ISSN 2277-3916 CVR Journal of Science and Technology, Volume 3, December 2012

Minimization of Backbone Nodes in Wireless Mobile Backbone Networks using Approximation Algorithms

Dr. B. B. Jayasingh[1] and Sumitra Mallick[2]
[1]CVR College of Engineering, Department of IT, Ibrahimpatan, R.R.District, A.P., India
Email: bbjayasingh9@rediffmail.com
[2] Guru Nanak Institute of Technology, Department of IT, Ibrahimpatan, R.R.District, A.P., India
Email: sumitra_mallick2008@rediffmail.com

***Abstract*—We study a novel hierarchical wireless networking approach in which some of the nodes are more capable than others. In such networks, the more capable nodes can serve as Mobile Backbone Nodes and provide a backbone over which end-to-end communication can take place. Our approach consists of controlling the mobility of the Backbone Nodes in order to maintain connectivity. We formulate the problem of minimizing the number of backbone nodes and refer to it as the Connected Disk Cover (CDC) problem. We show that it can be decomposed into the Geometric Disk Cover (GDC) problem and the Steiner Tree Problem with Minimum Number of Steiner Points (STP-MSP). We prove that if these sub problems are solved separately by γ- and δ–approximation algorithms, the approximation ratio of the joint solution is $\gamma+\delta$. Then, we focus on the two sub problems and present a number of distributed approximation algorithms that maintain a solution to the GDC problem under mobility. We show that this approach can be extended in order to obtain a joint approximate solution to the CDC problem.**

***Index Terms*—Approximation algorithms, controlled mobility, distributed algorithms, disk cover, wireless networks.**

I. INTRODUCTION

WIRELESS Sensor Networks (WSNs) and Mobile Ad Hoc Networks (MANETs) can operate without any physical infrastructure (e.g., base stations). Yet, it has been shown that it is sometimes desirable to construct a *virtual backbone* on which most of the multi-hop traffic will be routed [6]. If all nodes have similar communication capabilities and similar limited energy resources, the virtual backbone may pose several challenges. For example, bottleneck formation along the backbone may affect the available bandwidth and the lifetime of the backbone nodes. In addition, the virtual backbone cannot deal with network partitions resulting from the spatial distribution and mobility of the nodes. Alternatively, if some of the nodes are more capable than others, these nodes can be dedicated to providing a backbone over which reliable end-to-end communication can take place.

A novel hierarchical approach for a *Mobile Backbone Network* operating in such a way was recently proposed and studied by Rubin *et al.* (see [7, 9] and references therein) and by Gerla *et al.* (e.g., [8]). We develop and analyze novel algorithms for the construction and maintenance (under node mobility) of a Mobile Backbone Network. We focus on *controlling the mobility* of the more capable nodes in order to maintain network connectivity and to provide a backbone for reliable communication.

A Mobile Backbone Network is composed of two types of nodes. The first type includes static or mobile nodes (e.g., sensors or MANET nodes) with limited capabilities. We refer to them as *Regular Nodes* (RNs). The second type includes mobile nodes with superior communication, mobility, and computation capabilities as well as greater energy resources (e.g., Unmanned-Aerial-Vehicles).We refer to them as *Mobile Backbone Nodes* (MBNs). The main purpose of the MBNs is to provide a mobile infrastructure facilitating network-wide communication. We specifically focus on minimizing the number of MBNs needed for connectivity. Yet, the construction of a Mobile Backbone Network can improve other aspects of the network performance, including node lifetime and Quality of Service as well as network reliability and survivability.

The set of MBNs has to be placed such that (i) every RN can directly communicate with at least one MBN, and (ii) the network formed by the MBNs is connected. We assume a *disk* connectivity model, whereby two nodes can communicate if and only if they are within a certain communication range. We also assume that the communication range of the MBNs is significantly larger than the communication range of the RNs.

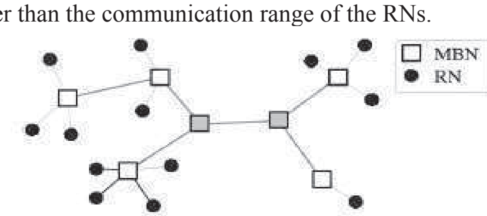

Figure 1. Illustrates an example of the architecture of a Mobile

DOI: 10.32377/cvrjst0316

We term the problem of placing the *minimum* number of MBNs such that both of the above conditions are satisfied as the *Connected Disk Cover (CDC)* problem. While related problems have been studied in the past [2], [6], [3] this papers are one of the first attempts to deal with the CDC problem. Our first approach is based on a framework that *decomposes* the CDC problem into two sub problems. We view the CDC problem as a two-tiered problem. In the first phase, the minimum number of MBNs such that all RNs are *covered* (i.e., all RNs can communicate with at least one MBN) is placed. We refer to these MBNs as *Cover MBNs* and denote them in Fig. 1 by white squares. In the second phase, the minimum number of MBNs such that the MBNs' network is connected is placed. We refer to them as *Relay MBNs* and denote them in Fig. 1 by gray squares. In the first phase, the Geometric Disk Cover (GDC) problem [1, 3] has to be solved, while in the second phase, a Steiner Tree Problem with Minimum Number of Steiner Points (STP-MSP) [4, 5, 10] has to be solved. We show that if these sub problems are solved separately by γ- and δ– approximation algorithms, the approximation ratio of the joint solution is γ+δ.

We develop a number of practically implementable distributed algorithms for covering mobile RNs by MBNs. We assume that all nodes can detect their position via GPS or a localization mechanism. This assumption allows us to take advantage of location information in designing distributed algorithms. We obtain the worst case approximation ratios of the developed algorithms and the average case approximation ratios for two of the algorithms. Finally, we evaluate the performance of the algorithms via simulation, and discuss the tradeoffs between the complexities and approximation ratios.

Our first main contribution is a decomposition result regarding the CDC problem. Other major contributions are the development and analysis of distributed approximation 1A δ approximation algorithm for a minimization problem always finds a solution with value at most δ times the value of the optimal solution. Algorithms for the GDC problem in a mobile environment, as well as the design of a novel Discretization Approach for the solution of the STP-MSP and the CDC problem.

II. Problem Formulation

We consider a set of *Regular Nodes (RNs) distributed in the* plane and assume that a set of *Mobile Backbone Nodes (MBNs)* has to be deployed in the plane. We denote by N the collection of Regular Nodes {1,2,...n} , by M={d_1, d_2, …d_n} collection of MBNs, and by d_{ij} the distance between nodes i and j. An RN_i can communicate bi-directionally with another node $_j$ (i.e., an MBN) if the distance between i and j, d_{ij} <=r, we denote by D=2r the diameter of the disk covered by an MBN communicating with RNs. Regarding the MBNs, we assume that MBN_i can communicate with MBN_j if d_{ij} <= R, where R>r.

We assume that the RNs and MBNs have both a communication channel (e.g., for data) and a low-rate control channel. For the communication channel, we assume the disk connectivity model. Namely, an RN can communicate bi-directionally with another node (i.e., an MBN) if the distance between i and j, d_{ij} <=r. We denote by D=2r the diameter of the disk covered by an MBN communicating with RNs. Regarding the MBNs, we assume that MBN i can communicate with MBN j if d_{ij} <=R, where R>r. For the control channel, we assume that both RNs and MBNs can communicate over a much longer range than their respective data channels. Since given a fixed transmission power, the communication range is inversely related to data rate, this is a valid assumption.

A. Connected Disk Cover (CDC) problem

Given a set of RNs (N) distributed in the plane, place the smallest set of MBNs such that:

1) For every RN_i ε N there exists at least one MBN $_j$ ε M such that d_{ij} <=r
2) The undirected graph G = (M,E) imposed on M (i.e., $\forall_{k,l}$ M, define an edge (k, l) ε E if d_{kl} <=R) is connected.

The RNs are mobile and some of the MBNs move around in order to maintain a solution the CDC problem. We will study both the case in which the nodes are static, and the case in which the RNs are mobile and some of the MBNs move around in order to maintain a solution the CDC problem. We assume that there exists some sort of MBN routing algorithm, which routes specific MBNs from their old locations to their new ones.

We propose to solve the CDC problem by decomposing it into two NP-Complete sub problems: the Geometric Disk Cover (GDC) problem and the Steiner Tree Problem with Minimum number of Steiner Points (STP-MSP).

B. Geometric Disk Cover (GDC)

Given a set N of RNs distributed in the plane, place the smallest set M of cover MBNs such that:

1) For every RN_i ε N there exists at least one MBN $_j$ ε M such that d_{ij} <=r
2) A set of cover MBNs is given and there is a need to place the minimum number of relay MBNs such that formed network is connected.

The second sub problem deals with a situation in which a set of *Cover MBNs* is given and there is a need to place the minimum number of *Relay MBNs* such that the formed network is connected (i.e., satisfies only property (2) in the CDC problem definition). This sub problem is equivalent to the Steiner Tree Problem with Minimum Number of Steiner Points (STP-MSP) [10].

DOI: 10.32377/cvrjst0316

C. Steiner Tree Problem (STP)

Given a set of cover MBNs (M_{cover}) distributed in the plane, place the smallest set of Relay MBNs (M_{relay}) such that the undirected graph G = (M,E) imposed on M = M_{cover} U M_{relay}(i.e., $\forall_{k,l}$ M, define an edge (k, l) ε E if d_{kl} <=R) is connected.

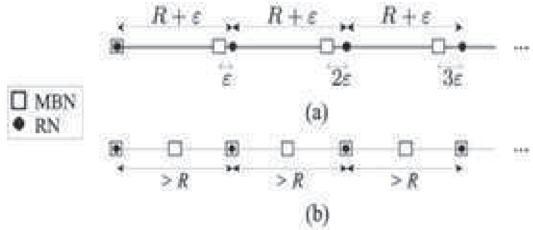

Figure 2. Tight example of the approximation ratio of the decomposition algorithm (a) optimal solution and (b) decomposition algorithm solution.

A tight example of this fact is illustrated in Fig. 2. Fig. 2(a) shows an node instance of the CDC problem, where ε<<r refers to a sufficiently small constant. Also shown is the optimal solution with cost n MBNs. Fig. 2(b) shows a potential solution obtained by using the decomposition framework (with γ=δ=1), composed of an optimal disk cover and an optimal STP-MSP solution. The cost is n+n-1=2n-1MBNs. This example highlights the facts that under the Decomposition Framework, the cover MBNs are placed without considering the related problem of placing the relay MBNs.

III. Placing The Cover MBNS—Strip Cover

Hochbaum and Maass[5, 11] introduced a method for approaching the GDC problem by (i) dividing the plane into equal width strips, (ii) solving the problem locally on the points within each strip, and (iii) taking the overall solution as the union of all local solutions. Below we present algorithms that are based on this method. These algorithms are actually two different versions of a single generic algorithm. The first version locally covers the strip with rectangles encapsulated in disks while the second version locally covers the strip directly with disks. We then generalize (to arbitrary strip widths) the effects of solving the problem locally in strips and use this extension to provide approximation guarantees. Finally, we discuss distributed implementations of these algorithms.

3.1 Centralized Algorithms

For simplicity of the presentation, we start by describing the centralized algorithms. The two versions of the Strip Cover algorithm (*Strip Cover with Rectangles*—SCR and *Strip Cover with Disks*—SCD) appear below. In line 6, the first version (SCR) calls the *Rectangles* procedure and the second one (SCD) calls the *Disks* procedure. *The input is a set of points (RNs) N={1, 2, ...n}* and their (x, y) coordinates, (i_x, i_y) ∀ i. The output includes a set of disks (MBNs) M={d_1, d_2, …d_n} and their locations such that all points are covered. The first step of the algorithm is to divide the plane into K strips of width αD (D=2r). We denote the strips by S_j and the set of MBNs in strip by Ms_j. Figure 3. shows an example of the SCR algorithm and in particular of step 9 in which disks are placed such that they compactly cover all points in the rectangular area with -coordinate range.

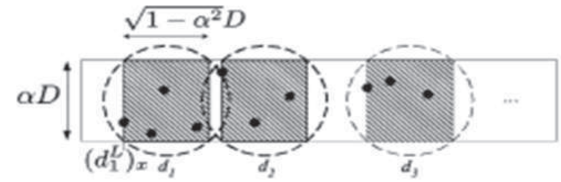

Figure 3. Illustrating step 9 of SCR algorithm

Algorithm 1: Strip Cover with Rectangles/Disks (SCR/SCD):

1. Divide the plane into K strips of width q_{sc} =αD
2. $Ms_j \leftarrow \emptyset$, $\forall_j$ = 1,…,K
3. For all strips S_j, j = 1,…,K do
4. While there exist uncoverd RNs in S_j
5. Let I be the leftmost uncovered RN in S_j
6. Call Rectangles(i) or call Disks(i)
7. $Ms_j \leftarrow Ms_j \cup d_k$
8. Return $\cup_j Ms_j$

Procedure Rectangles(i):

9. Place an MBN d_k such that it covers all RNs in the rectangular area with x-coordinates [i_x,i_x+ $\sqrt{1-\alpha^2}$D]
10. Return d_k

Procedure Disks(i):

11. $Pd_k \leftarrow \emptyset$ {set of RNs covered by the current MBN d_k}
12. While $Pd_k \cup$ i coverable by a single MBN (disk) do
13. $Pd_k \leftarrow Pd_k \cup$ i
14. If there are no more RNs in the strip then
15. Break
16. Let i be the nest leftmost uncovered RN in S_j not currently in Pd_k
17. Place MBN (disk) d_k, such that it covers the RNs Pd_k
18. Return d_k

3.3 Distributed Implementation

The SCR and SCD algorithms can be easily implemented in a distributed manner. The algorithms are executed at the RNs and operate within the strips. The SCR algorithm executed at an RN is described below. Recall that we denote the RNs within a strip according to their order from the left (i.e., i<j if i_x <= j_x).Ties are broken by node ID. Every RN that has no left neighbors within distance D initiates the disk placement procedure that propagates along the strip. The propagation stops once there is a gap between nodes of at least. If an RN arrives from a neighboring strip or leaves its MBN's coverage area, it initiates the disk placement procedure that may trigger an update of

DOI: 10.32377/cvrjst0316

the MBN's locations within the strip. Notice that MBNs only move when a recalculation is required. Although the responsibility to place and move MBNs is with the RNs, simple enhancements would allow the MBNs to reposition themselves during the maintenance phase. If after a recalculation, an MBN is not repositioned, then it is not required and can be used elsewhere. The time complexity (i.e., number of rounds) is O(n). The computation complexity is Olog n). Control information has to be transmitted between RNs over a Distance D=2r.

Algorithm 2: Distributed SCR (at RN i)

Initialization:

1. Let G_i be the set of RNs j such that j<i and $i_x - j_x \leq D$
2. If $G_i = \emptyset$ then
3. Call place MBN

Construction and Maintenance:

4. If MBN placed message received then
5. Call place MBN
6. If i is disconnected from its MBN or enters from a neighboring strip then
7. If there is at least one MBN within distance r then
8. Join one of these MBNs
9. Else call place MBN

Procedure place MBN

10. Let i^R be the rightmost RN s.t $(i^R)_x \leq i_x + \sqrt{1-\alpha^2}D$
11. Place MBN d_k covering RNs j, where $j_x \in [i_x, (i^R)_x]$
12. If $(i^R+1)_x - (i^R)_x \leq D$ then
13. Send an MBN placed message to i^R+1

Algorithm 3: Simple 1-D [13] with $\sqrt{1-\alpha^2}D = 2/3$

1. Initialize the cover greedily {using the SCR algorithm}
2. Maintain the leftmost RN and rightmost RN of each MBN rectangle
3. If two adjacent MBN rectangles come into contact then exchange their outermost RNs
4. If a set of RNs covered by an MBN becomes too long {the separation between its leftmost ad rightmost RNs become greater than 2/3} then

   Split off its rightmost RN into a singleton MBN

   Check whether rule 4 applies
5. If two adjacent MBN rectangles fit in a 2/3 rectangle then

   Merge the two MBNs

Conclusion

The architecture of a hierarchical Mobile Backbone Network has been presented only recently. Such a design can significantly improve the performance, lifetime, and reliability of MANETs and WSNs. We concentrate on placing and mobilizing backbone nodes, dedicated to maintaining connectivity of the regular nodes. We formulated the Backbone Node Placement problem as the Connected Disk Cover problem and showed it can be decomposed into two sub problems. A new approach for the solution of the second sub problem (STP-MSP) and of the joint problem (CDC) has also been discussed. We showed that when it is used to solve the CDC problem in a centralized manner, the number of the required MBNs is significantly reduced.

References

[1] Anand Srinivas*, Gil Zussman, and Eytan Modiano,* Construction and Maintenance of Wireless Mobile Backbone Networks, IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 17, NO. 1, FEBRUARY 2009, pp. 239-252.

[2] D. Chen, D.-Z. Du, X.-D. Hu, G.-H. Lin, L. Wang, and G. Xue, "Approximations for Steiner trees with minimum number of Steiner points," *J. Global Optimization, vol. 18, no. 1, pp. 17–33, 2000.*

[3] M. Franceschetti, M. Cook, and J. Bruck, "A Geometric Theorem for Approximate Disk Covering Algorithms," Caltech, Technical Report ETR035, 2001.

[4] T. Gonzalez, "Covering a set of points in multidimensional space," *Inf. Proc. Lett., vol. 40, no. 4, pp. 181–188, 1991.*

[5] G. Lin and G. Xue, "Steiner tree problem with minimum number of Steiner points and bounded edge-length," *Inf. Proc. Lett., vol. 69, no.* 2, pp. 53–57, 1999.

[6] B. Das, R. Sivakumar, and V. Bharghavan, "Routing in ad-hoc networks using a virtual backbone," in *Proc. IEEE ICCCN'97*, Sep. 1997.

[7] I. Rubin, A. Behzad, R. Zhang, H. Luo, and E. Caballero, "TBONE: A mobile-backbone protocol for ad hoc wireless networks," in *Proc. IEEE Aerospace Conf.*, Mar. 2002.

[8] D. Gu, G. Pei, M. Gerla, and X. Hong, "Integrated hierarchical routing for heterogeneous multi-hop networks," in *Proc. IEEE MILCOM'00*, 2000.

[9] J. Hershberger, "Smooth kinetic maintenance of clusters," in *Proc. ACM SoCG'03*, Jun. 2003.

[10] J. S. Provan, "Convexity and the Steiner tree problem," *Networks*, vol. 18, no. 1, pp. 55–72, 1988.

[11] P. Klein and R. Ravi, "A nearly best-possible approximation algorithm for node-weighted Steiner trees," *J. Algorithms*, vol. 19, no. 1, pp. 104–115, 1995.

DOI: 10.32377/cvrjst0316