Dynamic Topology Implementation and Maintenance for the CHORIST Network

Daniel Câmara, Christian Bonnet, Fethi Filali
Mobile Communications Department
EURECOM
Sophia Antipolis, France
{daniel.camara, christian.bonnet, fethi.filali}@eurecom.fr

Abstract— **This paper presents the implementation and evaluation of a distributed topology management algorithm for public safety networks (PSNs) to implement the CHORIST architecture[1]. Topology management for this kind of network is a mission critical problem, for a mission critical network. PSNs are the networks installed by the authorities to coordinate relief/rescue efforts in case of disasters. The main concerns for PSNs are rapid deployment and survivability. The whole operation and teams coordination schema depends on how well deployed the network was and how stable and reliable the network is during its lifetime. The more stable the network structure is in a rescue operation field the better. The proposed algorithm is able to dynamically adapt to the nodes mobility thus maintaining the desired topology**

***Keywords**- cluster heads; mesh networks; connection cost; public safety*

I. INTRODUCTION

The deployment and management of nodes for wireless Public Safety Networks (PSNs) is a fundamental and challenging problem. A well defined and maintained network structure is an indispensable step to enable the creation of efficient higher layer algorithms [2]. For this reason topology control becomes a basic functionality to enhance scalability and capacity for large-scale networks [1]. Differently of other networks the main concerns for public safety networks are rapid deployment and survivability [2].

The main contribution of this work is the proposal of a stable and efficient solution to implement and manage the structure designed by the CHORIST project [13], taking into account the constraints imposed by the communication model. CHORIST is an European Commission project that addresses environmental risk management focusing natural hazards and industrial accidents [13]. The backbone topology, depicted in Figure 1, is composed of Cluster Heads (CHs), Mesh Routers (MRs) and Relay Nodes (RNs). All the nodes' roles must be defined dynamically, based only on local information and following the channel model defined by the consortium [14].

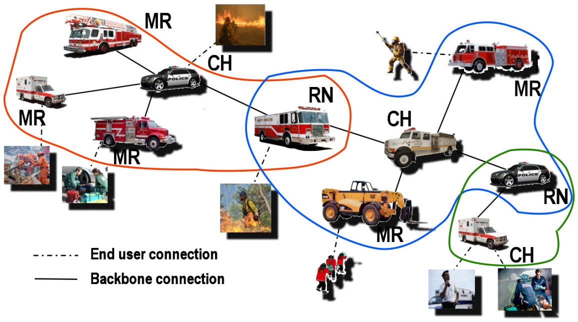

Figure 1. CHORIST network description and main components

The CHORIST structure was designed to be efficient and decrease interference among nodes. Hierarchical structures are normally scalable and decrease the overall need for controlling messages among the nodes [15]. However, creating and maintaining the structure has a cost in terms of bandwidth and delay. Understanding the mechanics of such costs, and the tradeoffs involved, is a fundamental step to enable the creation of efficient and useful networks. Our proposal builds and efficiently coordinates the proposed CHORIST two-level hierarchical topology.

The remainder of this paper is organized as follows. In Section II, we discuss related work. In Section III, we introduce the CHORIST architecture and discuss some of its characteristics. In Section IV, we present the experiments and analyze their results. Section V draws conclusions and points to the next steps for this work.

II. RELATED WORK

Midkiff and Bostian [3] present a two-layer network deployment method for public safety networks. Their network consists of a hub and, possibly, many purpose specific routers to provide access to the nodes in the field. In some sense our work provides the same kind of topology, since we are interested in the backbone creation to provide access for the end nodes, e.g., firefighters in the field, however, our main constraints, nodes roles and organization are different.

Bao and Lee introduce in [2] a rapid deployment method to create a wireless ad hoc backbone for public safety networks. Our work has some similarities with theirs since we also use nodes' connections and link quality to decide their role. However, we dynamically define the roles using the current nodes position and states. We do not request nodes to move to enable connections. In some cases, this may lead to

---

[1] CHORIST is an European project that focuses on the deployment of PSNs (http://www.chorist.eu/)

non optimal configurations. However, it is not realistic to ask, for example, a firefighter, while in a rescue operation, to move the truck to improve the network connectivity.

Sarrafi et al. present in [4] an interesting algorithm for topology control, however, its main objective is to decrease reach the power consumption optimality of the network. Our objective here is distinct, we want to maintain a specific topology to enable robust and efficient communication.

Aschenbruck et al. [5] introduce a realistic model for node distribution over disaster area scenarios and evaluate the impact of different planar topology control strategies over the network connectivity. The proposed model divides the target area into different purpose specific sub-areas, e.g. incident location, patients waiting for treatment, hospital. Aschenbruck et al. model is elegant and based on a real maneuver simulation; however, it hardly covers all the possible mobility and distribution scenarios for PSNs. The work does not proposes a new control strategy but evaluate three planar ones. No hierarchical strategy is evaluated, even though hierarchical networks are more scalable than planar ones. We used the proposed model to evaluate the CHORIST architecture against the proposed scenario.

III. THE CHORIST ARCHITECTURE

The core of the CHORIST network is a two-level hierarchical structure. A firefighter, for example, could use any node as an access point, however, inside the proposed structure each node has its specific role. Cluster Heads (CHs) are the nodes responsible for managing the radio resources for their clusters. Relay Nodes (RNs) are the nodes that are part of two, or more, clusters and act as a bridge among them. Mesh Routers (MRs) are the nodes attached to CHs, MRs obey the CHs scheduling in order to communicate with other nodes. Nodes not yet attached to the network, or that for some reason lost their roles, are called Isolated Nodes (IN). If required, an IN may become a CH or a MR. The organization of these elements follows a well defined and strict organization. Neither two CHs nor two RNs can be directly connected. For example, if a CH needs to exchange control data with another CH, the messages must be forwarded through a RN.

The CHORIST backbone follows the channel model defined by the OpenAirInterface [14]. The main architecture of the CHORIST, is derived from the adopted channel model and frequency reuse pattern. From the topology management point of view the two main constraints of the channel model are: no CH should be in the range of another CH and broadcast channels are reserved to CHs. No other node should broadcast messages. Two neighbors MRs may communicate directly, if previously agreed, but the communication must be direct, not through a broadcast channel. A MR, when inside a CH area, should be attached to it.

Our protocol assumes a reactive approach; nodes perceive changes in vicinity through periodic connections update messages sent by the CHs. As a result, the delay to react to changes is linked to the frequency of the update messages. If a mobile node evades from the CH range it takes a few seconds for the node to realize that now it may be in an area uncovered by any other CH and, thus, it is his duty to become a CH. Algorithm 1 presents the protocol in further details.

```
1. Node Arrives (IN)
2. Waits for Connection Updates
3. If received any Update
4.    Sends a Connection Update to the CHs
5.    Becomes a MR
6. Else if number of trials less than 3
7.    Return to 2
8. Else
9.    Becomes a CH
10.   Sends a connection Update
11. End if
12. Wait for messages
13. If receive a Connection Request (only if it is a MR or a RN)
14.    Responds with a Connection Response informing all its
         neighbors
15. Else if received a Connection Confirmation (Only if it is a CH)
16.    Registers the connection
17. Else if receive a Connection Response
18.    Sends a Connection Confirmation
19.    Registers Connection
20. Else if receive a Connection Update
21.    Registers the Update
22.    Registers the Neighbor
23.    If actual state == CH and sender == CH
24.       Other CH on the range
25.       Decides, based on the ranks, his and the sender one, if
            gives up being a CH or not
26.       Sends an Update Message
27.       Waits a Random time
28.    End if
29.    From time to time Evaluate Updates to find not Connected
         CHs
30. Else if receive a Connection Cancel
31.    Removes the connection
32.    Reevaluate actual state (may become a MR)
33. End if
34. Return 12
35. If connection time out occurred
36.    Remove neighbor
37.    Reevaluate state (may became a IN or a MR)
38.    If become a IN
39.       Return 2
40.    End if
41. End if
42. From time to time sends a Connection Update for the
      connected nodes
```

Algorithm 1. CHORIST Network topology control algorithm

If we consider the network as a graph, taking nodes as vertices and connections as edges we can reduce the CHORIST architecture to a two steps Weakly Connected Independent Dominating Set (WCIDS) [8]. For a given graph $G = (V, E)$ and a subset S of the set of vertices $V(G)$, S is called a dominating set if, for any vertex $v \in G$, v is either inside S or it is adjacent to a vertex in S. In our case S can represent both, the CH and MR sets. A set S is called connected if S is a dominating set and the sub graph induced by *S* is connected. The minimum independent set is the one with the lowest possible cardinality. In graph theory a set of vertices is called independent if no two elements in it are adjacent, i.e. there is no edge that connects any pair of vertices of the set. In our problem we have exactly this configuration, the CH set must be a dominating set, since all MR and RN should be connected to a CH. More over, two CHs should not be in the range one from each other. It is important to notice also that the RN set also needs to be a dominating set, regarding the formed CH set. I.e. if we consider the CH set as *S*, than *V(G)* would be the whole network. If we consider the MR set as *S*, than *V(G)* would be the selected CH set. This makes the problem even more interesting. The minimum dominating set is desirable since we want to decrease, as much as possible, the number of links and signaling messages exchanged among CH nodes.

Reducing the CHORIST network structure the solution of the WCIDS problem, helps the understanding of the topology but does not solve the problem. Unfortunately, both the dominating set and the connected dominating set problems are NP-Complete [9][10]. One of the most well known heuristics for solving the connected dominating set problem is the centralized approach proposed by Guha and Khuller [11]. Although there are distributed implementations of this heuristic [7], our topology is not exactly the same and the distributed approach can not be used directly in this case. We must also consider that our topology is dynamic, nodes may attach and detach from the network at any time.

CHORIST hierarchy provides scalability to the network structure. Hierarchical networks present better performance and are more robust. These enable the achievement of higher data throughputs [15]. Another important characteristic of hierarchical networks is the decrease in the number of required links among nodes. This can be perceived in Figure 2 which shows comparatively the connectivity of the four evaluated methods. The number of links varies considerably among the approaches. The first diagram, WCIDS, shows the result of the application of the Weakly Connected Independent Dominating Set over the network connection graph. The second diagram shows the application of our technique over the same scenario. We can see that even though the clusters are on different positions the number of CHs, represented by bigger squares, is the same. More over, the number of generated edges is also nearly the same, even though for our approach they are generated dynamically and only with local information. The next two diagrams show the same nodes distribution connected through planar techniques. The number of created links, for both, is considerably bigger. In the k nearest neighbor technique, each node connects to at least k other neighbor nodes. For the fixed range technique, if two nodes are on the communication range they are connected.

An important characteristic we want to emphasize about the problem is that both, RN and CH sets, should be WCIDS and as minimum as possible. However, both sets are not independent. The RNs selected to compose the RN WCIDS must be selected among the CH WCIDS nodes neighbors.

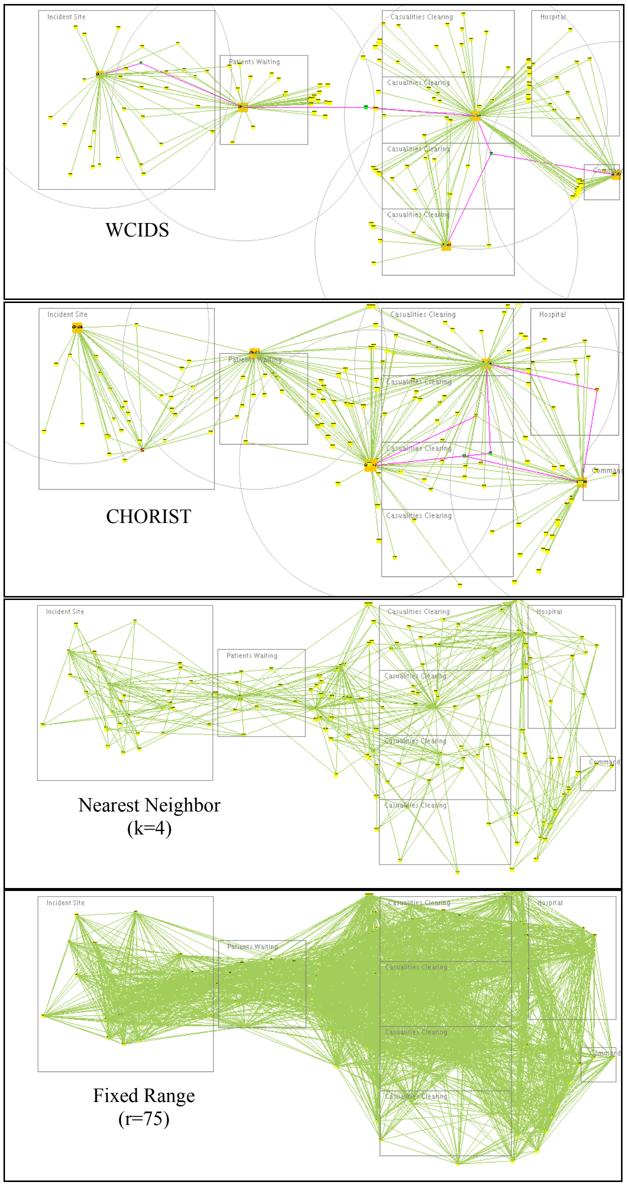

Figure 2. The connectivity of the different topology controll strategies for the evaluated scenario (300x200m area, 150 nodes, 75m communication range)

IV. EVALUATIONS

The evaluations were made using Sinalgo simulator [12] in a 2000x2000 meters area for the WCIDS. When using the Aschenbrucket al. distribution model we used the same area described in [5], 300x200 meters area. We vary the number of nodes and the communication range of the nodes. All experiments were conducted using Linux Fedora Core release 6 on an Intel Xeon 1.86GHz machine with 16GB of RAM. All graphs are presented with a confidence interval of 99% and each point is the result of the mean of 34 runs of 3 hours simulation time with different network configurations and 1% of message loss. For the comparisons with the WCIDS

Topology Control Strategy	Pedestrian Avg. node degree	Pedestrian + vehicular Avg. node degree	Pedestrian % of nodes disconnected	Pedestrian % of nodes disconnected	Pedestrian Avg. path size	Pedestrian +vehicular Avg. path size	Pedestrian max path size	Pedestrian +vehicular max path size
CHORIST	2.90	3.38	10.22	19.25	2.23	2.13	4.18	3.97
K nearest neighbor (k=3)	3.25	2.29	6.78	48.03	5.35	4.4	10.51	8.85
K nearest neighbor (k=4)	4.27	3.01	2.03	37.62	4.16	3.86	7.91	7.56
K nearest neighbor (k=6)	6.09	4.66	0.41	20.77	3.23	3.36	6.13	6.43
K nearest neighbor (k=8)	7.97	6.49	0.22	8.64	3.08	1.12	5.33	5.89
K nearest neighbor (k=10)	9.74	8.21	0.21	3.84	2.89	1.13	2.88	2.94
Fixed range 100m	75.14	82.14	0.20	0.22	1.12	1.13	2.88	2.94

Table 1. Disaster area scenario summarized results considering pedestrian and pedestrian plus vehicular scenarios

algorithm scenarios, nodes arrive randomly and are placed uniformly over the observed area. The centralized WCIDS implementation works directly over the connection graph, it is an oracle that knows the position of all nodes and creates the minimum arrangement in an offline manner. The final result is the best possible one and it is hardly achievable with distributed algorithms, where nodes have only local information and new nodes arrive at different moments throughout the network lifetime. However, it represents a base of comparison to evaluate how far our implementation is from the theoretical minimal CH/RN optimal solution.

To evaluate the CHORIST network stability and availability we use the same distribution and mobility scenario proposed by Aschenbruck et al. [5]. However, instead of just evaluating the connectivity we implemented the protocols and compared our results with the other algorithms simulated behavior on the same conditions. We use the same area size, nodes distribution and organization described in [5][6]. However, we simulated the network in two distinct situations, the first one when all nodes have pedestrian speed, (0.5 m/s on average and variance of 1 m/s) and another scenario where we have a mix of pedestrian and vehicular nodes. For the second scenario nodes inside the defined zones are pedestrian and nodes that travel from one zone to the other have vehicular speed (average of 40 Km/h and variance of 4 Km/h). When exposed to higher mobility rates, transmission failures, delays, and lack of information the performance of the planar algorithms were slightly worse than the one observed in [5]. Table 1 summarizes the obtained results. We can observe that the degrees of the nodes for the CHORIST architecture are the lowest ones, for both pedestrian and vehicular speed experiments. The percentage of nodes disconnected from the point of view of each node, i.e. how many percent of the other nodes are unreachable at each time. For example, for an isolated node this value would be 100%, for the others, if all connected, would be 0.67%. Two nodes connected only if the protocol recognizes them to be attached, and if they are indeed inside the communication range.

For all the evaluated protocols the addition of the vehicular speed nodes presented a considerable impact. Every communication protocol needs a time to adapt to topology changes. As nodes are mobile, the view nodes have of the topology, connectivity and other nodes positioning information, may be outdated. Sometimes a node recognizes other nodes, which moved, as connected and at the same time may fail to recognize nodes in the range as reachable. The CHORIST structure is a more sophisticated one, and it takes slightly more time for the nodes to get organized (e.g. recognize new clusters, attach to them). For this reason more nodes fail to recognize connections, when compared with the k-neighborhood algorithms. However with the increasing in mobility, the k neighborhood needs considerably more recourses, i.e. number of links, to reach the same results presented by the CHORIST structure.

From Table 1 we also have, measured in number of hops, the average path size, and the average longest shortest path for each node $(\forall\ i,j \in V(G)\text{: } ls=\max_{ij}d(i,j))$. Again, CHORIST paths were smaller than the comparable k-neighborhood ones. The k-neighborhood algorithm needs k=8 or k=10 to present the same path sizes CHORIST does. However, this also means spend more resources to generate and maintain the structure.

Figure 3 presents a comparison between CHORIST and the offline WCIDS implementation over different density scenarios. The number of CHs created for both is close, normally with an overlap on the 99% confidence interval. Even though, for our approach we do not have a complete view of the network and nodes arrive randomly during the network uptime. We can also perceive that the number of clusters increase sub linearly, considering the number of nodes in the network. We find out with this that for the CHORIST network the number of clusters more related to the covered area than the number of network nodes. On the other hand, the number of nodes per cluster increases almost linearly with the number of network nodes. Nevertheless the number of cluster nodes for both approaches, CHORIST and WCIDS keeps basically the same for all evaluated scenario. The number of RNs generated by our implementation of CHORIST has, on average 6.75% more RNs than the WCIDS implementation.

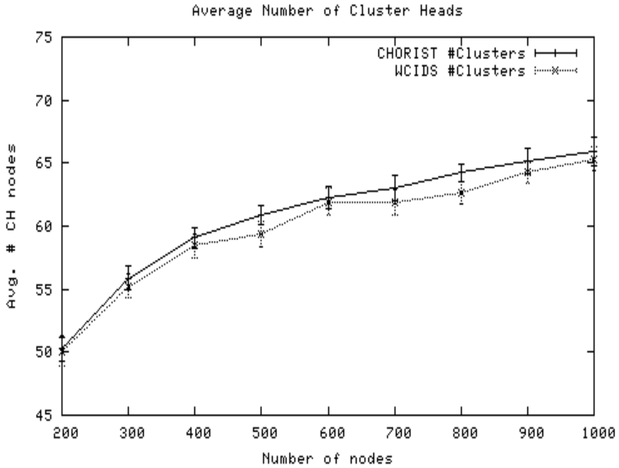

Figure 3. Average number of clusters on the network varing the number of nodes on the network and using 200m communication range

This occurs mainly because. For our approach, CHs chose their RNs in a selfish way. A CH picks the most interesting nodes, for their point of view to become its RNs. Although this, by no means, means that these are the best nodes, from the network point of view. Thus, can happen of two CHs consider two different nodes to be relays between them, on each communication sense. However, this has a good side since it reflects on the size of the paths passing through the CHs. Figure 4 presents the average path sizes, between CHs, on the network. This measure is important because reflects the traffic of controlling messages, e.g. scheduling, topology management, among the CHs. This traffic can be intense, so the smaller the paths the better.

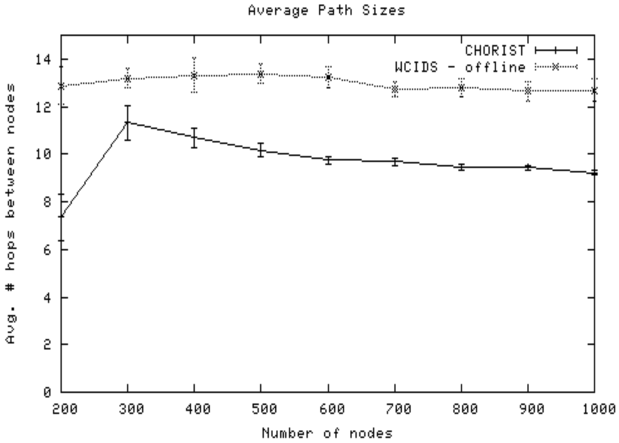

Figure 4. Average path size passing only through CHs and RNs, varing the number of nodes on the network and using 200m communication range

V. CONCLUSIONS

This work presents an implementation and evaluation of the network architecture proposed by the CHORIST project. The problem was reduced to the minimum Weakly Connected Independent Dominating Set. Even though this problem being a NP-complete one, our solution reaches values close to the theoretical minimum, working only with local information and with nodes arriving at the network at different times. From the mobility experiments we can also conclude that implementation of the CHORIST architecture is stable and able to guarantee relatively low percentage of disconnected nodes at the same time it decreases the average path lengths and number of links per nodes. The proposed topology is stable and resilient to nodes mobility.

Eventhough the plannar thechniques seems to be more stable, and capable to provide lower paths, with the increasing in the number of connections, plannar techniques present a high cost in terms of link management and are not scalable from the point of view of higher layers algorithms, i.e. routing.

On the next steps for this work we intend to work in a more detailed analytical analisys of the proposal and implement the proposed algorithm on the fremework of the OpenAirInterface to validate the simulation results with real ones.

REFERENCES

[1] P. Santi, “Topology Control in Wireless Ad Hoc and Sensor Networks”, WILEY, July 2005

[2] J. Q. Bao and W. C. Lee, “Rapid deployment of wireless ad hoc backbone networks for public safety incident management”, in Proc. IEEE Globecom, Nov., 2007

[3] S. F. Midkiff and C. W. Bostian, “Rapidly-deployable broadband wireless networks for disaster and emergency response”, In Proc. First IEEE Workshop on Disaster Recover Networks, June 2002

[4] A. Sarrafi, M. H. Firooz, and H. Barjini, “A Cluster Based Topology Control Algorithm for Wireless Ad-Hoc Networks”, International Conference on Systems and Networks Communication , Oct., 2006

[5] N. Aschenbruck, C. de Waal, P. Martini “Distribution of Nodes in Disaster Area Scenarios and its Impact on Topology Control Strategies”, 2nd MCN, in conjunction with the IEEE Infocom, Phoenix, AZ, USA, April 18, 2008

[6] N. Aschenbruck, E. Gerhards-Padilla, M. Gerharz, Frank, and M. Martini, “Modelling mobility in disaster area scenarios”, In Proceedings of the ACM MSWiM '07, Chania, Crete Island, Greece, October , 2007

[7] R. Gandhi and S. Parthasarathy, “Distributed algorithms for connected domination in wireless networks”, J. Parallel Distrib. Comput., Jul. 2007

[8] T. W. Haynes, S. T. Hedetniemi, Peter J. Slater, “Fundamentals of Domination in Graphs”, CRC; 1 ed., Jan, 1998

[9] M. Garey and D. Johnson, “Computers and Intractability: A Guide to the Theory of NP-Complet”., Freeman, New York, 1979

[10] V. Chvatal, “A greedy heuristic for the set-covering problem”, Math of Operation Research, vol. 4, no. 3, 1979

[11] S. Guha and S. Khuller, “Approximation Algorithms for Connected Dominating Sets”, Algorithmica, volume 20, 1998

[12] http://dcg.ethz.ch/projects/sinalgo/, last visited June 2009

[13] http://www.chorist.eu/, last visited June 2009

[14] http://www.openairinterface.org/, last visited June 2009

[15] E. M. Belding-Royer, “Multi-Level Hierarchies for Scalable Ad hoc Routing” , Kluwer, Wireless Networks 9, 461\u2013478, 2003