



Gateway Placement in Hybrid MANET-Satellite Networks

Monia Hamdi, Laurent Franck, Xavier Lagrange

► To cite this version:

Monia Hamdi, Laurent Franck, Xavier Lagrange. Gateway Placement in Hybrid MANET-Satellite Networks. VTC 2012: IEEE Vehicular Technology Conference 2012, Sep 2012, Quebec City, Canada. <10.1109/VTC-Fall.2012.6399355>. <hal-00797569>

HAL Id: hal-00797569

<https://hal.science/hal-00797569v1>

Submitted on 7 Jun 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY 4.0 - Attribution - International License

Gateway Placement in Hybrid MANET-Satellite Networks

Monia Hamdi, Laurent Franck

Telecom Bretagne, MicroWave Department

Site of Toulouse, France

Email: {Monia.Hamdi, Laurent.Franck}@telecom-bretagne.eu

Xavier Lagrange

Telecom Bretagne, RSM Department

Site of Rennes, France

Email: Xavier.Lagrange@telecom-bretagne.eu

Abstract—Mobile ad hoc networks (MANETs) are self-organized and dynamically reconfigurable wireless networks. It is a promising technology for emergency deployment scenarios where communication systems should be highly reliable and able to operate in potentially adverse environments. However, MANETs experience severe impairments because of node motion or scarce node density. Satellite communications may help in this respect by setting up recovery links.

In this article, we study the problem of selecting MANET nodes that will provide access to the satellite capacity (i.e. gateway nodes). A key challenge is that nodes are mobile resulting in topology changes and network partitioning. We address this challenge by extending the use of the clustering techniques initially designed to solve scalability issues in MANETs. In our proposed architecture, the roles of clusterhead and gateway are merged. Our approach contrasts with prior work in that the gateway role assignment is performed in a distributed and mobile environment.

Through network dynamic analysis, we investigate the suitability of our novel approach for solving the gateway placement problem.

keywords: mobile ad hoc network, k-hop clustering, connectivity recovery, emergency communication, satellite, gateway placement

I. INTRODUCTION

The services offered by traditional wireless systems such as cellular networks depend on established infrastructures. In the aftermath of a disaster, communication infrastructures may be totally destroyed. There is a need for mobile technologies being independent of infrastructure where network management is the result of cooperation among terminals.

In MANETs, terminals self-organize yielding temporary topologies. MANETs display a great potential in emergency and rescue operations because of their instant deployment and reconfiguration capabilities. However, due to terminal mobility and dynamic topology changes, network partitioning may occur. The network is then split into unconnected groups. In such a situation, satellite communications may be of a great use to bridge these unconnected islands. Also, a minimal number of gateways should be deployed so as to save satellite and terrestrial resources. In the literature, this problem is known as the *Gateway Placement Problem*.

Similar issues exist in wireless mesh networks (WMNs), however the solutions proposed are not designed for a mobile environment. We propose the use of mechanism called *clustering* which was originally deployed for addressing scalability issues in MANETs.

The remainder of this paper is organized as follows. Existing gateway placement approaches are outlined in Section II. The behaviour of the clustering algorithm is evaluated and analyzed in Section III. Finally, Section IV concludes this paper.

II. RELATED WORK

A. Gateway Placement in WMN

The most common application of MANETs is expanding the coverage of wired and cellular networks. In that context, the architecture comprises a WMN (Wireless Mesh Network) through which communications take place between the Internet and the last hop network, the MANET (Fig. 1). In such architectures, the gateway placement arises in the Wireless Mesh Network.

[9] proposes a centralized gateway placement algorithm derived from the facility location problem and based on local search operations. FACE [10] makes use of the notion of centrality in graph theory to solve the gateway placement problem in a distributed manner. An algorithm derived from graph theory is also proposed in [3] where minimum Weighted Dominating Sets are recursively computed.

The abovementioned works assumed gateways to be fixed and gateway designation is performed during the deployment planning. In our application context, gateways are mobile. In such a challenging context, there is a need for new mechanisms able to cope with network dynamics.

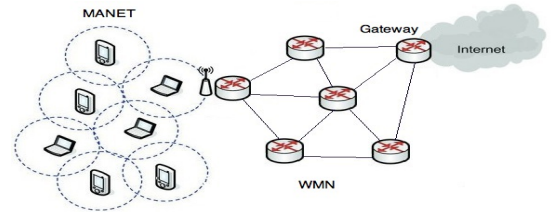


Fig. 1. Gateway placement in WMN

B. k - hop Clustering

Originally, the use of clustering techniques was proposed to solve scalability issues in large MANETs [7]. The network is divided into virtual groups of mobile nodes called clusters. A clusterhead is elected among mobile hosts to be the local

coordinator. In k -hop clustering, each node is k -hop distant from its clusterhead.

We start by describing clustering algorithms and then explain the choice of the most suitable clustering algorithm for our target deployment scenario. MaxMin [1] is the pioneering algorithm in this field. Cluster formation is composed of two phases (FloodMax and FloodMin) and each phase comprises k rounds of message exchanges. FloodMax aims to propagate the winner identifier and FloodMin avoids candidates with small identifier to be overtaken by candidates with larger identifier. A node should be aware not only that it has the largest identifier in its k -neighborhood but also that it has the largest identifier in any other node's k -neighborhood. MaxMin inspired several later works. The KCMBC algorithm [8], based on MaxMin introduces an expiration time metric to take into account the impact of node mobility.

Unlike MaxMin, k-lowestID and k-CONID [5] rely on clustering request and clustering decision flooding. Huang [6] improves these algorithms by introducing clusterhead back up mechanisms and a weight factor based on link quality. But like its ancestors, it results in broadcast storms.

In DSCAM [2], the cluster-based network structure forms a (r, k) -Dominating Set, where r is the minimum number of clusterheads per node and k is the maximum number of hops between a node and its clusterhead. The construction of an initial dominating set is based on node identifiers.

As already stated, the node properties impact the suitability of a node to serve as satellite access point. Therefore, the approaches where clusterhead selection is not based on inherent node characteristics are excluded. In addition to that, [8] proves that MinMax-based approaches outperform k-LowestID- based approaches with respect to cluster formation overhead. As a result of the previous analysis, the KCMBC algorithm is selected for this study.

The performance of the KCMBC algorithm has been already simulated and validated in a mobile environment [8]. However, prior work assumes dense networks where each node can communicate over multi-hop paths with any other node of the network. It is the first time a clustering algorithm is evaluated in a scarce-density network. The network may be splitted into isolated islands. These partitions are dynamic, nodes can leave a partition to join another one. Our aim is to analyze the KCMBC behavior in such a partitioned network.

C. KCMBC Overview

The KCMBC algorithm introduces an expiration time metric in order to keep the cluster structure as stable as possible and avoid frequent re-affiliations. This metric is computed for each link and represents the expected delay before the link disappears. As a result, nodes able to maintain longer connections with their neighbors are likely clusterheads.

The second step is the information gathering and dissemination. Each node propagates its decision for k rounds in FloodMax and k rounds for FloodMin.

Let us consider two neighboring hosts i and j . We denote the position vector of i by s_i and the velocity vector by

v_i . T_{ij} denotes the expiration time of the the link (i, j) . In cartesian coordinates, $s_i \equiv (s_{ix}, s_{iy})$ and $v_i \equiv (v_{ix}, v_{iy})$. The expiration time metric is expressed in seconds as:

$$T_{ij} = [\pm \sqrt{r^2(\Delta_{vx}^2 + \Delta_{vy}^2) - (\Delta_{sx}\Delta_{vy} - \Delta_{sy}\Delta_{vx})^2} - \Delta_{sx}\Delta_{vx} - \Delta_{sy}\Delta_{vy}] / (\Delta_{vx}^2 + \Delta_{vy}^2), \text{ where:}$$

$$\begin{cases} \Delta_{sx} = s_{ix} - s_{jx}, \Delta_{sy} = s_{iy} - s_{jy} \\ \Delta_{vx} = v_{ix} - v_{jx}, \Delta_{vy} = v_{iy} - v_{jy} \end{cases}$$

III. SIMULATION

A. Network Model

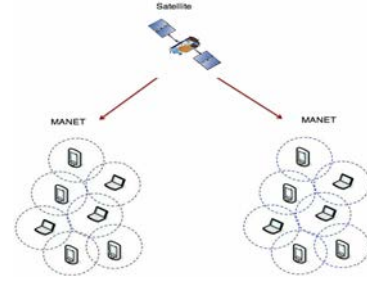


Fig. 2. Network architecture

The emergency deployment scenario relies on a specific mobility model, FireMobility, describing group motion behavior during forest fighting operations [4]. FireMobility complies with actual operation modes. This model yields a hierarchical network organization where fire-fighting forces are arranged into columns. Each column is divided into 4 groups, each group features 4 water tank trucks and 4 firemen pairs ($1 + 4 * (1 + 4 + 4) = 37$ nodes in each column).

B. Results and Analysis

The KCMBC algorithm comprises 3 main steps. The first step is node metric computation using the degree and the expiration time. The second step is the clusterhead selection through FloodMax and FloodMin and the third step is the cluster maintenance. During the first and second steps ($< 1s$), the network is supposed to be quasi-static. As we are interested on the impact of mobility on the gateway selection, this section focuses mainly on the third step, the cluster maintenance.

Cluster maintenance is required because of node mobility. It allows by means of information exchange to guarantee that the cluster structure integrity throughout topology changes.

Table I summarizes the simulation parameters. Each simulation is repeated 10 times and the results are averaged. The radio range is typical of WLAN technologies.

1) *Validation of the proposed architecture:* We need first to assess the necessity of using satellites for network connectivity recovery. The primary aim of this work is to provide and analyze a reliable communication system dedicated to emergency scenarios. Consequently, we examine the partitioning lifetime

Parameter	Value
Number of nodes	37
Playground size ($m \times m$)	1000×1000
Radio range (m)	[70, 100]
Simulation duration (s)	10000

TABLE I
SIMULATION PARAMETERS

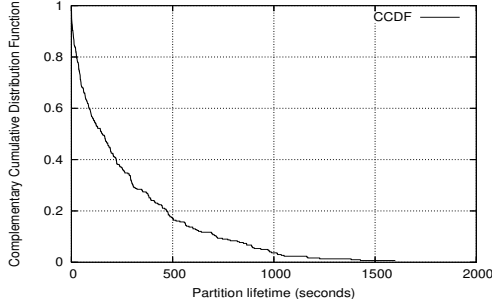


Fig. 3. Complementary Cumulative Distribution Function (CCDF) of the partition lifetime - Radio Range = 70 m

distribution with a communication radio range equal to 70 m, which is the worst case.

Figure 3 reveals that over 60% of partitioning events have a lifetime greater than 100 s and 19% of the network partitioning events have a lifetime greater than 500 s. FireMobility is a group mobility model where the nodes are often clustered and the update of the node positions takes place only in the case of safety distance transgression. If network connectivity is lost, partitioning may last for a long period, hence the high occurrence of long partitioning lifetime.

As pointed out by [10], tactical networks usually deploy a single gateway in each part of the network. Consequently, we assume one clusterhead per partition. In order to meet this requirement, the parameter k in KCMBC should be properly tuned. This parameter is bounded by the diameter of network partitions. Figure 4 shows the partition diameter distribution for different radio communication ranges in FireMobility. According to simulation results, k is set to the value of 12 so to ensure that one clusterhead per partition is sufficient to meet the emergency scenario deployment requirements.

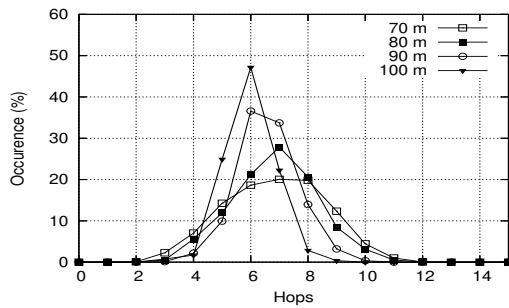


Fig. 4. Partition diameter distribution

To sum up, this initial analysis confirms the necessity of satellite for connectivity recovery. It also dimensions the parameters of KCMBC.

The following paragraphs address the impact of topology changes and evaluate the resulting KCMBC algorithm behaviour.

2) *Cluster Maintenance*: This part describes the topology changes in a partitioned network, evaluates the KCMBC behavior and proposes guidelines for cluster maintenance. Three cases represent the possible evolution of the network topology: splitting, merging and node migration. For each case, the topology evolution is illustrated, the KCMBC behavior is analyzed and simulation results with FireMobility are presented.

a) *Splitting and Merging*: Splitting occurs when two groups of nodes, initially located in the same partition, move away from each other and form two different partitions (Fig. 5). The nodes, located in a partition where there is no clusterhead, trigger reclustering. In this context, reclustering consists in a new clusterhead election for this subset of so-called orphaned nodes. In KCMBC, if an orphan node detects more than d orphan neighbors, those orphans attempt to trigger a new cluster formation. This rule guarantees that there is at least one clusterhead per partition. However, KCMBC does not detail how to detect the loss of a clusterhead.

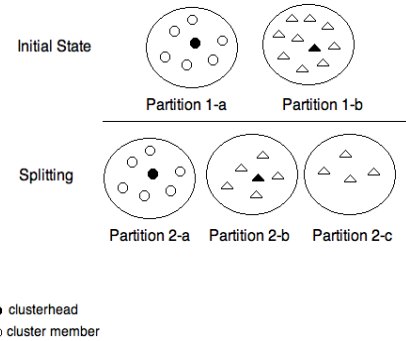


Fig. 5. Partition splitting

Partitions may also move toward each other to form a single partition (merging). After merging, full network connectivity may be recovered. In this case, the use of satellite, hence clustering is no more required. The network may also remain partitioned (Fig. 6). In KCMBC, two clusterheads upon partition merge, keep their status, unless they become neighbors. A functional requirement in this work is to have only one clusterhead per partition. One of the clusterhead should therefore resign. According to the value of k chosen above, the diameter of the partition resulting from the merging of two partitions is lower than k . If one of the two clusterheads loses its clusterhead status, previous cluster members are still within k hops from the other clusterhead. In order to manage partition merging, a clusterhead should therefore be able to detect the presence of other clusterheads in its partition.

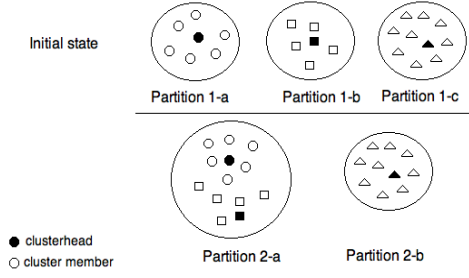


Fig. 6. Partition merging: partitioned network

Before describing the third case (node migration), the next paragraphs show how often splitting and merging take place in the FireMobility scenario. A partitioning event is defined as the splitting of a previously connected topology into several partitions. Splitting and merging occurrence is defined as the amount of partition splitting and merging respectively during a partitioning event. Figure 7 shows that splitting and merging occurrences are approximately similar. During a partitioning event there is a continuous oscillation between splitting and merging. It can also be noticed that the lower the radio range, the more dynamic the topology. For a radio range lower than 80 m, partition merging and splitting occur at least once every partitioning event. As a result, the cluster maintenance guidelines given above in the case of a general network also apply to FireMobility: each clusterhead should detect the presence of other clusterheads in its partition, each cluster member should detect the loss of its clusterhead and orphan nodes should trigger reclustering.

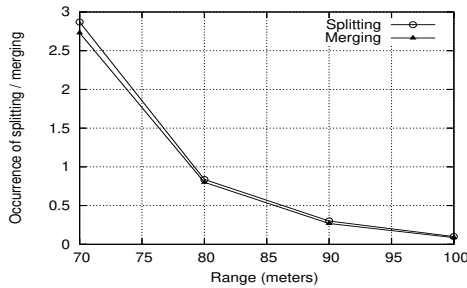


Fig. 7. Average number of partition splitting and merging during a partitioning event

The previous paragraphs show that because of partition merging, several clusterheads may be located in the same partition. As already stated, KCMBC does not implement the detection of the presence of multiple clusterheads in the same partition. We aim at analyzing the effect of KCMBC on the clusterhead diversity which is defined as the ratio between the duration several clusterheads are observed in the same partition and the total partitioning lifetime. The total partitioning lifetime is considered instead of the simulation duration because the maintenance procedure is only relevant when the network is partitioned. If the cluster maintenance

procedure described in KCMBC is applied, Figure 8 shows that during a partitioning event, several clusterheads are observed in the same partition for at least 20% of the time. As a result, the current maintenance procedure in KCMBC does not totally meet the needs of a partitioned network.

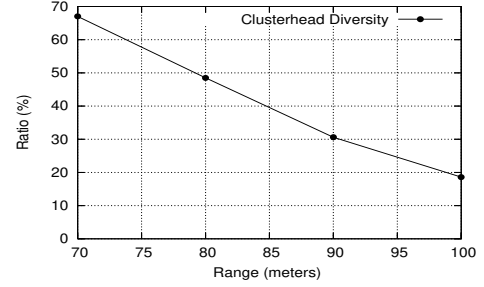


Fig. 8. Ratio between duration of clusterhead diversity over total partitioning duration

b) Node Migration: Because of the mobility, a node may move to a different partition. The node may be a cluster member (Fig. 9(a)) that joins an already formed cluster. In KCMBC, if a node loses the path to its former clusterhead and detects the creation of a new radio link, it joins the cluster to which its new neighbor belongs. This assumes that nodes communicate the identifier of their clusterhead to their neighbors.

The node may also be a clusterhead (Fig. 9(b)). A node u is affiliated to v , if the node v is the clusterhead of u . If the clusterhead has no more affiliated neighbors, it joins the other cluster. Otherwise, it keeps its clusterhead status. The latter case is similar to merging, resulting in a partitioned network. A clusterhead should therefore be able to detect the presence of other clusterheads in its partition.

The clusterhead migration is similar to partition merging in regard to the cluster maintenance. Next, the cluster member migration is considered. It is defined as the number of cluster members leaving their partitions during a partitioning event. Figure 10 shows that the nodes do not stay always in the same partition. Consequently, in the cluster maintenance procedure, nodes leaving their partition should be able to detect the neighboring clusters in their new partitions.

In KCMBC, each node includes the identifier of its clusterhead in the "Hello" messages. However, to be able to make relevant decision, the information sent by a node to its neighbors should also be updated. For instance, if a node leaves its cluster and joins another cluster, the information included in its "Hello" messages is no more relevant. Moreover, if a node loses its clusterhead status, its affiliated members have to be informed. The KCMBC authors suppose that each node has the required information to make relevant decisions without specifying the underlying signaling messages.

As a conclusion, the maintenance procedure in KCMBC does not meet all the needs of the FireMobility model. For partition splitting or clusterhead migration, KCMBC does not detail how a node detects the loss of its clusterhead.

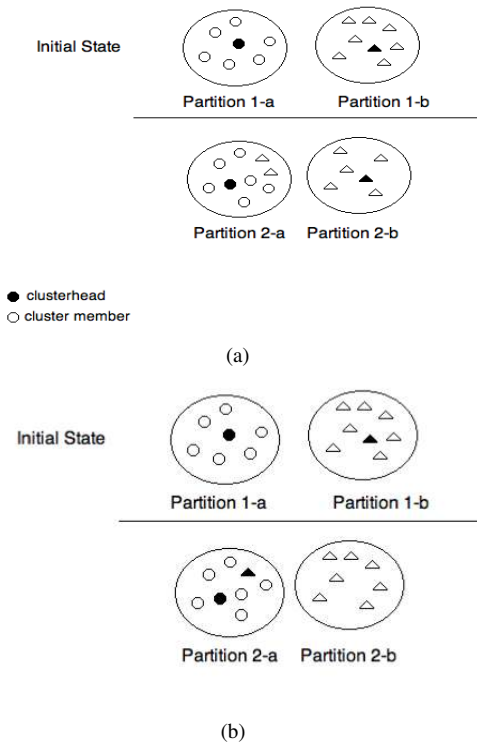


Fig. 9. Node migration: cluster member migration (a), clusterhead migration (b)

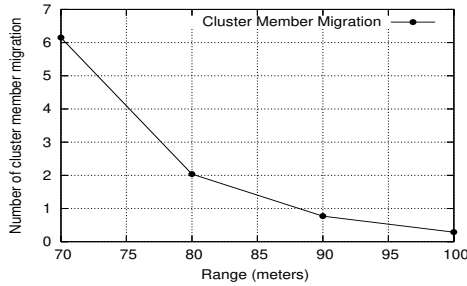


Fig. 10. Average number of cluster member migrations during a partitioning event

Furthermore, in case of partition merging, clusterheads keep their status, which transgresses the functional requirement of one clusterhead per partition. This study allows to highlight additional cluster maintenance requirements: each clusterhead should be able to detect the presence of other clusterheads in the partition, each cluster member to detect the loss of its clusterhead and the presence of neighboring clusters and orphan nodes to trigger reclustering. The status transition may be reaffiliation for cluster members or resignation for clusterheads. The description of the rules determining the node status transition is determinant for the cluster maintenance design, but it is not sufficient. The description of the messages exchanged by the nodes is also compulsory to have a maintenance procedure able to manage the dynamic topology change in a partitioned network.

IV. CONCLUSION

In this contribution, we address the issue of coping with network partitions in MANETs during emergency situations. The scenario selected is based on forest firefighting operations. Our initial analysis shows that in such a situation, 50% of the partitions last for more than 146 s, calling for the set up of satellite links to bridge unconnected network partitions. However, the nodes hosting these satellite links must be selected so to minimize the economical cost and optimise the network operation.

We propose to use a clustering technique called KCMCB and assess that it is fit for identifying the nodes serving as satellite gateways. However, we also show that in a context where partitioning is highly dynamic, KCMCB does not completely fulfill the requirements of cluster maintenance. The contribution ends with proposals in order to extend KCMCB and meet the requirements of the application.

REFERENCES

- [1] A. D. Amis, R. Prakash, T. H.P. Vuong, and D. T. Huynh. Max-min d-cluster formation in wireless ad hoc networks. *Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 2000.
- [2] V. S. Anitha and M. P. Sebastian. Scenario-based diameter-bounded algorithm for cluster creation and management in mobile ad hoc networks. *13th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications*, 2009.
- [3] B. Aoun, R. Boutaba, Youssef Iraqi, and G. Kenward. Gateway placement optimization in wireless mesh networks with qos constraints. *Selected Areas in Communications, IEEE Journal on*, 24(11):2127 – 2136, nov. 2006.
- [4] L. Franck, M. Hamdi, and C. G. Rodriguez. Topology modelling of emergency communication networks: Caveats and pitfalls. *TIEMS (The International Emergency Management Society) Workshop*, 2011.
- [5] F. G. Nocetti G. Chen, J. S. Gonzalez, and I. Stojmenovic. Connectivity based k-hop clustering in wireless networks. *Proceedings of the 35th Hawaii International Conference on System Sciences*, 2002.
- [6] T. C. Huang, L. C. Shiu, and H.C. Ke. A double-manager k-hop clustering algorithm in mobile ad hoc networks. *Fourth International Conference on Computer and Information Technology*, 2004.
- [7] P. Krishna, M. Chatterjee, N. H. Vaidya, and D. K. Pradhan. A cluster-based approach for routing in ad hoc networks. *Proceedings of the 2nd Symposium on Mobile and Location-Independent Computing*, 1995.
- [8] S. Leng, Y. Zhang, H. H. Chen, L. Zhang, and Ke. Liu. A novel k-hop compound metric based clustering scheme for ad hoc wireless networks. *IEEE Transactions on Wireless Communications*, 8(1), 2009.
- [9] J. Robinson, M. Uysal, R. Swaminathan, and E. Knightly. Adding capacity points to a wireless mesh network using local search. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 1247 –1255, april 2008.
- [10] Bo Xing, M. Deshpande, S. Mehrotra, and N. Venkatasubramanian. Gateway designation for timely communications in instant mesh networks. In *Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2010 8th IEEE International Conference on, pages 564 –569, 29 2010-april 2 2010.