Journal of Network and Computer Applications 108 (2018) 53–63

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Interconnecting isolated network segments through intermittent links

Wassila Lalouani [a,*], Mohamed Younis [b], Nadjib Badache [c]

[a] High National School of Computing Science, Algiers, Algeria
[b] Dept. of Computer Science and Elect. Eng., University of Maryland, Baltimore County, USA
[c] Department of Theories and Computer Engineering, CERIST, Algiers, Algeria

ARTICLE INFO

Keywords:
Sensor networks
Connectivity restoration
Failure recovery
Mobile data carriers
Federating network segments

ABSTRACT

Wireless Sensor Network (WSN) deployed within hostile environments may suffer from large scale damage where many sensors fail simultaneously and cause the WSN to split into disconnected segments. Restoring inter-segment connectivity is primordial to the effectiveness of the WSN. When it is not feasible to replace the lost nodes, a set of mobile relays is often employed in order to establish intermittent connectivity between segments. Basically these relays serve as mobile data carriers (MDCs) that tour segments to transport the data. A key objective is to minimize the tour length in order to limit the travel overhead and data latency. Existing solutions have simplified the optimization problem by representing each segment by just one terminal and ignoring the shape and the size of the individual segments. In this paper, we consider the recovery optimization under realistic segment topology and constrained number of MDCs, which make the problem very challenging. We present a two-step heuristic for Connecting Isolated Segments through Intermittent Links (CISIL). CISIL first uses a high order Delaunay triangulation to efficiently determine all efficient tours among segments, and then selects the optimal subset of these tours that matches the MDC count and yields a strongly connected network. The selection optimization is mapped to a k-edge minimum spanning tree problem within a hypergraph. The performance of CISIL is validated through simulation and compared to a prominent competing scheme.

1. Introduction

Wireless sensor network (WSNs) can be beneficial in many applications like forest monitoring, target tracking, and battlefield surveillance. In many applications, the sensor nodes are deployed within hostile environments which make them highly susceptible to failure. For example a WSN serving in a battlefield could be impacted by explosive; similarly nodes could fail due to any disastrous event like forest fires. The severity of the events in these scenarios may be so significant that the scope of the damage includes many collocated sensors and causes the network to be fragmented into isolated segments (Younis et al., 2014). Restoring connectivity after these events and with limited resources is critical for the continuation of the WSN services. Contemporary solutions either deploy additional nodes or identify some of the survived nodes in the network to be reassigned a different role without negatively affecting the application. For example, nodes may be relocated (Joshi and Younis, 2012), or additional relays are deployed (Senel and Younis, 2016; Lee and Younis, 2012; Cheng et al., 2008; Singh and Al-Turjman, 2016; Lloyd and Xue, 2007) to restore connectivity by forming stable inter-segment links.

However, when the number of available nodes, within the network or externally supplied, is insufficient for forming a connected inter-segment topology, existing solutions (Senel and Younis, 2012; Zhao et al., 2004; Kalyanasundaram and Younis, 2013; Stanislaus and Younis, 2012; Abbas and Younis, 2013; El-Moukaddem et al., 2013; Almasaeid and Kamal, 2007, 2008; Abuarqoubet al, 2017; Wu and Tseng, 2013; Jea et al., 2005; Jain et al., 2006; Moazzez-Estanjini et al., 2013; Ang et al., 2017; Alsalih et al., 2010) tend to establish intermittent links by employing multiple mobile data carriers (MDCs). Basically, these MDCs tour the individual segments to transport data from one segment to another. Every MDC has to tour part of the area such that some of the network segments along the MDC's travel path are visited regularly to collect and transport data from and towards its designed segments. The tours are to be formed such that each segment could be reachable to every other segment in the WSN. In essence the segments are federated rather than tightly-coupled in this case. The number of tours should be determined based on the number of available MDCs, k. As the movement of the MDCs incurs significant energy overhead that diminishes their lifespan, which has a direct impact on the quality of the federation service for the application, the main objective of the interconnection solution is to find the shortest k-tours.

Published heuristics that form k-tours to restore connectivity

* Corresponding author.
E-mail addresses: w_lalouani@esi.dz (W. Lalouani), younis@cs.umbc.edu (M. Younis), badache@mail.cerist.dz (N. Badache).

https://doi.org/10.1016/j.jnca.2018.02.010
Received 21 September 2017; Received in revised form 21 December 2017; Accepted 17 February 2018
Available online 21 February 2018
1084-8045/

generally formulate the problem by representing each segment by one terminal. Basically, the network is modeled as a general graph of terminals and then clustering techniques are exploited to deal with the MDC count constraint. However, the topological properties of the damaged area as well as the size and shape of segments are not factored in when forming the tours. Such a limitation significantly detriments the quality of existing solutions since in practice a segment could be accessible to multiple MDCs using distinct boundary nodes and could exploit intra-segment links to disseminate the data within the segment. As illustrated in Fig. 1, the consideration of boundary nodes significantly reduces the length of the tours and the energy required for MDCs displacement. The figure illustrates clearly the advantage of exploiting the exact segment topology in order to federate the network. To overcome the aforementioned shortcoming of the existing solutions, this paper proposes a novel approach for boundary-aware tour formation. Specifically, we focus on the interconnection problem using k MDCs necessary for forming inter-segment topology while considering multiple interfaces for each segment. To do so, we represent each segment by a simple polygon delimited by its boundary nodes. To the best of our knowledge such a federation problem has not been investigated before in the literature.

Although our formulation of the federation problem is more practical by reflecting the exact topological proprieties of the damaged network, it makes the federation problem more challenging as it is both constrained by the number of MDCs and the topological proprieties of the damaged area. Basically, forming k shortest tours can be mapped to the problem of k travel salesman problem, which is known to be NP-hard (Bektas, 2006). Nonetheless, in addition to partitioning the segments into k sets that correspond to the shortest tours, in our formulation the complexity of forming the individual tours is higher as it depends not only on the number of involved segments but also on the location of nodes on the boundaries of these segments. Our heuristic approach for Connecting Isolated Segments through Intermittent Links (CISIL) opts to tackle such complexity by pursuing a two-step process, namely, forming a set of efficient tours among the various subset of segments, and then selecting a subset of these tours that minimizes the total MDC travel distance and ensures inter-segment connectivity.

Since there are exponential number of ways for partitioning n objects, a brute-force search to find all feasible tours for each distinct subset of segments with their respective boundary nodes incurs excessive computational complexity that is exponential in the number of boundary nodes and segments. Therefore, CISIL uses high-order Delaunay diagrams to exploit the neighboring relationship between boundaries nodes and segments. Such an approach enables CISIL to incrementally find the set of compatible segments and boundary nodes and can thus lead to major complexity reduction as it will be discussed in Section 5. Specifically, CISIL constructs the first order Delaunay triangulation (DT) and uses it to form tours of two and three segments, then constructs second order DT and uses it to grow the tours by combing the nearest segments, and follows that by the third order DT, and so on. Once efficient tours are determined for all cardinality of segment subsets, CISIL opts to identify the k shortest tours that span and connect segments. We show that such a tour selection problem corresponds to the k-minimum length tree for the hypergraph that has segments as vertices and the formed tours as hyperedges. CISIL formulates such a tree formation problem as a mixed integer program and applies the branch-and-cut technique to find the set of k tours that federate the network. The simulation results confirm the effectiveness of CISIL and its performance advantages over competing schemes.

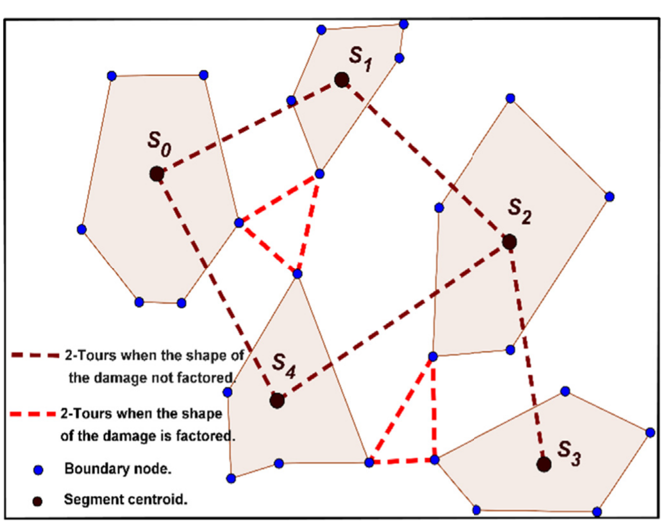

Fig. 1. Comparing the length of two MDCs tours for federating the five segments (i) using the centroid, and (ii) the boundary nodes of the segments.

In summary, the main contribution of the paper is the development of CISIL, a novel heuristic for interconnecting disjoint network segments using k mobile relays. The relays serve as MDCs that form intermittent links among the segments. To ensure strong connectivity, CISIL overlaps some tours to enable data relaying across multiple MDCs. CISIL is distinct by the fact that it factors in the boundary nodes in forming shortest tours that covers all segments. The effectiveness of CISIL is validated through mathematical analysis and simulation. The rest of the paper is organized as follows. The next section compares CISIL with related work in the literature. Section 3 discusses the system model, states the assumptions and analyzes the problem. CISIL is explained in Section 4. The runtime complexity and travel overhead reduction will be analyzed in Section 5. Section 6 presents the simulation results. Finally Section 7 concludes the paper and highlights our planned extension.

2. Related work

The problem of federating disjoint network segments have been extensively researched in recent years (Younis et al., 2014). Published techniques can be classified based on the formed inter-segment topology as stable and intermittent. To form a stable topology, either the failed nodes are replaced by heathy ones, or a connected inter-segment topology is formed by deploying relays. The former can be achieved by deploying spares, or picking unessential nodes for the network operation. Meanwhile, the latter is achieved by mapping the problem to a variant of the Steiner minimum tree formation problem (Cheng et al., 2008). On the other hand, if insufficient resources are available to form inter-segment data paths, the failure is tolerated by establishing intermittent links through the use of mobile nodes as MDCs. Given the contribution of this paper, we focus on prior work on the use of mobile nodes for data transportation.

Some published work exploited the use of mobile nodes to tour certain locations in an area of interest to serve in WSNs as (i) mobile relays (Senel and Younis, 2012; Zhao et al., 2004; Kalyanasundaram and Younis, 2013; Stanislaus and Younis, 2012; Abbas and Younis, 2013; El-Moukaddem et al., 2013), which transport data between certain nodes or network segments, (ii) mobile data collectors (Almasaeid and Kamal, 2007; Abuarqoubet al, 2017; Wu and Tseng, 2013; Jea et al., 2005), which visit the individual segments and carry data to the sink, or (iii) mobile base-stations (MBSs) (Almasaeid and Kamal, 2008; Jain et al., 2006; Moazzez-Estanjini et al., 2013; Ang et al., 2017; Alsalih et al., 2010). However, these approaches do not consider the travel path optimization when multiple mobile nodes are involved. While a key objective of algorithms in this category is to find the shortest tour along which the node visits a set of locations, the delay and buffer size are the main concerns in quite of few of them. This particularly applies to work on mobile data collectors and significantly affects the solution strategy. To illustrate, if the mobile node cannot store more than a certain volume of data, it will be forced to go to the destination(s) to offload the data payload (Jea et al., 2005). That makes the optimization to be derived by the data volume rather than the travel distance. In addition, the data delivery delay makes it necessary to provision rendezvous points

(Kalyanasundaram and Younis, 2013; Moazzez-Estanjini et al., 2013). Assuming no buffer constraints, CISIL focuses on shortening the tour length of the mobile relays, which also contributes to reducing the data delivery latency.

On the other hand, mobile nodes have been used for collecting data in an energy-efficient and reliable manner when routes between some nodes are not available. For example, in (Moazzez-Estanjini et al., 2013) the authors have proposed energy efficient data collection protocols in single-hop WSNs by employing data MULEs which are capable of short-range wireless communication and move in an uncoordinated manner to provide connectivity in sparse WSNs. In addition, Alsalih et al. (2010) have proposed data collection schemes which place MBSs, such as autonomous unmanned vehicles, in order to prolong the network lifetime. However, the tour length is not considered as an optimization metric in most of these schemes. Finally, grouping stops, either event location or data sources, based on proximity has been pursued in (Senel and Younis, 2012; Kalyanasundaram and Younis, 2013; Stanislaus and Younis, 2012; Abbas and Younis, 2013) as a means for minimizing the MDC tour. However, these approaches deal only with terminals, even if the goal of some of them is to federate segments using k MDCs. Unlike CISIL, segment are modeled as terminals, and the node positions on the segment boundaries have not been factored in the solution. We compare the performance of CISIL to FOCUS (Kalyanasundaram and Younis, 2013), SS-WSN (Wu and Tseng, 2013), and MiMSI (Abbas and Younis, 2013) through simulation as discussed in Section 5.

In summary, existing approaches that use mobile relays to interconnect disjoint segments do not factor in the topological aspect of the problem where a segment is modeled using only one terminal with presumed location, e.g., at the centroid of the segment. We argue that ignoring the boundary of the individual segments oversimplifies the federation problem and yields longer tours than necessary. The increased tour length negatively affects the lifespan of the mobile relay and increases the data delivery latency. To the best of the authors' knowledge, CISIL is the first approach that avoids these shortcomings.

3. System model and problem statement

We consider a connected WSN that has suffered simultaneous failure of multiple collocated nodes due to external events such as a forest fire, explosion in a combat field, sand storm, etc. Consequently, the network is partitioned into a set Ψ of multiple disjoint segments (connected components), where $\Psi = \{S_1, S_2, \dots, S_n\}$. In other words, the intra-segment topology stays strongly connected. We assume that a segment is a simple polygon P inside the deployment area and is delimited by the concave hull of its nodes. To federate the segments by establishing intermittent communication links, the objective is to form tours that interconnect the n segments using k MDCs. The nodes as well as the MDCs are assumed to have the same communication range "R". The MDC is assumed to have sufficiently large buffering space to store the data to be shared among segments. This is a practical assumption given the high storage capacity of computing and communication devices nowadays. An MDC is assumed to be operating based on a limited energy supply. To extend the lifespan of the MDC and also reduce the inter-segment data delivery latency, the MDCs should achieve the federation objective through the shortest tours. Such a problem is in essence a variant of k travel salesman problem (k-TSP), which is NP-hard. Moreover, the complexity of forming the individual tours is higher in our case since it depends not only on the number of involved segments but also on the location of nodes on the boundaries of these segments. In other words, the tour formation has to identify the boundary nodes that are to be used so that the tour length is minimized. The scope of the paper is focused on the algorithmic aspect of the problem assuming unconstrained terrain, i.e., there are no obstacles in the area that MDCs have to avoid while traveling.

In addition to the inherit k-TSP complexity, in our case the tours should overlap so that data can be shared between segments that are served by different MDCs. Furthermore, tour overlap should enable the nodes of every pair of segments to communicate. In other words, if a tour is modeled as a vertex in a graph, such a graph should be strongly connected. Two tours can overlap if they share a segment or if the involved MDCs rendezvous at a specific mutually agreed upon location. The latter radically complicates the tour formation problem and also introduces an MDC travel scheduling problem for reducing the waiting delay. Therefore, CISIL pursues the first option and enables tours to share segments. To summarize, the problem that we study in this paper can be formally defined as follows: "Given the set of n segments represented by their boundary nodes and k MDCs", we should determine the shortest k tours $\{T_1, T_2, \dots, T_k\}$ such that: (1) each segment S_i should be assigned to at least one tour; (2) the k tours should enable communication between the nodes of every pair of segments that are part of two different tours, i.e., the tours should overlap such that every pair of MDCs should be able reach each other either through a segment if their tours overlap, or through other MDCs (inter-MDC path).

4. Detailed CISIL approach

CISIL pursues a two-step process, namely, forming a set of effective tours and then selecting a subset of these tours that minimizes the total MDC travel distance and ensures strongly-connected inter-segment connectivity. These steps are explained in the balance of this section.

4.1. Effective tour formation

CISIL first determines the set of effective tours. A tour "T" of a subset of segments, $\eta \subseteq \Psi$, is the shortest possible cycle that includes one collection point from each of the segments in η. Since the k-shortest tours may cover subsets of Ψ of different cardinalities, all possible subsets of Ψ with cardinalities of two or more should be considered (i.e., $\forall\eta$ such that $|\eta| \geq 2$). As pointed out earlier, finding T is equivalent to the Euclidean Traveling Salesman Problem which is known to be NP-Hard. Nonetheless, in addition we should partition the segments into sets each is covered by a tour, in our formulation the complexity of forming the individual tours is higher as it depends not only on the number of involved segments but also on the selection of the data collection points from the boundaries of those segments. A brute-force search to find all feasible tours for each subsets of segments with their respective boundary nodes incurs excessive computational complexity that is exponential in the number of boundary nodes and segments. Therefore, the tour formation is intrinsically harder to solve, compared to the case where a segment is represented just by a single terminal.

Given the exponential number of possible tours, CISIL limits the consideration to only segment combinations that are topologically neighbors. To illustrate consider the topology in Fig. 2(a). It does not make sense in this topology to get an MDC to tour S_0 and S_2 since they are distant from one another and the tour could include other segments, namely, S_5, without extra travel overhead. Therefore, CISIL minutely explores the open space between segments to infer geometric neighboring relationship. The neighboring relationship between boundary nodes will allow determining compatible segments that may be visited within the same tour and discard unfruitful combinations. To do such segment combination classification, CISIL uses Delaunay triangulation (DT) for the set of nodes on the segment boundaries.

Delaunay triangulation is a computational geometry technique with many practical applications. Delaunay triangulation of a set of points (sites) in a plane is a triangulation in which the circumcircle of each triangle does not contain any sites in its interior. Popular methodologies for forming Delaunay triangulation include Edge flipping, Incremental vertex addition, Divide-and-conquer, and Sweep-hull (de Berg et al., 2008). The Delaunay triangulation is unique and have two key properties (Borradaile and Eppstein, 2015): (1) the minimum spanning tree (*mst*) of a set of points is a subgraph of the Delaunay triangulation; (2) Every point in the plane is covered by only a few circumcircles. Therefore, according to the first property the length of the shortest tour connecting three

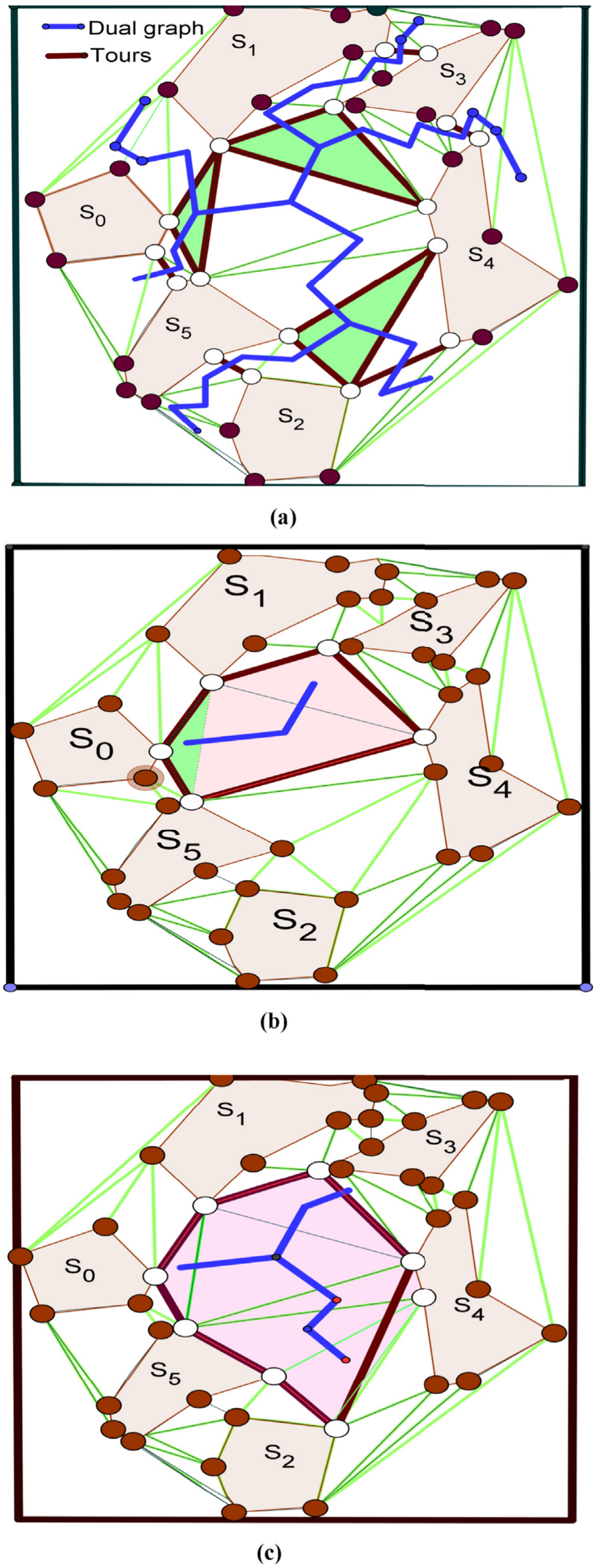

Fig. 2. **(a)**: An example 1st order feasible Tours formation. **(b):** An example 3rd order feasible Tours formation. **(c):** The *mst* of the tours hypergraph.

segments corresponds to the weight of the Delaunay triangle, where the weight refers to the length of an edge. Furthermore, the optimal tour of two segments should imperatively belong to a side of Delaunay triangulation because the shared space between segments is partitioned into triangles containing the closest sites (boundaries nodes). Even though the proprieties of Delaunay ensure that we have the optimal tour for groups of two and three segments (i.e., $|\eta| = 2$ and 3), CISIL still needs to determine all tours of higher cardinality up to n. CISIL does so using higher-order DT.

The Delaunay triangulation of the set of boundaries nodes is referred to as 1st order DT. A higher-order Delaunay gauges the proximity among more than three segments. Considering the 1st order triangulation, we form the dual graph of the triangulation in order to determine higher cardinality tours. Every triangle in the 1st order DT corresponds to a vertex in the dual graph, an edge exists if two triangles share one side, i.e., have two segments in common. The expansion of every already-formed tour to include a new segment will allow us to update the list of effective tours. To elaborate, based on the 1st order DT growing the tour for three segments that are part of DT_i can be done by considering a neighboring vertex to DT_j in the dual graph since DT_j and DT_i share an edge and the combined tour will thus involve four segments. CISIL updates the list of tours with the newly formed ones until all segments are included. In other words, the tour generation process in CISIL could be seen as DT concatenation to cover the set of segments according to their neighboring relationship. Intuitively, the higher-order DT will be a fairly natural way to find the smallest tours containing at least m segments as the DT is mainly constructed according to neighboring relationship.

From the aforementioned, the triangulation allows us to determine the set of compatible segments and the set of boundaries sensors suitable to form the optimal tours. The set of suitable boundaries nodes to form the optimal tours is the set of vertices of triangles relating the segments. It is important to note that two neighboring DT_i and DT_j in the dual graph may be interfacing a common segment S_q using two different boundary nodes x_q and y_q. Thus, the concatenation of DT_i and DT_j implies that the edge $\overline{x_q y_q}$ may be part of the combined tour. Therefore, a tour may be further optimized by picking a combination of single-collection points per segments to minimize the tour length. As the number of candidate collection nodes is very limited, successive elimination could be applied in this context.

Fig. 2 illustrates the tour formation process through an example topology of six segments. The boundary of a segment is defined by the concave hull of its nodes. A first-order Delaunay diagram defines the set of triangles by mapping each boundary node in one segment to the closest node in another segment, as indicated by the green triangles in Fig. 2(a). The two and three cardinalities tours are also highlighted by brown line in Fig. 2(a). A higher order DT can be inferred using the dual graph of the triangulation marked by the blue lines in the figure relating the centroid of the neighboring first-order Delaunay triangles. Obviously, segments S_0 and S_2 could not be connected within a tour that does not include S_5. Starting from the formed tours, we follow the dual graph of the triangulation to form the higher cardinality tours. Considering for example, the tour S_0, S_1, S_5, which is formed in the 1st order triangulation is then concatenated to S_4 and to S_3 in the second order triangulation as indicated by the blue line in Fig. 2(b). Once the neighboring segments are determined, CISIL designates a collector point from each segment that reduces the tour length.

To reduce the number of combinations of concatenated tours, we refer to the triangular pattern highlighted in pink in Fig. 2(b) that relates the segments S_1, S_0, S_3, S_4 and S_5. This triangular pattern allows us to limit the number of boundaries nodes that may serve as collection points as it groups the nearest nodes of the designated segments. In other words, we consider the vertices of the triangles within the pink pattern to reduce the number of combinations to concatenated tours. The set of unconsidered nodes when forming the optimal tour are highlighted by dark vertices in Fig. 2(b); these nodes are deemed non-optimal for connecting S_1, S_0, S_3, S_4 and S_5. The effect of the reduction in the number combinations of boundary nodes that may serve as collection points is more visible when considering the connection of all segments within a single MDC tour. So, it is also worth noting that applying DT effectively restricts the possible collection points to boundary nodes residing within the open space between the candidate segments, which in essence shortens the tour.

4.2. Selection of k tours

Given the set of effective tours, CISIL opts to identify a subset of k of them that spans and interconnects all segments with the least total travel overhead. In addition to the MDC count constraint, in our case the tours should overlap in order to enable nodes of every pair of segments to communicate. In other words, every pair of MDCs should be reachable to each other either by sharing a segment or through other MDCs. To address these constraints, traditional approaches model the segments as vertices of a graph and then partition such a graph into k connected components (clusters) (Kalyanasundaram and Younis, 2013; Stanislaus and Younis, 2012; Abbas and Younis, 2013). To ensure connectivity among all segments, selected pairs of clusters are forced to overlap. To cause two clusters to overlap, the boundary of one of them is extended so that both share a common segment. First the *mst* of the segment graph is found to ensure the connectivity and then the *mst* is partitioned into k parts, each is to be served by an MDC (Senel and Younis, 2012). However, the tours formed by these contemporary approaches could be inefficient since they do not account for the boundary nodes of the federated segments. For example, in Fig. 2(a) the shortest tour for $S_0S_2S_5$ may unnecessarily go across segment S_5.

Unlike these approaches, we attempt to provide a formalization of the k tours selection. Basically, we model the set of tours as a hypergraph H(Ψ, E) where Ψ is the set of segments, and E is the set of hyper-edges that represent the tours. A hypergraph is a generalization of a graph where an edge can connect two or more vertices. A hyper-edge (tour) is associated with each $\eta \subseteq \Psi$, and its weight reflects the length of the shortest tour connecting the segments in η. The selection of k tours that ensure inter-segment connectivity throughout the network is then mapped to the problem of finding the minimum cost k-connected hyper-subgraph that spans all segments. To ensure connectivity, we refer to the following definition.

Definition 2. (Warme, 1998): A hypergraph H is connected if for every $S_i, S_j \in S$, there is a chain from S_i to S_j in H.

Obviously, the least cost chain should not contain any cycle. Therefore, the minimum chain connecting all vertices of a hypergraph should form a tree. According to (Warme, 1998), a hyper-subgraph is a spanning tree of H if and only if it is a tree and it covers all the vertices of H. Therefore, if sufficient MDCs are available, a minimum spanning tree of all segments may be formed and MDCs can be designated to serve on the individual inter-segment links, i.e., $k = |S| - 1$. Since the MDC count is insufficient, we formulate the minimum k tours concatenation problem as an instance of the k-minimum spanning tree in a hypergraph (MSTHG). To solve such MSTHG problem, we adopt the mixed integer linear program formulation of (Warme, 1998) and factor in the number of tours, i.e., k, as a constraint. Specifically, we solve the following optimization model:

$$\sum_{e \epsilon E,} (|e| - 1) x_e = |\Psi| - 1 \tag{1}$$

$$\sum_{e \epsilon E} \max(|e \cap \eta| - 1, 0) \leq |\eta| - 1, \quad \text{where } 2 < |\eta| < |\Psi| \forall \eta \subset \Psi \tag{2}$$

Subject to

$$x_e \geq 0, e \in E \tag{3}$$

$$\sum_{e \epsilon E} x_e \leq k \tag{4}$$

Where x_e is a binary variable equal to 1 if the hyper-edge e belongs to the *mst*. The set η includes a subset of segments. Just like a connected graph of m nodes that contains at least $m - 1$ edges, constraints (1) and (2) guarantee that the *mst* is not disconnected and does not contain a cycle. To do so, constraint (1) imposes that the number of segments on all edges (tours), except the ones (|e|-1) that connect the chain is equal to $|\Psi|$-1, which reflects the number of link in the *mst*. Constraint (2) ensures that there is no cycle in the *mst* by imposing constraint (1) for every subset of vertices (segments). If there is a cycle in any subset, constraint (1) will not be met and the graph cannot be connected. Constraint (4) guarantees that the *mst* contains k edges.

As there are exponentially many instantiations of constraint (2) when the number of segments increases, an LP relaxation is proposed in (Warme, 1998) by removing constraint (3) and recovering the integrality of search space by applying branch-and-bound. Constraints (2) are added to the formulation dynamically as violations are discovered. The violation is determined based on the incompatible set of edges. In fact, the iterative construction of the n order DT allows determining the set of compatible tours and thus eliminates significant number of combinations that may never lead to a connected chain. Our tour selection formulation can be solved in polynomial time (Warme, 1998). To deal with the limited number of MDCs, we consider only solutions that have k hyper-edges. A summary of the various steps in CISIL is provided in Fig. 3; the runtime complexity will be analyzed in the next section.

```
1. Determine the Delaunay triangulation
2. Extract tours from the 1st order DT (find tours that
   involve two or three segments)
3. Determine the dual graph of the triangulation.
3. For each not expanded tour t:
4.    Expand t using the dual graph to identify neigh-
   boring triangles
5.    For each combination of t with its neighboring
   segments
6.       If there is no tour including the previous
   combination of segments
7.          Determine the candidate boundaries nodes
   (the vertices of the triangles)
8.          Optimize the discovered tours by choosing
   the best collection point in each segment for a given
   tour
```

1. Determine the Delaunay triangulation
2. Extract tours from the 1st order DT (find tours that involve two or three segments)
3. Determine the dual graph of the triangulation.
3. **For each** not expanded tour t:
4. Expand t using the dual graph to identify neighboring triangles
5. **For each** combination of t with its neighboring segments
6. **If** there is no tour including the previous combination of segments
7. Determine the candidate boundaries nodes (the vertices of the triangles)
8. Optimize the discovered tours by choosing the best collection point in each segment for a given tour
9. Keep an optimal tour for each subset of segments that has been connected by a tour at previous stages.
10. **END If**
11. **END For**
12. **END For**

// Determine the optimal k-tours.

13. Construct the hypergraph H using the generated tours.
14. Determine the k-minimum spanning tree of the hypergraph H.

Fig. 3. Pseudo code of CISIL.

```
        Keep an optimal tour for each subset of segments that has been connected by a tour at previous stages.
    END If
  END For
  END For
//Determine the optimal k-tours.
Construct the hypergraph H using the generated tours.
Determine the k-minimum spanning tree of the hypergraph H.
```

5. Approach analysis

5.1. Runtime complexity

As explained earlier, CISIL opts to tackle the complexity of connecting isolated segments through intermittent links by pursuing a two-step process, namely, (i) forming a set of efficient tours among the various subset of segments, and then (ii) selecting a subset of these tours that minimizes the total MDC travel distance and ensures inter-segment connectivity. Since the second step could be computed in polynomial time using (Warme, 1998), we will focus on analyzing the complexity of the first step. To the best of our knowledge no prior work has solved the federation problem using MDCs while considering the exact segment boundaries. Therefore, we will compare the complexity of CISIL with the brute force approach. The complexity of the problem depends mainly on the number of segments n and the largest number of boundaries nodes B that delimit the segment polygons.

Theorem1. *The runtime complexity of forming k tours using brute force is:* $\sum_{k=2}^{n}\binom{n}{k}*B^{k}*(x-1)!$.

Proof. The number of all possible tours that include x segments is equal to $\binom{n}{x}$. Since we should determine all tours with cardinality more than 2, the number of all possible tours will be $\sum_{k=2}^{n}\binom{n}{x}$. Assuming that each segment is represented by B boundary nodes, the number of all possible tours becomes $\sum_{k=2}^{n}\binom{n}{x}*B^{x}$. For each tour involving x segments we should consider the appropriate boundary node among the B choices for a segment that better optimizes the tour. To do so, we need to determine the best segment boundary nodes depending of their order within the tour, which in essence depends on forming the Hamiltonian cycle (Martello, 1983) of the x terminals where the complexity is $(x-1)!$. ■

Theorem 2. *Let P be a set of n points in the plane, not all collinear, and let q denote the number of points in P that lie on the boundary of the convex hull of P. Then any triangulation of P has* $2n-2-q$ *triangles* (Golden et al., 1980).

Theorem 3. *The runtime complexity of the tour formation step in CISIL is* $\left\{\left[(nB)^{*}\log(nB)\right]+\binom{n}{3}\left(nB-2-nb+2n\right)\quad[1+\sum_{j=4}^{n}2^{*}(B-b)^{j}]\right\}$ *where b is the average number of boundary nodes of a segment that reside on the convex hull of all segments covered by the tour.*

Proof. Assuming that each segment has B boundary nodes, of which b lie on the convex hull. CISIL first computes the Delaunay triangulation of $n.B$ points (boundary nodes for all segments) with $O(NlogN)$ runtime complexity (de Berg et al., 2008), where $N=n.B$ in our case. While computing the triangulation, the two and three cardinalities tours as well as the triangulation graph could be extracted. Then, CISIL iterates by considering every combination of three Delaunay triangles to form the overall feasible tours. With those $\binom{n}{3}$ tours, CISIL spans the dual graph of the triangulation. Since we exclude triangles that connect nodes of the same segment, using Theorem 2 we can exclude $(2B-2-B)=(B-2)$ triangles assuming that segments are convex polygons. Therefore, from the set of all triangles connecting the $n.B$ nodes (i.e., $2nB$-2-nb triangles), we remove $(B-2)n$ triangles connecting segments nodes. Indeed, CISIL will span at most $(nB-2-n.b+2n)$ triangles. In other words, in each iteration CISIL merges neighboring Delaunay triangles with complexity $\binom{n}{3}\left(2n-2-n.b\right)$.

While merging neighboring using the triangulation graph, CISIL may expand the current tour t with at most 2 new segments (as triangle have three vertices). CISIL considers the expansion of t with each segment separately. While spanning the triangles, CISIL determines also the vertices of triangles for forming optimal tours. For each distinct set of segments, CISIL selects the optimal set of boundary nodes connecting those segments. As we have only $B-b$ nodes per segment that are inside the convex hull and that may connect distinct segments, the number of all possible combinations of boundary nodes connecting j segments is $(B-b)^{j}$. In the worst case, CISIL has to choose the optimal tour for all combinations of segments and thus the complexity will be $\left\{[(nB)^{*}\log(nB)]+\binom{n}{3}^{*}(B*n+2*n-2-n.b)[1+\sum_{j=4}^{n}2^{*}(B-b)^{j}]\right\}$. ■

In contrary to the brute force approach, CISIL does not have to determine the Hamiltonian cycle as the Delaunay triangulation captures the neighboring information. Fig. 4 demonstrates the massive reduction in the complexity of the tours generation step compared to the brute force, as n and B vary, respectively. It is important to note that the y-axis is a logarithmic scale. Obviously more reduction in the tour lengths is possible as b increases since more boundary nodes are skipped due to the fact that they do not relate distinct segments. Although having a large number of boundary nodes per segment, i.e., B, diminishes the effect of the optimization, the reduction is still major compared to the brute force approach.

5.2. Travel distance overhead

A key advantage of CISIL is optimizing the MDC tours by factoring in the nodes on segment boundaries. Contemporary approaches, e.g., (Senel and Younis, 2012; Zhao et al., 2004; Kalyanasundaram and Younis, 2013; Stanislaus and Younis, 2012; Abbas and Younis, 2013; El-Moukaddem et al., 2013) consider the centroid of a segment as a placeholder for where an MDC interacts with the segment nodes and where data upload and download take place. In the balance of this subsection we analyze the effectiveness of CISIL in reducing the travel distance overhead. We note, although not explicitly targeted by our analysis, that shorter travel paths imply reduced data delivery latency.

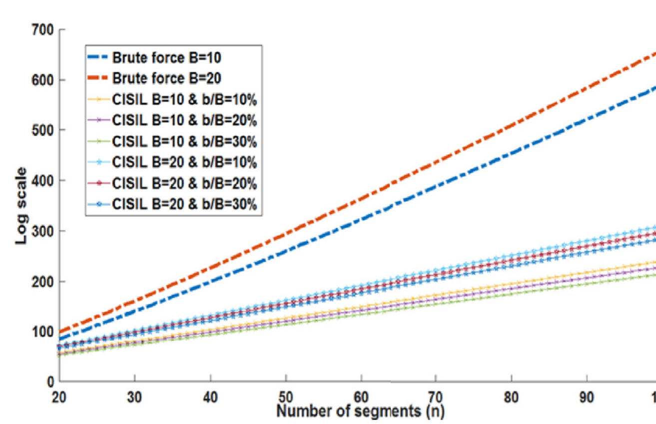

Fig. 4. Comparison between the runtime complexity of the tours formation step in CISIL and brute force while varying n, b and B.

Lemma 1. *Compared to centroid-based solutions, CISIL lowers the travel overhead on an MDC in serving a segment by at least a distance d, where d is the proximity of the segment centroid to its closest boundary node.*

Proof. Contemporary solutions assumes that the centroid "C" of a segment is the terminal at which data is uploaded to and downloaded from an MDC. Thus, the MDC needs to reach a point that is at most R units away from the centroid, where R is the communication range. In CISIL, a boundary node "B" serves as the interface for the segment, i.e., the designated terminal. The MDC has to only be R units away from "B". The travelled distance will be thus shortened by the distance between "B" and "C", i.e., d', as illustrated in Fig. 5. The least saving in the travel overhead corresponds to the shortest distance, $d < d'$, between "C" and the boundary sensors. ∎

Corollary 1. *In CISIL, an MDC that is touring between two segments S_1 and S_2 will incur travel overhead that is at least 4Δ less than that of a centroid-based solution, where $\Delta = \min(d_1, d_2)$, and d_1 and d_2 reflect the proximity of the centroid of S_1 and S_2 to the closest boundary node in S_1 and S_2, respectively.*

Proof. Based on Lemma 1, the length of the path between two segment boundaries is at least $d_1 + d_2$ shorter than that been the centroids of the two segments. Since an MDC that covers two segments will travel back and force between them, the tour (cycle) involves a round trip and will be thus $2(d_1+d_2)$ shorter than the corresponding tour between the two centroids. Thus, if $\Delta = \min(d_1, d_2)$, the MDC's trip is at least 4Δ shorter. ∎

Corollary 2. *Unless the centroids of three segments S_1, S_2 and S_3 are collinear, CISIL yields a tour that is at least 6Δ shorter than that of a centroid-based solution, where $\Delta = \min(d_1, d_2, d_3)$ and d_1, d_2 and d_3 are the shortest distances between of the centroid of S_1, S_2 and S_3 to a boundary node in S_1, S_2 and S_3, respectively. If the centroids of S_1, S_2 and S_3 are collinear, the tour length reduction is at least 4Δ.*

Proof. Based on Lemma 1, the length of the path between boundaries of two segments S_1 and S_2, is at least $d_1 + d_2$ shorter than that been the centroids, i.e., distance between C_1 and C_2. The same applies for S_2 and S_3, as well as S_3 and S_1. If C_1, C_2 and C_3 are not collinear (Fig. 6), the reduction in the tour will be at least $2(d_1 + d_2 + d_3)$, i.e., at least 6Δ shorter, where $\Delta = \min(d_1, d_2, d_3)$. On the other hand, if C_1, C_2 and C_3 are collinear, the CISIL-formed tour may cross the boundary of S_2 at two points that are at least $2d_2$ apart and thus the reduction in the tour length will be based on only the two end segments on the tour. In such a case, the travel distance reduction achieved by CIISL will be based on Corollary 1, i.e., at least 4Δ. ∎

Lemma 2. *CISIL reduces the MDC travel distance in a tour by at least $2\Delta(n-2q)$, where $\Delta = \min_{i=1,..n} d_i$, and d_i is the proximity of C_i, the centroid of*

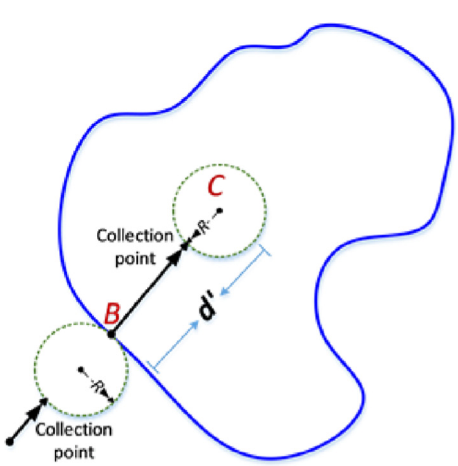

Fig. 5. Difference in MDC travel distance when data is collected at the segment boundary and its centroid.

segment S_i, to the closest node on the boundary of S_i, n is the number of segments covered by the tour, and q is the number of collinear inter-centroid links on the centroid-based tour.

Proof. We prove this Theorem by induction. Corollary 1 proves the case for $n = 2$ where q has to be zero. Based on Corollary 2, the theorem statement holds for $n = 3$, where $q = 0$ or 1 if the three centroids C_1, C_2 and C_3 form a convex polygon, or are collinear, respectively. Let's consider extending the shortest tour traversing these three centroids to cover a 4th segment. The additional centroid C_4 can be either: (i) collinear with C_2 and C_3; in that case C_4 replaces C_3 as the end point of the path and the least reduction of tour length compared to the centroid-based approach stays at 4Δ since three links are collinear (i.e., $q = 3$), or (ii) non-collinear with C_2 and C_3 in which case additional reduction of 2Δ may be achieved as proven in Lemma 1. Thus, the theorem hold for $n = 4$.

Assume that the theorem statement holds for $n = (m-1)$, the same logic can be applied to extend the tour of $(m-1)$ segments in order to support the mth segment. The least tour length reduction either stays the same when C_m is collinear with C_{m-1} and C_{m-2} (q gets incremented by one in this case) or additional gain of 2Δ is made (q stays the same). Hence, the theorem holds for $n = m$. ∎

Lemma 3. *Compared to the centroid based approach, CISIL achieves reduction of $2(k-1)\Delta$ in the total travel distance for interconnecting k MDCs.*

Proof. To route data to destinations, MDCs need to meet at rendezvous points to exchange some of their data payload. Such rendezvous requires certain MDC tours to overlap where a pair of tours shares a common segment. To ensure connectivity, CISIL determines the tour pairs that should overlap such that an *mst* is formed. Thus, compared to the case of one MDC, i.e., having one tour for all segments, overlapping the individual MDC tours fundamentally adds one inter-segment travel link (to visit the shared segment) per each additional MDC. For example, touring 5 segments by a single MDC requires traveling over 6 inter-segment links (to form a cycle) while using two MDCs that tour 2 and 3 segments require traversing 3 and 4 inter-segment links, respectively, i.e., a total of 7 inter-segment links. In CISIL, the rendezvous points are located at the segment boundaries, while the segment centroids are used in the centroid-based approaches. Based on Lemma 1, CISIL saves a distance of at least 2Δ by using the boundary node of a segment rather than its centroid. Therefore, CISIL reduces the total travel distance by at least 2Δ for every additional MDC after the first, by using rendezvous points on the segment boundaries rather than centroids. Thus, the total reduction will be at least $\sum_{j=2,..k} 2\Delta = 2(k-1)\Delta$ for all k MDCs. ∎

Theorem 4. *CISIL reduces the total MDC travel distance by at least $2\Delta(N+k-1-2Q)$, where $\Delta = \min_{i=1,..N} d_i$, and d_i is the proximity of C_i, the centroid of segment S_i, to the closest node on the boundary of S_i, k is the number of MDCs, N is the total number of segments to be federated, and Q is the total number of collinear inter-centroid links on the individual centroid-based tours.*

Proof. Based on Lemma 2, when using a MDC, CISIL achieves a reduction of at least $2\Delta(n-2q)$, where q is the number of collinear links within the inter-centroid tour. For k tours that inter-connect all segments, the reduction will be at least $\sum_{j=1,..k} 2\Delta(n_j - 2q_j)$, where $N = \sum_{j=1,..k} n_j$, and $Q = \sum_{j=1,..k} q_j$. Lemma 3, on the other hand, asserts the contribution of the MDC multiplicity, which is $2(k-1)\Delta$. Thus, the total reduction in travel distance for all MDCs will be at least:

$$
\begin{aligned}
&2(k-1)\Delta + \sum_{j=1,..k} 2\Delta(n_j - 2q_j) \\
&= 2(k-1)\Delta + 2N\Delta - 4Q\Delta \\
&= 2\Delta(N + k - 1 - 2Q)
\end{aligned}
$$

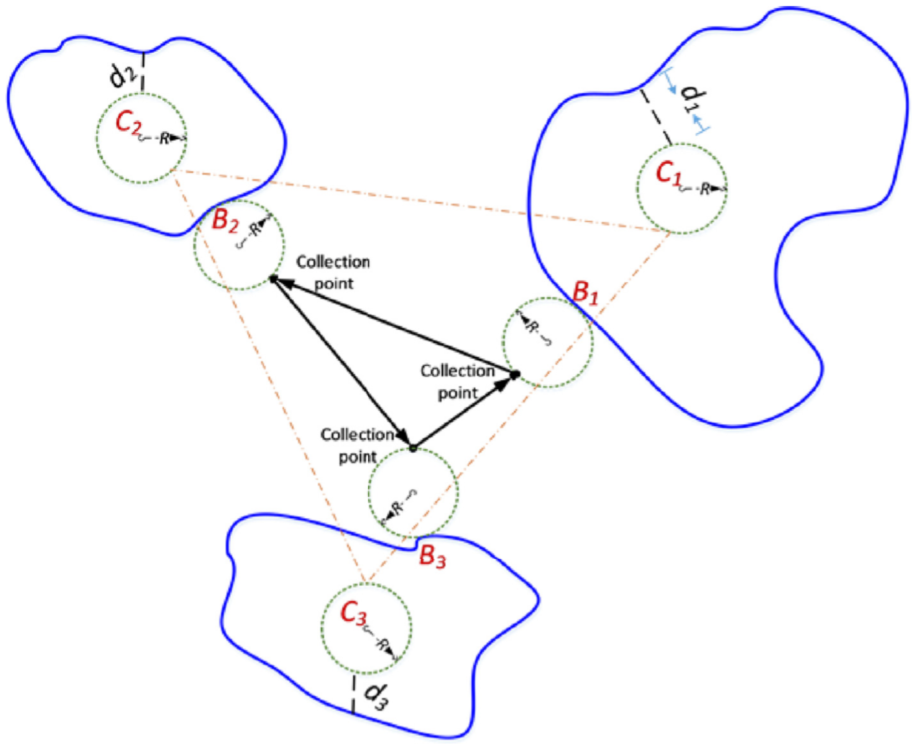

Fig. 6. Illustrating the difference in MDC tour between CISIL (inner dark triangle) and centroid-based approaches (dotted orange triangle). (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

6. Performance validation

6.1. Simulation environment and performance metrics

The performance of CISIL has been validated through extensive simulation experiments. The simulation environment is developed in Python. The geosteiner library (Juhl et al., 2014) is used for supporting hyper-graphs. We compare the performance of CISIL to that of MiMSI (Abbas and Younis, 2013), IDM-kMDC (Senel and Younis, 2012), FOCUS (Kalyanasundaram and Younis, 2013) and SS-WSN (Wu and Tseng, 2013). MiMSI models each segment as a terminal located at the centroid and forms a Steiner minimum tree to connect all terminals. MiMSI then partitions the terminals and the Steiner points using the k-means algorithm in order to form k clusters, each is to be served by an MDC. Gateways, either Steiner points or terminals, are then identified to serve as rendezvous stops for the MDCs. A tour is then formed and optimized for each cluster. Meanwhile, IDM-kMDC forms the *mst* of the terminals (segment centroids) and assigns one MDC for each *mst* edge. The algorithm merges the tours successively in order to meet the MDC count constraint.

On the other hand, FOCUS forms k-non–overlapping clusters and selects representative nodes (terminals) from each segment using CURE (Guha et al., 2001). Once CURE groups segments into "k" clusters, intra-cluster connectivity is supported by assigning one MDC to every cluster. To establish data routes between segments in different clusters, FOCUS uses the *mst* among the complete directed graph of clusters in order to extend some of them so that they overlap and form a connected inter-cluster topology. Unlike FOCUS, in SS-WSN the k-means algorithm is applied to group segments into clusters; the center of gravity of each segment is used as a terminal in the clustering process. Then, for each cluster, the travel path of an MDC is determined based on the convex hull of the involved segments, i.e., by factoring in the boundary of segments. As the objective of SS-WSN is to ensure efficient transmission of the collected data to sink, SS-WSN assumes that all tours are to include the sink. Therefore, to fit out system model and ensure inter-segment connectivity, we overlap clusters using the *mst* of the complete directed graph of clusters, similar to FOCUS. However, unlike FOCUS, we consider all nodes on the segment boundaries in forming such *mst* and consequently in determining the clusters that should be overlapped in order factor in the boundary nodes of segments in SS-WSN solution.

In the simulation, we assume that nodes are deployed within $900 \times 900\,\mathrm{m}^2$ area using a uniform random distribution. The transmission range is set to 50 m for both the nodes and the MDCs. The number of segments is varied from 5 to 12. To generate the partitioned topology, nodes are randomly deployed in the area and are then grouped into the required number of segments based on proximity. Some nodes are removed to prevent the overlap of the communication ranges of two nodes in separate groups (segments). Substitutes for the removed nodes are deterministically placed within the formed segments so that the total count matches the planned node population. The segment boundary is then redefined based on the concave hull of its nodes. In the simulation experiments, the number of nodes in the network is changed between 100 and 200.

The experiments capture the performance while varying the scope of the damage, the node density in the segment, and the number of available MDCs. For the former, the number of nodes is set to 200, the number of MDCs is fixed to 4 and the number of segments is varied between 5 and 12. For the node density, we measure the performance while fixing the number of MDCs to 4, and the number of segment to 6, while varying the number of survived nodes within a segment. The performance under varying number of MDCs is also studied in order to gauge the effect of increased resources on performance. This enables trading off the cost of segment federation and the achievable performance. Total and maximum travel distances are used for assessing the energy burden of the approach given that it is the dominant factor for the energy consumption. This represents the overall displacement of MDC to complete one tour each. The next section presents the performance results.

6.2. Performance results

Fig. 7 shows the effect of damage on the total tour length results for all approaches. As expected, the total travel distance increases when the

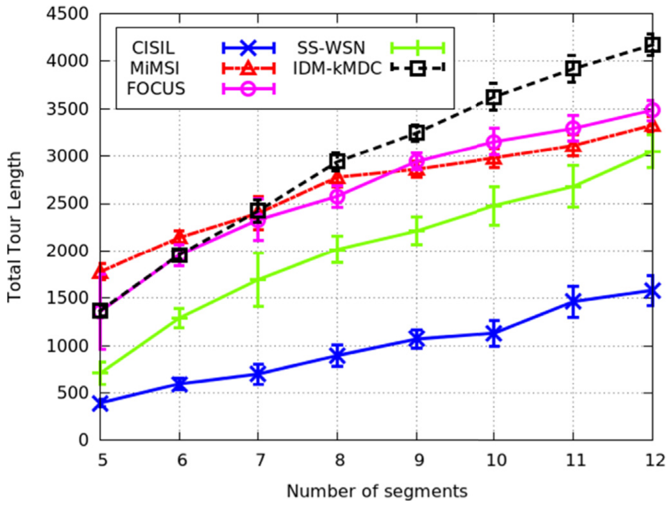

Fig. 7. The effect of the scope of the damage on the total tour length.

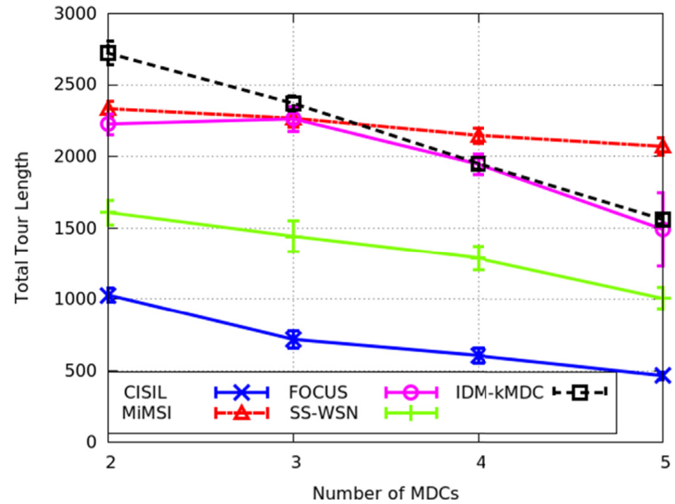

Fig. 9. Tour length as a function on the MDC count.

network is highly partitioned due to the increased recovery cost for such a case as the MDCs have more segments to visit. CISIL outperforms all baseline approaches under the various number of segments. In contrary to IDM-kMDC, FOCUS and MiMSI, both CISIL and SS-WSN consider all boundaries nodes as possible collection points and consequently yields shorter tours. On the other hand, CISIL considers all effective segment combinations by factoring in their neighboring relationship; such an optimization shortens the tour length of the individual MDCs for CISIL compared to SS-WSN. The advantage of CISIL over baseline approaches is sustained for highly partitioned networks with larger segment count.

The results in Fig. 8 shows that CISIL consistently yields better performance than the baseline approaches for various node densities. The increased number of survived nodes (within segments) have a positive impact on CISIL's performance. This is expected since the increased node count boosts the number of boundary sensors per segment and enables CISIL to optimize the formed tours. MiMSI, IDM-kMDC and FOCUS, on the other hand, are not affected positively because they consider representative nodes of a segment and do not factor in the segment shape. The gap in performance between CISIL and MiMSI IDM-kMDC and FOCUS in Fig. 8 speaks loudly for the need for considering the segment boundary in the solution and highlights the inefficiency of existing schemes that represent a segment as a terminal. Even though SS-WSN is affected positively when increasing the node count, the path planning scheme underperforms the tour formation procedure of CISIL, which factors in the relationship between segments to determine the collection points.

Fig. 9 shows the effect of the number of MDCs on performance. The plotted results in this figure provide insight that would enable effective resource planning and trade-off. Generally increasing the number of MDCs reduces the travel overhead for all approaches, which is very much expected since the individual tours will be more localized and shorter. When the number of MDCs grows, more clusters are formed, and consequently for the baseline approaches the number of terminals per cluster diminishes (as the number of segments is fixed). Since the baseline approaches pursue proximity-based grouping of terminals, the reduced cluster population shortens the tour length of the individual MDCs. Meanwhile, the performance of CISIL depends on the tour-based (hypergraph) mst. Increasing the number of MDCs enables them to serve closer segments and consequently shortens the overall travel distance. Moreover, more MDCs will pursue line tours which are less costly than polygon tours. In addition to the clear performance advantage over the baselines, overall increasing resources has significantly better impact on CISIL.

The maximum tour length results are reported in Figs. 10–12. Fig. 10 indicates that CISIL consistently outperforms IDM-kMDC, FOCUS and SS-WSN for increased segment count, mainly due to the optimized tour formation that CISIL applies. Unlike the total tour length (Fig. 7), Fig. 10 shows that the gap between CISIL and MiMSI closes as the segment count grows. This is not surprising as the presence of many segments narrows

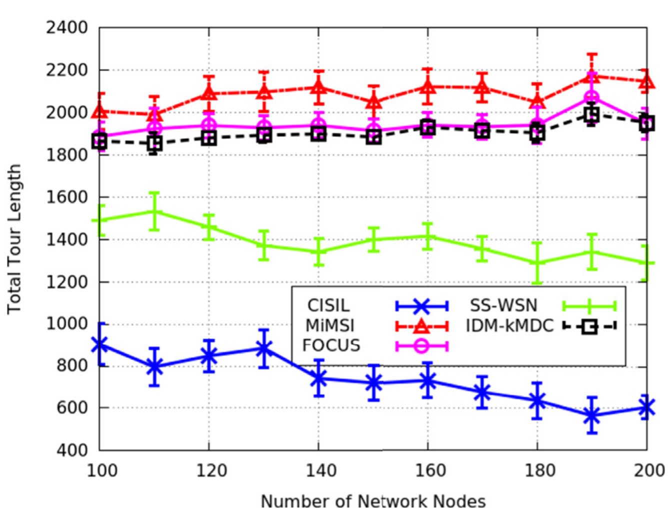

Fig. 8. The results of how the node density impacts the MDC travel overhead.

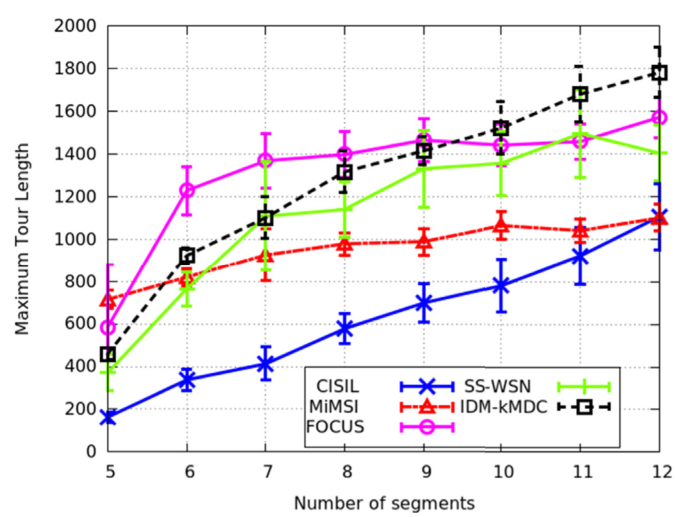

Fig. 10. How the maximum tour length is affected by the scope of the damage.

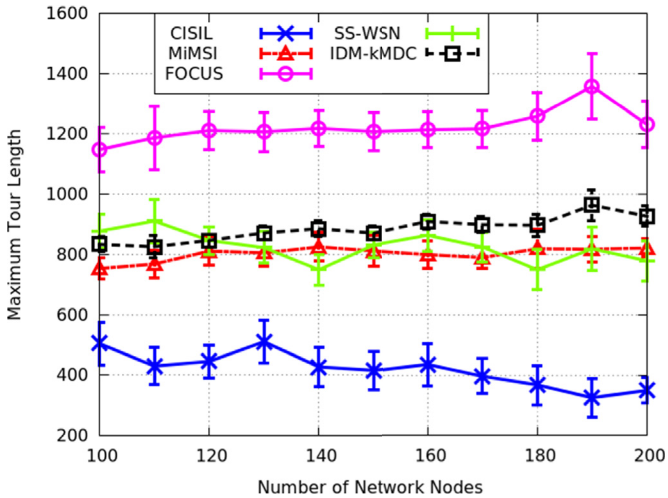

Fig. 11. The effect of node density the maximum distance travelled by an MDC.

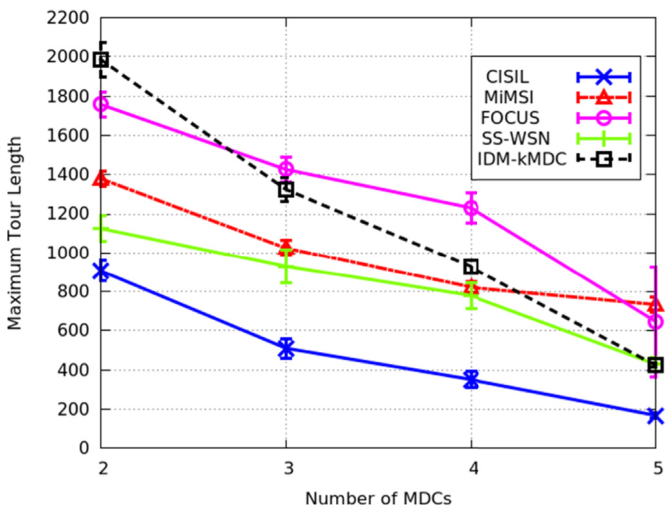

Fig. 12. How the MDC count impacts the maximum tour length.

the gap between them and enables the use of relays, i.e., Steiner points, as gateways instead of overlapping segments. Nonetheless, CISIL sustains its advantages for increased node density and MDC count, as shown in Figs. 11 and 12, respectively. The results in Fig. 11 are consistent with Fig. 8. For CISIL, the increased node population provides many options on the segment boundary for the MDC to determine the best interface node for a segment; the performance of MiMSI, IDM-kMDC and FOCUS, on the other hand, is not affected since they do not factor in the segment boundary in the tour formation. Compared to SS-WSN, CISIL better utilizes the boundary nodes in forming shorter tours. The results in Fig. 12 confirms the effectiveness of the tour formation of CISIL, which yields shorter MDC travel paths despite the proximity based clustering that baselines applies.

7. Conclusion

Sustaining strong connectivity is necessary in many WSN applications; therefore, repairing a partitioned network topology after the failure of multiple nodes is critical for the operation of the WSN. When insufficient resources are available to tolerate the failure, establishing intermittent links is deemed the only option, where multiple mobile data carriers (MDCs) pick data packets from sources and transport them to destinations. To simplify the problem, contemporary solutions represent a segment as a terminal and form tours by grouping the terminals to match the available MDC count. In this paper, we have argued that such a strategy is inappropriate because the shape and the size of the segments are not factored in. We have also proposed CISIL, a two step-heuristic to overcome such shortcoming. In the first step, CISIL determines the set of possible tours that can effectively interconnect various subsets of the segments using their boundary nodes. The second step opts to select the least cost subset of the formed tours with the cardinality that matches the available MDC count and ensures inter-segment connectivity. To do so, CISIL models the selection optimization as finding a k-edge *mst* for the hypergraph of tours. The simulation results have demonstrated the effectiveness of CISIL and its performance advantage over prominent competing schemes. In the future, we plan to extend CISIL by considering others objective functions, e.g., load balancing and inter-segment data delivery latency.

References

Abbas, A., Younis, M., 2013. Establishing connectivity among disjoint terminals using a mix of stationary and mobile relays computer communications. Comput. Commun. 36, 1411–1421.
Abuarqoub, A., et al., 2017. Dynamic clustering and management of mobile wireless sensor networks. Comput. Networks 117, 62–75.
Almasaeid, H., Kamal, A.E., October 2007. Data delivery in fragmented wireless sensor netowrks using mobile agents. In: Proc. The 10th ACM/IEEE Int'l Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), Chania, Greece.
Almasaeid, H., Kamal, A.E., Dec 2008. Modeling mobility-assisted data collection in wireless sensor networks. In: Proc. of the IEEE Global Telecomm. Conf. (GLOBECOM'08), New Orleans, LA.
Alsalih, W., Hassanein, H.S., Akl, S.G., 2010. Placement of multiple mobile data collectors in wireless sensor networks. J. Ad Hoc Networks 8 (4), 378–390.
Ang, K.L.M., Seng, J.K.P., Zungeru, A.M., 2017. Optimizing energy consumption for big data collection in large-scale wireless sensor networks with mobile collectors. IEEE Syst. J. https://doi.org/10.1109/JSYST.2016.2630691 to appear.
Bektas, T., 2006. The multiple traveling salesman problem: an overview of formulations and solution procedures. Omega 34 (3), 209–219.
Borradaile, G., Eppstein, D., 2015. Near-linear-time deterministic plane Steiner spanners for well-spaced point sets. Comput. Geom. 49, 8–16.
Cheng, X., Du, D.-z., Wang, L., Xu, B., 2008. Relay sensor placement in wireless sensor networks. Wireless Nets 14 (3), 347–355.
de Berg, M., Cheong, O., van Kreveld, M., Overmars, M., 2008. Computational Geometry Algorithms and Applications, third ed. Springer-Verlag Berlin Heidelberg.
El-Moukaddem, F., Torng, E., Xing, G., 2013. Mobile relay configuration in data-intensive wireless sensor networks. IEEE Trans. Mobile Comput. 12 (2), 261–273.
Golden, B., Bodin, L., Doyle, T., Stewart Jr., W., 1980. Approximate traveling salesman algorithms. Oper. Res. 28 (3), 694–711.
Guha, S., Rastogi, R., Shim, K., 2001. CURE: an efficient clustering algorithm for large databases. Infosystems 26 (1), 35–58.
Jain, S., Shah, R.C., Brunette, W., Borriello, G., Roy, S., 2006. Exploiting mobility for energy efficient data collection in wireless sensor networks. J. Mobile Networks Appl. 11, 327–339.
Jea, D., Somasundara, A.A., Srivastava, M.B., June 2005. Multiple controlled mobile elements (data mules) for data collection in sensor networks. In: Proc. 1st IEEE/ACM Int'l Conf. Distributed Computing in Sensor Systems (DCOSS'05), Marina del Rey, CA.
Joshi, Y., Younis, M., Dec 2012. Autonomous recovery from multi-node failure in wireless sensor network. In: Proc. of the IEEE Global Comm. Conference (GLOBECOM 2012), Anaheim, CA.
Juhl, D., Warme, D.M., Winter, P., Zachariasen, M., 2014. The GeoSteiner Software Package for Computing Steiner Trees in the Plane: an Updated Computational Study. Technical report. Dept. of computer Science, University of Copenhagen.
Kalyanasundaram, B., Younis, M., June 2013. Using mobile data collectors to federate clusters of disjoint sensor network segments. In: Proc. of the IEEE Int'l Conf. on Comm. (ICC 2013), Budapest, Hunger.
Lee, S., Younis, M., 2012. Optimized relay node placement for connecting disjoint wireless sensor networks. Comput. Networks 56 (12), 2788–2804.
Lloyd, E.L., Xue, G., 2007. Relay node placement in wireless sensor networks. IEEE Trans. Comput. 56 (1), 134–138.
Martello, S., 1983. An enumerative algorithm for finding Hamiltonian circuits in a directed graph. ACM Trans. Math Software 9 (1), 131–138.
Moazzez-Estanjini, R., Wang, J., Paschalidis, I.C., 2013. Scheduling mobile nodes for cooperative data transport in sensor networks. IEEE/ACM Trans. Netw. 21 (3), 974–989.
Senel, F., Younis, M., 2012. Optimized interconnection of disjoint wireless sensor network segments using K mobile data collectors. In: Proc. of IEEE Int'l Conf. on Comm. (ICC'12), Ottawa, Canada.
Senel, F., Younis, M., July 2016. Novel relay node placement algorithms for establishing connected topologies. J. Networks Comput. Appl. 70, 114–130.

Singh, G., Al-Turjman, F., 2016. A data delivery framework for cognitive information-centric sensor networks in smart outdoor monitoring. Comput. Commun. 74 (1), 38–51.

Stanislaus, L.V.M., Younis, M., September 2012. Delay-conscious federation of multiple wireless sensor network segments using mobile relays. In: Proc. of the 76th IEEE Vehicular Technology Conference (VTC2012-Fall), Québec City, Canada.

Warme, D., 1998. "Spanning Trees in Hypergraphs with Applications to Steiner Trees," *Ph.D. Thesis*. University of Virginia.

Wu, F.-J., Tseng, Y.-C., October 2013. Energy-conserving data gathering by mobile mules in a spatially separated wireless sensor network. Wireless Commun. Mobile Comput. 13 (15), 1369–1385.

Younis, M., Senturk, I., Akkaya, K., Lee, S., Senel, F., 2014. Topology management techniques for tolerating node failures in wireless sensor networks: a survey. Comput. Nets 58, 254–283.

Zhao, W., Ammar, M., Zegura, E., May 2004. A message ferrying approach for data delivery in sparse mobile Ad hoc networks. In: Proc. ACM MobiHoc, Tokoyo, Japan.

Wassila Lalouani received her BS and MS degrees from the Department of Computer Science at Computer Science at the university of science and technologies Houari Boumediene (USTHB), Algiers, Algeria, in 2006 and 2009, respectively. She is currently a lecturer at the High National School of Computing Science, Algiers, Algeria. Her research interest include fault tolerance and network management strategies for wireless sensor networks.

Dr. Mohamed Younis is currently an associate professor in the department of computer science and electrical engineering at the university of Maryland Baltimore County (UMBC). He received his Ph.D. degree in computer science from New Jersey Institute of Technology, USA. Before joining UMBC, he was with the Advanced Systems Technology Group, an Aerospace Electronic Systems R&D organization of Honeywell International Inc. While at Honeywell he led multiple projects for building integrated fault tolerant avionics and dependable computing infrastructure. He also participated in the development of the Redundancy Management System, which is a key component of the Vehicle and Mission Computer for NASA's X-33 space launch vehicle. Dr. Younis' technical interest includes network architectures and protocols, wireless sensor networks, embedded systems, fault tolerant computing, secure communication and distributed real-time systems. He has published over 200 technical papers in refereed conferences and journals. Dr. Younis has six granted and two pending patents. In addition, he serves/served on the editorial board of multiple journals and the organizing and technical program committees of numerous conferences. Dr. Younis is a senior member of the IEEE and the IEEE communications society.

Nadjib Badache was with CISTTT (Centre d'Information Scientifique et Technique et de Transfert Technologique) now CERIST (Centre de Recherche sur l'Information Scientifique et Technique) in Algiers, from 1978 to 1983, working on applied research projects in information retrieval, operating systems and library management. In 1983, he joined USTHB University of Algiers, as assistant professor and then professor, where he thought operating systems design, distributed systems and networking with research mainly focused on distributed algorithms and mobile systems. From 2000 to 2008, he was head of LSI (Laboratoire des Systèmes Informatiques of USTHB University), where he conducted many projects on routing protocols, energy efficiency and security in Mobile Ad Networks and Wireless Sensor Networks. Since March 2008 he is director of CERIST and professor at USTHB University.