Volume 2 No.4, APRIL 2011 ISSN 2079-8407
Journal of Emerging Trends in Computing and Information Sciences

http://www.cisjournal.org

Self-Stabilizing Leader Election Algorithm in Highly Dynamic Ad-hoc Mobile Networks

[1]K Ranganath, [2]L.Naveen Kumar, [3]Y.V.Sreevani
Department of Computer Science and Engineering Department
Hyderabad Institute of Technology and Management
Hyderabad, A.P, India.
[1]ranga1kanakam@gmail.com, [2]naveen.laskari@yahoo.com, [3]s_vanikumar@yahoo.co.in

ABSTRACT

We propose a self-stabilizing leader election algorithm that can tolerate multiple concurrent topological changes. By introducing the time interval based computation concept, the algorithm ensures that a network partition within a finite time converge to a legitimate state even if topological changes occur during the convergence time. An ad hoc network is a collection of mobile nodes forming a temporary network without any form of centralized administration or predefined infrastructure. In such a network, each node participating in the network acts as both a host and a router. Two nodes can communicate if they are within the transmission range of each other. Due to node mobility, link breakages and link formations might occur frequently. The failure of some links considered as critical and can split up the network into several disjoint network components. In addition, multiple components can also merge into a single connected component. In this paper we have investigated the functional system with the proposed algorithm and how it monitors the mobile non static hosts and the transmission process between them.

Keywords: *Self-stabilizing, Leader node, A**d hoc networks**, Bandwidth, MANET.*

I. INTRODUCTION

This paper proposes a self-stabilizing leader election algorithm that can work in highly dynamic and asynchronous ad hoc networks [1], [2]. The algorithm uses the stabilizing based approach for designing a leader election algorithm. This approach has the advantage of detecting partitions automatically using the TORA mechanism. It has fewer message overheads as compared to the algorithm. It is localized since the knowledge of each node is limited to one hop. Nodes have synchronized clocks, either through a global positioning system (GPS) or through an appropriate algorithm. The paper addresses an open issue that has been tackled by other previous works. The issue is how to guarantee that a self stabilizing algorithm converges to a legitimate state despite frequently changing ad hoc networks. In the literature, an algorithm is self-stabilizing if it can converge to a legitimate state in a finite time regardless of the initial state, and the system remains in a legitimate state until another topological change occurs. However, the convergence is guaranteed only when the network experiences no topological changes during the convergence time. The solutions presented earlier cannot work in highly dynamic environments like ad hoc networks, because they assume that topological changes stop from some point onward in order for the algorithms to stabilize. When the system experiences a new topological change before completing the convergence, the algorithms restart convergence to its legitimate state from scratch. In this case, the algorithms might never converge to the legitimate state.

II. BACKGROUND

A. Manets

A MANET consists of mobile platforms (e.g. a router with multiple hosts and wireless communications devices), here in simply referred to as **nodes** which are free to move about randomly [3]. The nodes may be located in or on airplanes, ships, trucks, cars, perhaps even on people or very small devices, and there may be multiple hosts per router. A MANET is an autonomous system of mobile nodes as shown in figure 1 and 2. The system may operate in isolation, or may have gateways to interface with a fixed network. In the latter operational mode, it is typically envisioned to operate as a **stub** network connecting to a fixed internet work. Stub networks carry traffic originating at or destined for internal nodes, but do not permit exogenous traffic to **transit** through the stub network.

MANET characteristics:

- Dynamic topologies: Nodes are free to move arbitrarily,Thus the network topology, which is typically multi hop-may change randomly and rapidly at unpredictable times, and may consist of both bidirectional and unidirectional links [4].
- Bandwidth-constrains and variable capability links: Wireless links will continue to have significantly lower capacity than their hardwired counterparts. In addition, the realized throughput of wireless communications after accounting for the effects of

multiple access, fading, noise, and interference conditions, etc. is often much less than a radio's maximum transmission rate [6]. One effect of the relatively low to moderate link capacities is that congestion is typically the normal rather than the exception, i.e. aggregate application demand will likely approach or exceed network capacity frequently. As the mobile network is often simply an extension of the fixed network infrastructure, mobile ad hoc users will demand similar services. These demands will continue to increase as multimedia computing and collaborative networking applications rise.

- Energy-constrained operation: Some or all of the nodes in a MANET may rely on batteries or other exhaustible means for their energy. For these nodes, the most important system design criteria for optimization may be energy conservation [5].
- Limited physical security: Mobile wireless networks are generally more prone to physical security threats than are fixed- cable nets. The increased possibility of eavesdropping, spoofing, and denial-of-service attacks should be carefully considered.

B. Slot Synchronization

The basic idea of the proposed decentralized synchronization scheme is to achieve locally common slot timing by a mutual adaptation of the individual slot timing. The synchronization procedure consists of two steps: First, the slot timing of a received burst is acquired in form of a one-shot synchronization. In the second step, the own slot timing is adapted according to the observed time difference to the node that transmitted the respective burst. Wireless network has become very popular in the computing industry. Wireless network are adapted to enable mobility. There are two variations of mobile network [4]. The **first** is infra-structured network (i.e. a network with fixed and wired gateways). The bridges of the network are known as base stations. A mobile unit within the network connects to and communicates with the nearest base station (i.e. within the communication radius). Application of this network includes office WLAN. The **second** type of network is infrastructure less mobile network commonly known as Ad-hoc network. They have no fixed routers. All nodes are capable of moving and can be connected in an arbitrary manner. These nodes function as routers, which discover and maintain routes to other nodes in the network. Non infrastructure based Manets are expected to become an important part of the 4G architecture. Ad-hoc networks can be used in areas where there is little or no communication infrastructure or as the existing infrastructure is expensive or inconvenient to use. Some applications of ad-hoc network are users using laptop to participate in an interactive lecture, business associates sharing information during a meeting, soldiers relaying information about situation awareness in a battlefield, and emerging disaster relief after an earthquake or hurricane [5]. Ad-hoc networks are created, for a group of people come together and use wireless communication for some computer based collaborative activities, this is also referred to as spontaneous networking.

C. The Communication Model

Protocols can be designed based on multi-channel and single channel communication. Multi-channel protocols are low-level routing protocol, which combines channel assignment and routing functionality. Such protocols are used in TDMA or CDMA based networks also includes CGSR and larger classes of protocols assume that nodes communicate over a single logical wireless channel. These protocols are CSMA/CA oriented, where in they rely on specific link layer behaviours. Some MANET routing protocols are based on specific link layer properties, such as RTS/CTS control sequence used by popular IEEE 802.11, MAC layers to avoid collisions due to hidden and exposed terminals. Specifically, before transmitting a data frame the source station sends a short control frame, named RTS, to the receiving station announcing the upcoming frame transmission [4]. Upon receiving the RTS frame the destination station replies by a CTS frame to indicate that it is ready to receive the data frame. Both the RTS and CTS frames contain the total duration of the transmission that is the overall time needed to transmit the data frame and the related ACK. This information can be read by any station within the transmission range of either the source or the destination station. Hence, the station becomes aware of transmission from hidden station and the length of time the channel will be used for transmission [10]. The exposed terminal problem results from situations where a permissible transmission from a mobile station to another station has to be delayed due to the irrelevant transmission activity between two other mobile stations within the sender's transmission range [11]. It is worth pointing that the hidden station and the exposed station problems are correlated within the transmission range. The transmission range is determined by the transmission power and the radio propagation properties. By increasing the transmission range, hidden station problem occurs less frequently but the exposed station problem becomes more important as the transmission range identifies the area affected by the single transmission. In addition to the transmission range the Physical carrier sensing range and the interference range must be considered to correctly understand the behaviour of wireless networks. There are few protocols that are based on physical layer information such as signal strength, or geographic position into the routing algorithm been fantastic in terms of the unpredictable growth of the Internet and the new ways in which people are able to communicate with one another [8]. Spearheading this phenomenon has been the World Wide Web (WWW), with thousands of new sites being launched daily and consumers being consistently offered numerous outstanding services via this new communications medium.

III. SYSTEM STRUCTURE AND IMPLEMENTATION

The system network is initialized with the number of nodes and number of edges as a basic input. After initialization some nodes are leave from the network or some nodes would join the network randomly. The pseudo code for the algorithm is depicted in the following section. These changes are to be observed and if the nodes leave the network then its neighbours has to be intimated and the information is to be dispersed to remaining nodes [5]. Later the algorithm assigns weights to each edge, and identifies the source node and destination node [6]. Thus keeping track of the neighbours to each corresponding node. The results are depicted in details as output screens shots figures 2-6.

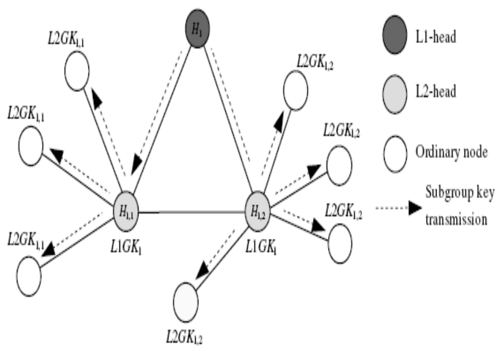

Fig. 1. Subgroup key transmission operation between nodes.

Upon execution of the system we can determine the leader node among available nodes. It is based on weights of each node [6], [7]. The weights depend on frequency of message sending, receiving, number of neighbours and is computed by stabilizing Algorithm [9]. If the leader node leaves the network due to network problems then that information passed to its neighbor's and subsequently the stabilizing algorithm select other node as a leader node or coordinator node.

A. Pseudo Code for the Algorithm

```
nodes = 0;   edges = 0;  Sstab = new int[100][100];
weights = new int[100];
 for(int i=0;i<100;i++)
 for(int j=0;j<100;j++)  Ssatb[i][j] = 0;
 for(int i=0;i<100;i++)  weights[i] = 0;
 leaderNode = -1;
 leaderWeight = 0;
 jTextArea1.append ("sstab");
 jTextArea1.append ("\n");
 String str="";
 for(int i=0;i<nodes;i++)
{
    for(int j=0;j<nodes;j++)
     {
 jTextArea1.append (""+sstab[i][j]+"  ");
      }
 jTextArea1.append ("\n");
  }
int i = Integer.parseInt(jTextField3.getText());
int j = Integer.parseInt(jTextField4.getText());
jTextArea1.append ("\n");
 int max=0;     int in = -1;
 for(int i=0;i<nodes;i++)

{
    if(weights[i]>max)  { max = weights[i] ;   in = i;    }
}
 Ssatb[i][j] = 1;  edges++;
 jTextArea1.append ("Edge "+i+" --> "+j+"Added");
  leaderNode = in;   leaderWeight = max;
 jTextArea1.append ("Leader Node:"+leaderNode);
 jTextArea1.append ("\n");    jTextArea1.append ("Leader
Weight:"+leaderWeight); jTextArea1.append ("\n");
```

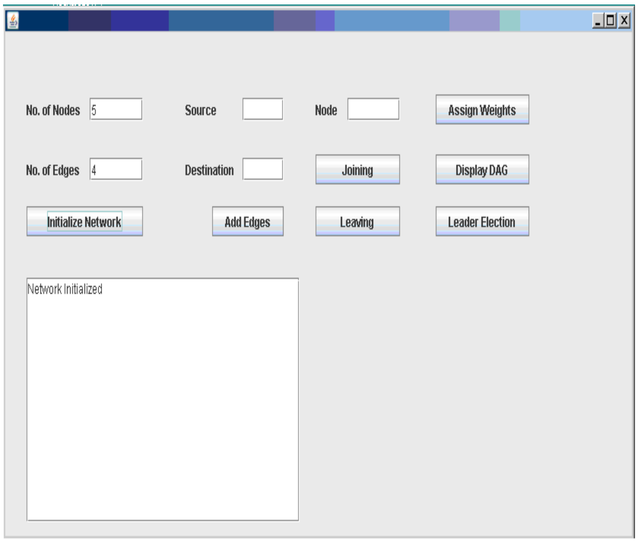

Figure 2: Initialize the Network

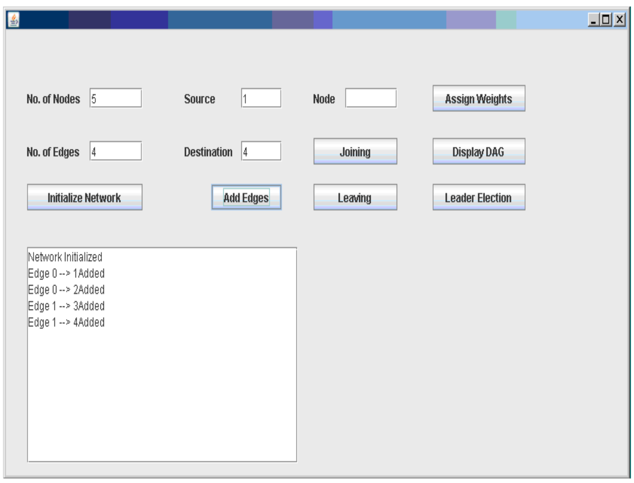

Figure 3: Assign the weights to each edge

Volume 2 No.4, APRIL 2011 ISSN 2079-8407
Journal of Emerging Trends in Computing and Information Sciences

http://www.cisjournal.org

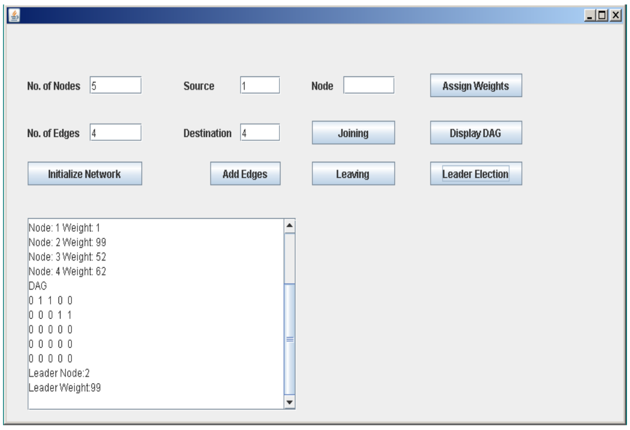

Figure 4: display the stabilizing matrix for Leader node

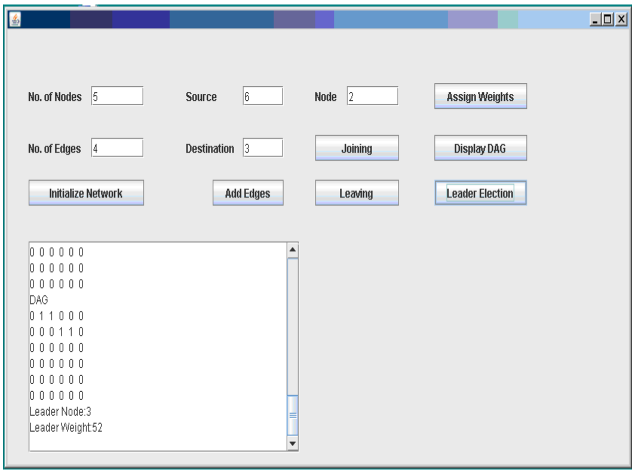

Figure 5: Display the leader node along with its Weight

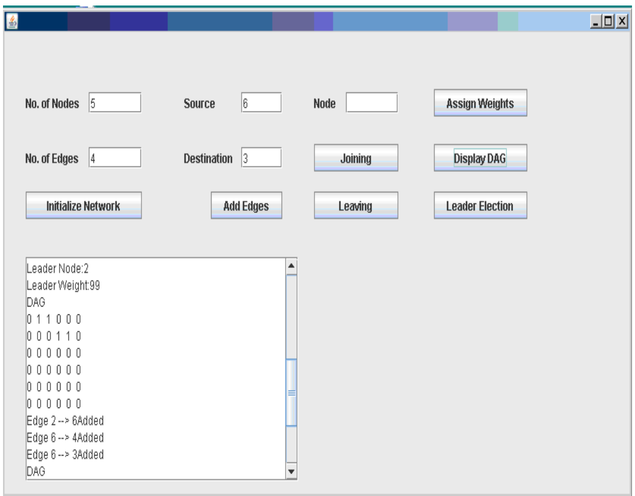

Figure 6: leader node leaving the network

IV. CONCLUSION

In this paper, we have proposed a self-stabilizing leader election algorithm that can converge to a legitimate state even in the presence of topological changes during convergence time. By defining concurrent and disjoint computations and their corresponding intervals, an older reference level encompasses any new one belonging to its equivalence class. In the same way, an older Stabilizing propagation encompasses new ones. The work illustrated here can be extended by incorporating extensively in terms of topology changes ,concurrent stabilizing propagations in a network components , concurrent reference level propagations, concurrent reference level propagations in partitioned networks, concurrent merging of networks.

REFERENCES

[1] A.D. Amis, R. Prakash, T.H.P. Vuong, and D.T. Huynh, "Maxmin D-Cluster Formation in Wireless Ad Hoc Networks," Proc. IEEE INFOCOM '00, pp. 32-41, 2000.

[2] N. Asokan and P. Ginzboorg, "Key Agreement in Ad Hoc Networks," Computer Comm., vol. 23, no. 17, pp. 1627-1637, 2000.

[3] T.-C. Hou and T.-J. Tsai, "An Access-Based Clustering Protocol for Multihop Wireless Ad Hoc Networks," IEEE J. Selected Areas in Comm., vol. 19, no. 7, July 2001.

[4] B. Lehane, L. Dolye, and D. O'Mahony, "Ad Hoc Key Management Infrastructure," Proc. Int'lConf. Information Technology: Coding and Computing (ITCC '05), vol. 2, pp. 540-545, 2005.

[5] N. Malpani, J.L. Welch, and N. Vaidya, "Leader Election Algorithms for Mobile Ad Hoc Networks," Proc. Fourth Int'l Workshop Discrete Algorithms and Methods for Mobile Computing and Comm., pp. 96-103, 2000.

[6] A.B. McDonald and T.F. Znati, "A Mobility- Based Framework for Adaptive Clustering in Wireless Ad Hoc Networks," IEEE J. Selected Areas in Comm., vol. 17, no. 8, pp. 1466-1487, Aug. 1999.

[7] V. Ramasubramanian, R. Chandra, and D. Mosse, "Providing a Bidirectional Abstraction for Unidirectional Ad Hoc Networks," Proc. IEEE INFOCOM, 2002.

[8] G.-C. Roman, Q. Huang, and A. Hazemi, "Consistent Group Membership in Ad Hoc Networks," Proc. 23rd Int'l Conf. Software Eng. (ICSE '01), pp. 381-388, 2001.

[9] K. Ro¨mer, "Time Synchronization in Ad Hoc Networks," Proc. ACM MobiHoc '01, pp. 173-182, 2001

[10] J. So and N. Vaidya, "MTSF: A Timing Synchronization Protocol to Support Synchronous Operations in Multihop Wireless Networks," technical report, Univ. of Illinois,Urbana-Champaign, Oct. 2004.

[11] S. Vasudevan, J. Kurose, and D. Towsley, "Design and Analysis of a Leader Election Algorithm for Mobile Ad Hoc Networks," Proc.12th IEEE Int'l Conf. Network Protocols (ICNP '04), pp. 350-360, Oct. 2004.

AUTHORS :

Mr.K.Ranganath, Graduated in Computer Science and Engineering from Osmania University Hyderabad, India, in 2006 and M.Tech in Software Engineering from Jawaharlal Nehru Technological University, Hyderabad, A.P., India . He is presently working as Assistant Professor in Department of Computer Science and Engineering, Hyderabad institute of Technology and Management [HITAM], Hyderabad, A.P, India. A keen research scholar and has many papers published to his credit. His research interests include Mobile Computing and Data Mining.

Mr. Naveen Kumar Laskari has received the Bachelor Degree and Master Degree from JNT University, Hyderabad, India. He is Assistant Professor in the Department of Computer Science and Engineering in Hyderabad Institute of Technology and Management [HITAM], Hyderabad, A.P, India. He has presented papers and participated in number of seminars and workshops across India. His Research interest includes Image Processing, Data Mining and Information Security.

Mrs.Y.V.Sreevani, Graduated in A.M.I.E.T.E. from I.E.T.E, New Delhi, India, in 1997 and M.Tech in Computer Science from Osmania University, Hyderabad, A.P., India in 2003. Currently working as Associate professor and research scholar in Department of Computer Science and Engineering in Hyderabad Institute of Technology and Management [HITAM], Hyderabad, A.P, India. She has 8 years of experience in computer technology and I.T related projects. Her research interests include Data mining, Distributed Systems, Information Retrieval systems.