motiwala M, Elmore M, Feamster N and Vempala S (2008) Path splicing. *SIGCOMM Comput. Commun. Rev.* 38(4): 27–38.
- Ohara Y, Imahori S and Meter RV (2009) MARA: Maximum Alternative Routing Algorithm. In: *Proc. of INFOCOM*. pp. 298 – 306.
- Ray S, Guérin R, Kwong KW and Sofia R (2010) Always acyclic distributed path computation. *IEEE/ACM Transactions on Networking* 18(1): 307–319.
- Reichert C, Glickmann Y and Magedanz T (2005) Two routing algorithms for failure protection in ip networks. In: *Proc. of ISCC*. pp. 97 – 102.
- Schlag W (2014) *A Course in Complex Analysis and Riemann Surfaces*. Amer Mathematical Society.
- Shand M and Bryant S (2009) Ip fast reroute framework.

- Shen Y, Yang P, Zhang P, Luo Y, Mei Y, Cheng H, Jin L, Liang C, Wang Q and Zhong Z (2013) Development of a multitype wireless sensor network for the large-scale structure of the national stadium in china. *International Journal of Distributed Sensor Networks* 2013: 1–16.
- Tan G, Bertier M and Kermarrec AM (2009) Visibility-Graph-based Shortest-Path Geographic Routing in Sensor Networks. In: *Proc. of INFOCOM*. pp. 1719–1727.
- Wang Y, Yi CW and Li F (2008) *Delivery Guarantee of Greedy Routing in Three Dimensional Wireless Networks*. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-540-88582-5, pp. 4–16. DOI:10.1007/978-3-540-88582-5_4. URL https://doi.org/10.1007/978-3-540-88582-5_4.
- Xia S, Ding N, Jin M, Wu H and Yang Y (2013) Medial axis construction and applications in 3d wireless sensor networks. In: *2013 Proceedings IEEE INFOCOM*. pp. 305–309. DOI: 10.1109/INFOCOM.2013.6566784.
- Xia S, Wu H and Jin M (2014) Trace-routing in 3d wireless sensor networks: A deterministic approach with constant overhead. In: *Proceedings of the 15th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc '14*. New York, NY, USA: ACM. ISBN 978-1-4503-2620-9, pp. 357–366. DOI:10.1145/2632951.2632977. URL <http://doi.acm.org/10.1145/2632951.2632977>.
- Yu T, Jiang H, Tang G, Wang C, Tian C and Wu Y (2013) Sinus: A scalable and distributed routing algorithm with guaranteed delivery for wsns on high genus 3d surfaces. In: *INFOCOM'13*. pp. 2175 – 2183.
- Yu X, Yin X, Han W, Gao J and Gu X (2012) Scalable routing in 3d high genus sensor networks using graph embedding. In: *INFOCOM'12, mini-conference*. pp. 2681–2685.
- Zeng W, Sarkar R, Luo F, Gu X and Gao J (2010) Resilient routing for sensor networks using hyperbolic embedding of universal covering space. In: *Proc. of INFOCOM*. pp. 1694–1702.
- Zhao MC, Lei J, Wu MY, Liu Y and Shu W (2009) Surface coverage in wireless sensor networks. In: *IEEE INFOCOM 2009*. pp. 109–117. DOI:10.1109/INFOCOM.2009.5061912.
- Zhao Y, Wu H, Jin M and Xia S (2012) Localization in 3d surface sensor networks: Challenges and solutions. In: *2012 Proceedings IEEE INFOCOM*. pp. 55–63. DOI:10.1109/INFOCOM.2012.6195798.
- Zhou H, Wu H, Xia S, Jin M and Ding N (2011) A Distributed Triangulation Algorithm for Wireless Sensor Networks on 2D and 3D Surface. In: *Proc. of INFOCOM*. pp. 1053–1061.

Euclidean circle (\mathbf{C}, R) with

$$\mathbf{C} = \frac{2 - 2\mu^2}{1 - \mu^2|\mathbf{c}|^2} \mathbf{c}, \quad R^2 = |\mathbf{C}|^2 - \frac{|\mathbf{c}|^2 - \mu^2}{1 - \mu^2|\mathbf{c}|^2},$$

where $\mu = \frac{e^r - 1}{e^r + 1}$.

Appendix

Hyperbolic Geometry

The hyperbolic model we used in this paper is the Poincaré disk. It is a unit disk on the complex plane, with Riemannian metric

$$ds^2 = \frac{4dwd\bar{w}}{(1 - \bar{w}w)^2}.$$

In the Poincaré disk, the geodesics are circle arcs which are orthogonal to the unit circle;

A hyperbolic circle can be easily converted to a Euclidean circle because a hyperbolic circle (\mathbf{c}, r) coincides with an