Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 63 (2015) 482 – 487

Procedia Computer Science

Seventh International Symposium on Applications of Ad hoc and Sensor Networks (AASNET'15)

Distributed Computation of Connected Dominating Set for Multi-hop Wireless Networks

Simi Surendran, Sreelakshmy Vijayan

AMRITA University,Kollam-690525,India

Abstract

In large wireless multi-hop networks, routing is a main issue as they include many nodes that span over relatively a large area. In such a scenario, finding smallest set of dominant nodes for forwarding packets would be a good approach for better communication. Connected dominating set (CDS) computation is one of the method to find important nodes in the network. As CDS computation is an NP problem, several approximation algorithms are available but these algorithms have high message complexity. This paper discusses the design and implementation of a distributed algorithm to compute connected dominating sets in a wireless network with the help of network spectral properties. Based on local neighborhood, each node in the network finds its ego centric network. To identify dominant nodes, it uses bridge centrality value of ego centric network. A distributed algorithm is proposed to find nodes to connect dominant nodes which approximates CDS. The algorithm has been applied on networks with different network sizes and varying edge probability distributions. The algorithm outputs 40 % important nodes in the network to form back haul communication links with an approximation ratio $\leq 0.04 * \partial + 1$, where ∂ is the maximum node degree. The results confirm that the algorithm contributes to a better performance with reduced message complexity.

Peer-review under responsibility of the Program Chairs

*Keywords:*Connected Dominating Set; Distributed Algorithm

1. Introduction

In large wireless multi hop networks, efficient mode of message transmission is always an overhead. To solve this overhead of passing messages, topology control mechanisms such as finding connected dominating set (CDS) can be devised. Selecting a set of cluster heads helps in sending messages easily, hierarchically from one cluster to the other and the cluster heads will oversee the routing inside the corresponding clusters and also among the different clusters. A connected dominating set[1] is a set D of vertices with the properties that any node in D can reach any other node in D by a path that stays entirely within D and every vertex in graph either belongs to D or is adjacent to a vertex in

Peer-review under responsibility of the Program Chairs
doi:10.1016/j.procs.2015.08.372

D. A Dominating Set is called a CDS if the sub-graph induced by the vertices in the DS is connected. As the nodes in the network are not with the capability to undertake such complex algorithms as for finding the CDS, it is an NP-complete problem and there by cannot compute an optimal solution in polynomial time. Thus, the possible way is to approximate the CDS.

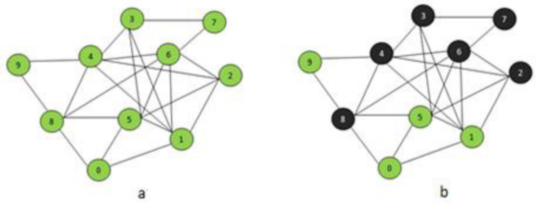

Fig. 1 (a) Initial graph; (b) CDS.

This paper proposes a method to approximate connected dominating set in a large network graphs using the spectral properties of graphs. It computes an ego centric network and uses the partial adjacency matrix for computing the importance factor. Node importance factor is computed using ego network bridge centrality and used a distributed algorithm to form the connection between important nodes. The nodes are considered to be placed arbitrarily according to some probability distribution and are considered to move frequently. The method of finding the CDS is such that it does not transfer much messages for finding CDS and here random geometric graphs are taken. The cluster heads are found which are distributed and such that they form a connection among them, thus forming the connected dominating set. In the proposed work 40% of the total network nodes were found to be selected as members of CDS with reduced message complexity and with a bound $|CDS| = 0.8 * |OPT| + 0.5 * \partial - 2.91$ where, OPT is the minimum connected dominating set and '∂', the maximum degree.

The remainder of the paper is organized as follows: Section 2 deals with the works related to the area. Section 3 describes the algorithms for computation of CDS and the analysis of the results. The performance analysis is presented in Section 4 followed by conclusion.

2. Related Works

The idea of using a CDS as a virtual backbone [7] was first proposed by Ephremides et al. in 1987. There have been many other works which were done based on this backbone formation (CDS). In the earliest works the authors proposes the polynomial self-stabilizing distributed algorithm for the minimal total dominating set problem in an arbitrary graph. In those works there were no connections that were formed among the nodes inside the dominating set. Some of the self-stabilizing algorithms for well known graph problems and some more recent algorithms are given in [17-18]. In the work [4] the authors have presented a polynomial self stabilizing algorithm to construct a minimal total dominating set in an arbitrary graph. This paper also provides polynomial self stabilizing algorithms for the minimal total k-dominating set. It is proved that these algorithms converge in O(mn) moves and the storage required per node was required about O(log n) storage per node. The paper finds the minimal total dominating set by assigning each node with a variable state showing whether the corresponding node is inside or outside the dominating set. It has been found that once a node gets dominated, it remains dominated and also if a node leaves the dominating set, it will never move again.

In the case of [5] the authors present a greedy approximation algorithm for computing a Minimum CDS in multi hop wireless networks with disparate communication ranges and prove that its approximation ratio is better than the best one known in the literature. In the real scenario the situation is different that the communication ranges may differ from nodes to nodes. This work looks for a tighter relation between the independence number and the connected domination number. These numbers are important in the cases of two phase approximation algorithm for CDS where the first phase constructs a dominating set, and the second phase selects additional nodes to interconnect the nodes in the dominating set. In the paper they have found a good approximation bound. size. But, on considering such a situation the routing paths also should be given due consideration and thereby reducing the routing cost. In the method given in [7] there are two steps for finding the backbone network, first is to find MIS(maximal independent set) and then to find the connected dominating set by inducing connection between them. This was done both in general graphs as well as unit disk graph (UDG) and the results in this paper indicate that when the constant value is increased, the size of the CDS can be reduced significantly.

In [6] a fast, silent self stabilizing protocol has been proposed which builds a distance - k - independent dominating set . In the work a set of nodes is a distance – k - independent dominating set if and only if this set is a distance k independent set and a distance k dominating set and an undirected graph is being considered here A set of nodes are distance k dominating if the other nodes are at a distance of at most k. The protocol discussed in the paper tends to reach a legitimate state in 4n+k rounds and here n is the network size. Here each node is given unique identity numbers as well each node maintains a set of shared variables. During the computation each nodes perform an action to jump from one configuration to the other.

In [8] the work has concentrated in the problem of constructing an energy efficient CDS with certainly bounded diameter in the field of wireless networks. Here three main algorithms has been implemented for finding the minimum dominating set, 2 centralized algorithms and one distributed algorithm which is the distributed version of one of them. All the algorithms that exist are either centralized or distributed. Centralized algorithms usually yield smaller CDS with better performance ratios than distributed ones. Our work is more related to the algorithms [10-15] which were proposed with a motive of fast approximation with small performance ratio. The algorithms mainly consist of two steps where in the first stage n MIS is found and in the second stage some nodes are selected as connectors to form the CDS.

3. Connected Dominating Set Computation

In large wireless networks, the nodes are distributed and spans over a wide range of area. Each node will be having only a partial knowledge of the whole network so; CDS computation can be done in a distributed manner. In this paper, we computed dominating sets based on Maximal Independent Set and ego network centrality. A distributed algorithm is used to connect the dominators in the connected dominating set.

3.1 Independent set based CDS computation

Here, we assume each node to be assigned with a unique id and each of them being in a NON-IND state initially. Every NON-IND node sent its ID to its immediate neighbours and then compares its ID with all the received IDs from its neighbours. If its ID is smaller than every received ID, then it turns from the initial NON-IND state to MIS. Now, every MIS nodes send message MIS to its neighbours. If a NON-IND node receives a message MIS, then it turns its state from NON-IND to NON-MIS. These procedures are repeated until no NON-IND node exists. The left over set of MIS nodes form the Maximal Independent Set. A Connection Algorithm presented in algorithm 3 is used to connect these MIS nodes to form final CDS.

3.2 Centrality based CDS construction

In large wireless networks, we can compute the connected dominating sets based on the spectral properties of the network. If the network is very large, it is difficult to get the global topology of the network. So based on the neighbourhood relationship of each node, we can generate the partial adjacency matrix of each network zone. Finding nodes with highest degree and bridge centrality in this partial adjacency matrix gives most important nodes in the network. The set of these important nodes and connector nodes to connect important nodes yields CDS in the network. The bridging centrality [2] of a node is the product of the betweenness centrality and the bridging coefficient (BC), which measures the global and local features of a node.

$$B_{centrality}(v) = BC(v) \times C_{Betweenness}(v) \quad (1)$$

$$C_{Betweenness}(v) = \sum_{\substack{s \neq v \neq t \\ s,v,t \in V}} \frac{\delta_{st}(v)}{\delta_{st}} \quad (2)$$

$$BC(v) = \frac{d(v)^{-1}}{\sum_{i \in N(v)} \frac{1}{d(i)}} \quad (3)$$

where $B_{centrality}\ (v)$ is the bridging centrality of the node which is the product of bridging coefficient BC(v) and

Algorithm 1. Graph Partitioning

1. **For** *each node u* do
    Create a list L containing itself and all its immediate neighbours
2: *Create an empty matrix of size as* $L * L$
3: **For** *each pair u,v in L do*
4: **if** *they are neighbours then*
5: *L [u] [v] = 1*
6: *else*
7: *L [u] [v] = 0*
8: **End if**
9: **End For**

betweenness centrality $C_{Betweenness}$. $C_{Betweenness}$ is the fraction of total number of shortest paths going fron sto t through node v to the total number of shortest paths between s and t. $\boldsymbol{\delta_{st}}$ represents the shortest path between s and t. Bridging coefficient is defined as ratio of reciprocal of degree of a given node to the sum of reciprocals of degrees of its neighbours $BC(v)$. The given network is at first partitioned among the nodes by generating different ego networks based on each nodes k-hop neighbourhood. This is implemented by finding the adjacency matrix of the original graph . For finding the ego-network, each node is taken along with its immediate neighbours and a corresponding matrix is generated whose entries are similar to the original adjacency matrix forming a small sub graph which including the node itself and its neighbours. Now, with the obtained ego networks, the ego-centric bridge centrality for each node is calculated for which a threshold value is set. For all nodes passing the threshold will be selected as a dominator collectively forming the independent set or the dominating set.

Algorithm 1 shows the network partitioning procedure. Each node find its immediate neighbours and create a neighbourhood matrix which shows the relationship between neighbours in the network. If any two nodes in the list are direct neighbours, then the neighbourhood matrix entry will be 1 otherwise 0. Algorithm 2 describe the steps to find bridge centrality of each node in the network. It outputs the nodes with their bridging factor. Select the nodes as dominant nodes if their centrality value crosses threshold. To connect the dominant nodes, a distributed algorithm is presented in Algorithm 3.

Algorithm 2. *Centrality Based Dominating Set Construction*

For *each node in the network* **do**
    Compute the egocentric adjacency matrix A
End
    For *all nodes* ***do***
        Compute A^k *[1-A]*
        Find $C_B\ (v)$*=Σ1/Ai For all Ai ≥ 1*
        Find the degree value d(v) of each node in the graph

$$BC(v) = \frac{d(v)^{-1}}{\sum_{i \in N(v)} \frac{1}{d(i)}}$$

$$B_{centrality}(v) = BC(v) \times C_B\ (v)$$

Algorithm 3. *Connecting Dominant Nodes*
For *each node marked dominant* **do**
Map its COMPONENT value to its own ID.
For *each node u marked candidate* **do**
Map its count value to the number of its adjacent dominant nodes who are in different components.
Find the competitors of candidate nodes
Form a dominant list which consists of neighbouring dominant nodes of u with its COMPONENT value .
For *each node u marked candidate* **do**
if *u ranked higher than all its competitors then*
Mark it as dominant.
Send an dominant updating message to all neighbours
On receiving UP dominant updating message of u
Competitors of u updates its competitor list and dominant list
Update COMPONENT id

4. Results

Random graphs with varying edge probability distributions are applied to the CDS computation algorithm. The nodes with maximum bridge centrality is considered as the dominant nodes. Fig 2a shows the sample input graph and the dominant nodes and connected dominated set is shown in fig 2b.

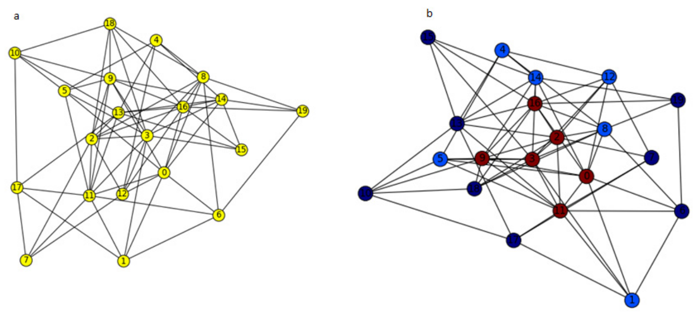

Fig. 2. (a) Input network (b) dominant nodes in red color.

Table 1. Bridging centrality of nodes in Fig 2.

Node	Bridge Centrality	Node	Bridge Centrality	Node	Bridge Centrality
0	0.0713	7	0.015	14	0.022
1	0.015	8	0.028	15	0.004
2	0.034	9	0.049	16	0.048
3	0.051	10	0.030	17	0.019
4	0.028	11	0.049	18	0.030
5	0.038	12	0.047	19	0.009
6	0.018	13	0.046		

The results were analyzed by computing the approximation ratio. The analysis was based on finding the minimum CDS and then comparing it with the obtained results of the proposed algorithm. The results proved that the algorithm constructs CDS with an approximation ratio $\leq 0.04 * \partial + 1$ and with CDS size bounds as $|CDS| = 0.8 * |OPT| + 0.5 * \partial - 2.91$ or $|CDS| = 0.65 * \partial + 2.6$. Figure 4 is a graphical representation on the variation of CDS Size and maximum degree with respect to node size. Figure 5 represents the variation of CDS Size with respect maximum degree. From the graph, it is seen that maximum degree varies linearly as the number of nodes are increased while the CDS size shows a slight variation.

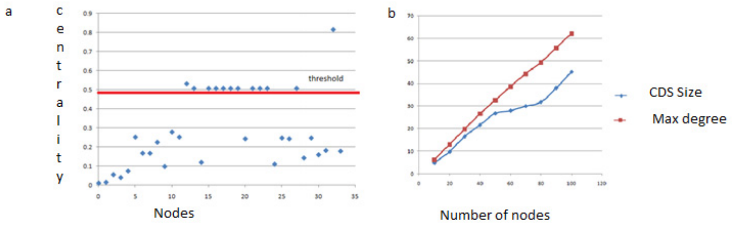

Fig. 3. (a) Dominant nodes above threshold bridging centrality (b) variation of CDS size with number of nodes

Conclusion

Wireless multi-hop networks have gained much attention in recent years due to their self-organising characteristics. Identifying important nodes in the network helps to improve packet forwarding scheme. In this paper we presented a method to approximate the connected dominating set in the network with less computational effort. The algorithm computed the importance factor of each node using bridging centrality measure without knowing the global topology of the network. Using a distributed connection algorithm, dominant nodes are connected and any node in the network can reach any other node using this CDS. The proposed method was able to select 40% of the total network nodes as dominating nodes. The approximation ratio was found to be $\leq 0.04 * \partial + 1$, where ∂ is the maximum node degree. The results have shown that the CDS bound using the proposed algorithm was $|CDS| = 0.65 * \partial + 2.6$.

References

1. S. Guha, S. Khuller, Approximation algorithms for connected dominat ing sets, Proceedings of 4th Annual European Symposium on Algorithms, (1996).
2. Woochang Hwang, Young-rae Cho, Aidong Zhang, Murali Ramanathan , Bridging Centrality: Identifying Bridging Nodes In Scale-free Networks, KDD (2006)
3. Lixin Wang, Peng-Jun Wan, and Frances YaoMinimum CDS in Multi hop Wireless Networks with Disparate Communication Ranges, IEEE(2013).
4. Yacine Belhoul a, b, Sad Yahiaouia,b,Hamamache KheddoucicE_cient self-stabilizing algorithms for minimal total k-dominating sets in graphs,(2013).
5. Hongwei (David) Du, Member, Weili Wu, Qiang Ye, Member, Deying Li, Wonjun Lee, , and Xuepeng XuCDS-Based Virtual Backbone Construction with Guaranteed Routing Cost in Wireless Sensor Networks, IEEE, (2013).
6. Colette Johnen Fast, self-stabilizing distance k independent dominating set construction, (2014).
7. Well Y.Chiu, Chiuyuan Chen, Shih-Yu Tsai, A 4n-move self-stabilizing algorithm for the minimal dominating set problem using an unfair distributed daemon, (2014).
8. Donghyun Kim, Yiwei Wu, sYingshu Li, Feng Zou, and Ding-Zhu Du, Member Constructing Minimum Connected Dominating Sets with Bounded diameters in Wireless Networks, IEEE,(2009).
9. A. Ephremides, J. Wieselthier, and D. Baker, A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling, IEEE, (1987)
10. M. Cardei, M.X. Cheng, X. Cheng, and D.-Z. Du, Connected Domination in Ad Hoc Wireless Networks, Proc. *Sixth International Conference on Computer Science and Informatics* (CS&I), 2002.
11. S. Funke, A. Kesselman, and U. Meyer, A Simple Improved Distributed Algorithm for Minimum CDS in Unit Disk Graphs,*ACM Trans. Sensor Networks*, (2006).
12. P.-J. Wan, L. Wang, F. Yao, and C.-W. Yi, Two-Phased Approximation Algorithms for Minimum CDS in Wireless Ad Hoc Networks, Proc. IEEE 28th *Int'l Conf. Distributed Computing Systems (ICDCS)*, (2008).
13. Y.S. Li, M.T. Thai, F. Wang, C.-W. Yi, P.-J. Wan, and D.-Z. Du, On Greedy Construction of Connected Dominating Sets in Wireless Networks, Wiley *J. Wireless Comm. and Mobile Computing* (2005).
14. M. Min, X. Huang, C.-H. Huang, W. Wu, H. Du, and X. Jia, Improving Construction for Connected Dominating Set with Steiner Tree in Wireless Sensor Networks, J. *Global Optimization*, vol. 35, no. 1, pp. 111-119, 2006.
15. M. Li, P.-J. Wan, and F.F. Yao, Tighter Approximation Bounds for Minimum CDS in Wireless Ad Hoc Networks, Proc. 20th *Int'l Symp. Algorithms and Computation* (ISAAC '09), (2009).
16. S.Hedetniemi, S.Hedetniemi, H.Jiang, K.Kennedy, A. McRae, A self-stabilizing algorithm for optimally efficient sets in graphs, *Inf.Pro-cess.Lett*.112 (2012)621–623.
17. J.Blair, F.Manne, An efficient self-stabilizing distance-2 coloring algorithm, *Theor.Comput.Sci*.444 (2012).
18. S.Devismes,K.Heurtefeux,Y.Rivierre,A.Datta,L.Larmore,Self-stabilizing small k-dominating sets ,in :*Second International Conference on Networking and Computing*(2011)