International journal of Engineering Research-Online
A Peer Reviewed International Journal
Articles available online http://www.ijoer.in

Vol.1., Issue.4., 2013

RESEARCH ARTICLE

ISSN:2321-7758

IDENTIFYING AND AVOIDING SELFISHNESS IN REPLICA ALLOCATION OVER 'MANET'

D SHEETAL SHARMA[1], RAVI KIRAN VARMA P
[1]M.Tech Student MVGR College of Engineering, Vizianagaram, A.P

Article Received: 17/11/2013 | Article Revised on:26/12/2013 | Article Accepted on:29/12/2013

D SHEETAL SHARMA

ABSTRACT
In mobile ad hoc network the mobile nodes move freely across the network leading to frequent network division and in turn decrease in data accessibility and performance. In order to overcome this problem there are many replica allocation methods proposed by different authors. It is assumed that all mobile nodes participate fully in self-organizing functions. Some nodes may decide against cooperating with others. Some nodes may act egoistically and decide only to contribute partly or not at all, with other nodes. Selfish behavior of node can probably result in wide range of problems for MANET. The main aim is to avoid the selfishness problem in the framework of replica allocation in a MANET.
The main aim is to estimate the selfish nodes in Mobile ad-hoc networks which lead to the performance degradation of the network and to improve some parameters like data accessibility, query delay.
Keyword: selfish node, SCF tree, replica allocation, MANET

INTRODUCTION

A Mobile Ad Hoc Network is a self-organizing, productive wireless network of autonomous mobile devices (nodes) where no special infrastructure is required. The network is called ad hoc because there is no fixed and known network structure that every other node forwards data. Mobile ad hoc network (MANET) is popular research area due to the rapid growth of mobile wireless communications. In Mobile Ad Hoc Network (MANET) each node represent different devices like Laptop computers, personal digital assistants (PDAs) and mobile phones [2]. Nodes in MANET are battery powered devices and have certain resource constraints like memory usage and bandwidth. In ad hoc network the nodes apart from acting as source and destination, also plays the role of a router, host and perform network functions like routing, multi-hop packet delivery, and mobility management. Technology of MANET is used in many different applications ranging from military communication system, wireless peer-to-peer networks to emergency search/rescue operations.

As nodes in MANET move freely from one network to other, disconnections occur frequently, and this causes frequent network division which in turn raises several new challenges to data availability and access efficiency for data access applications. One possible solution for this challenge is to employ replication

techniques, which improve data availability and decrease query response delay. For suppose network is partitioned into two due to the migration of nodes and the node in one of the divided two networks say n1 is unable to access data item d2 of node n2 which is in other network and the node n2 is unable to access the data item d1 of node n1 resulting in deterioration of data accessibility at the point of network division as shown in "Figure 1". Here data replication technique can be employed as it is a traditional technique for improving data availability [7] i.e. if replicas of data item d1 and d2 are allocated to one of the nodes in opposite network, then every node will be able to access both data items even after network division.

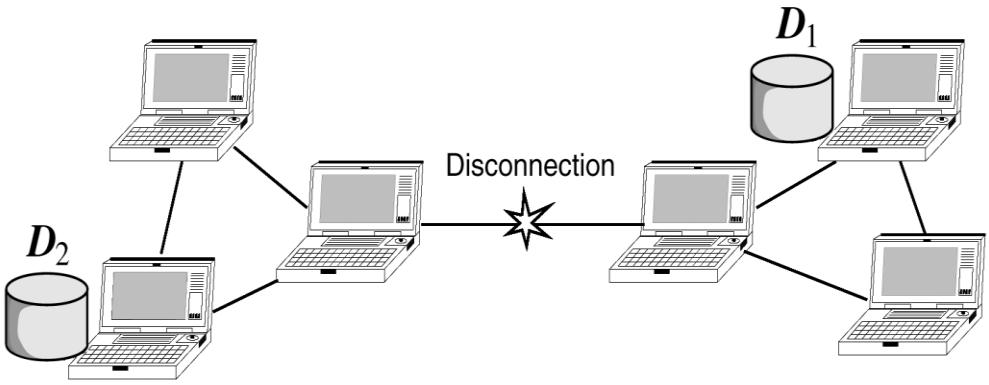

Figure 1. Showing network division and disconnection

Although many replication technique are used to increase data availability [7] but some nodes act selfishly increasing the query response delay. These selfish nodes are also called as misbehaving nodes which use the network for their own communication purpose and refuse to cooperate in packet forwarding for other nodes in order to save their battery power.

A selfish node utilizes the resources provided by other nodes but will not avail its own resources for the benefit of others. These selfish nodes are different from malicious nodes, unlike malicious nodes these selfish nodes does not cause any damage to the network. In this paper selfish nodes are identified and detected and using SCF tree replica allocation is done.

Related Work: Reference [3] proposed two network-layer acknowledgment-based schemes, termed the TWOACK and S-TWOACK schemes, which can be merely added on to any supply routing protocol like the DSR protocol. The schemes are used to detect the selfish nodes in order to improve end to end packet delivery ratio. The results of simulation show that 40 percent of the nodes are misbehaving and the TWOACK scheme improves the end to-end packet delivery quantitative relation from around 70 percent to almost 90 percent while increasing the overhead from 4% to 7 % [3]. The STWOACK scheme which is spinoff of the TWOACK scheme, achieves nearly an equivalent performance improvement without any routing overhead, however the TWOACK scheme have some overhead problems and also the S-TWOACK theme have some expected increase of false alarms.

A new method was proposed where a mathematical technique named game-theoretic [1] is used in order to achieve cost-efficiency for routing layer of mobile unplanned networks. In this paper [1] the author has introduced Ad hoc-VCG, a reactive routing protocol for mobile unplanned network which accomplishes the task of cost-efficiency. Ad hoc-VCG works well when the two conditions are met, they are firstly the communication session taking place between two nodes must be long and secondly during the communication session the routing path must not change very often. If these conditions are not met the overhead of route discovery phase will halt the network.

In this paper [5] author has proposed five different routing approaches having different complexity for peer to peer file sharing over mobile unplanned networks. The quality of the projected approaches is evaluated and compared. In this paper comparisons among various routing protocols in terms of routing

quality, measurability, implementation quality, maintenance quality, energy potency, the shortest path and cross-layer property is done and the DHT routing table and neighbourhood table need to be maintained. This paper [4] analyses the performance of each protocol in ad hoc network. In this paper different routing protocols like DSDV, TORA, DSR, and AODV are studied and simulations were conducted using two different node movement speeds: a maximum speed of 20 m/s (average speed 10 m/s) and a maximum speed of 1 m/s. The simulation is done with 50 nodes. Each protocol which is studied performs well in some cases but has some drawbacks in other. The performance of AODV and DSR were very good at all mobility rates and movement speeds [4] and the goal of eliminating source routing overhead is achieved.

In this paper [12] author has studied how multicast in selfish wireless network can be conducted efficiently by assuming that each communication link while transmitting some data will incur a cost and keep it to itself. A strategy proof multicast mechanism without using VCG mechanism is being designed so that each agent gets maximum profit when it reports its cost truthfully. This strategy proof mechanism is designed only for tree structured leaving the mesh based structure for future work. In this paper only tree based structure is studied for multicast coming to practicality mesh based structures are more needed for wireless network in order to improve fault tolerance and extensive simulations were conducted in order to study the practical performance of proposed mechanism. In this paper construction of tree and calculation of payment are done in a centralized way lacking the distributed approach.

Although efficient multicast mechanism has been proposed but there are some challenges which are unsolved and are left for further research works.

METHODS:

Assumptions are made that each node has finite local memory area and acts as a data provider for several data items and a data user. Each node maintains a copy of data items, and maintains these copies in local memory space. These copies are also called as replicas. There are 3 types of behavioral cases for nodes from the view point of selfish replica allocation [6] they are as follows:

- Non-selfish nodes: Non-selfish nodes holds the replicated data within the limit of its memory that are allocated by other nodes
- Fully-selfish nodes: Fully-selfish nodes do not hold the copies of other nodes i.e. replica of other node which are allocated, but for their accessibility, allocates replicas to other nodes.
- Partially-selfish nodes: Partially-selfish nodes use their memory space for allocating replicas by other nodes partially. The memory space is divided logically into 2 parts in this context; they are selfish and public area.

This paper discusses about only the first two types i.e. about non selfish node and fully selfish node. Several knowledge replication techniques are proposed to attenuate performance degradation. Most of the techniques assume that every mobile node cooperate fully in terms of sharing their memory house. In general, replication improves knowledge accessibility and at the same time cut back question delay, i.e., question interval, if the mobile nodes have enough memory space for replicated data and original data. A node sometimes could act egotistically, since every node has resource constraints, like battery and storage. In the existing system the selfish nodes are not detected properly and the time delay is more

Proposed Strategy:

The proposed strategy consists of three parts 1) path finding, 2) detecting selfish nodes and 3) replica allocation. Each node does the following task given below

1) Each node calculates credit risk value and based on it selfish nodes are detected. 2) Each node makes its own topology graph and based on it builds SCF-tree by excluding selfish nodes. 3) After building SCF-tree, based on it nodes allocate replica in a distributed manner. The credit risk value is calculated by each and every node to which it is connected in order to measure the degree of selfish node.

A self-serving node may not share its memory resource to store replicated data for the benefit of other node, this self-serving node is said to be selfish. This type of problem is referred to as selfish replica allocation and

this type of situation can be found in typical peer-to-peer application. The problem of selfishness leads to performance degradation and decrease in data accessibility. Selfish replica allocation specifies to a node's non cooperative behavior meaning that the node refuses to coordinate fully in sharing its memory area with other nodes. This paper discusses a selfish node detection algorithm and replica allocation techniques using self-centered friendship tree (SCF-tree) to handle the selfish replica allocation. Each node calculates credit risk value on the other node that are connected to it in order to measure the degree of selfishness.

Algorithm:

1. Initialize :
   {
   a. Initialize node memory(in packet)
   b. Initialize the update timer
   }
2. Send request for allocating replica
3. If (node is selfish)
   a. Give wrong reply or no response
4. Else
   {
   Send correct reply
   }
5. If (reply is received by originator)
   {
   a. Process the reply
   b. And make the route
   }
6. If (timer is triggered)
   a. Send the query for checking
7. If the node is corresponding node then sends the query reply
8. Originator checks the query reply
   a. And forms the SCF tree
   b. And allocates priority and replica according to the tree

Architecture:

In "Figure 2", the architecture diagram of the project is shown where the routing manager is the heart of the diagram as all the controlling and managing is done by this unit.

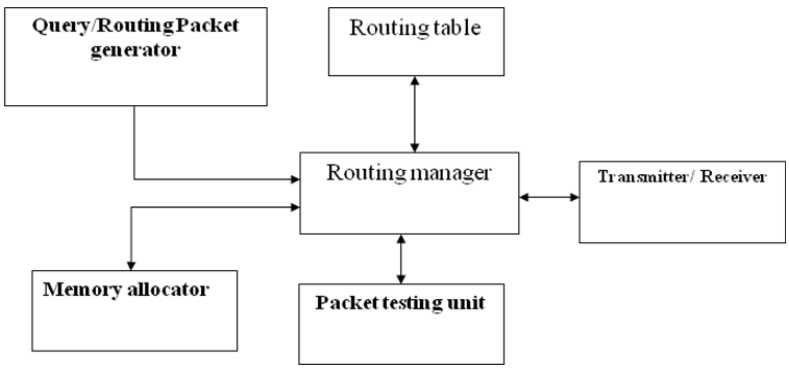

Figure 2. Block Diagram of Architecture

Routing packet generator creates the routing packet and broadcast it to all the nodes. When the nodes receive routing packet, each node allocates memory to source with the help of memory allocator and normal transmission takes place. Whenever source gets doubt about a particular node that it is acting selfishly it sends query packet to that node with the help of query packet generator. The particular node to which this query is transmitted replies back to source. The purpose of Packet testing unit is to check the replies coming from the nodes in order to validate whether the reply is correct or not, if the reply is not satisfactory it will fix the node as selfish node. Routing manager controls and manages the whole block.

Project Description:

As earlier said, this paper discusses three main parts firstly finding path secondly detecting selfish node and thirdly allocating replica.

A. Finding path

Initially when source wants to communicate to destination it broadcasts route request packet RREQ and all the nodes collect information about the neighboring nodes and allocate some memory for storing information. The destination sends route reply packet RREP to source by finding best path using SCF-tree and the path will be stored in the source routing table.

B. Detecting selfish node

Source generates query in order to check whether any node is acting selfishly. The destination node replies back to query giving the memory allocated information. The source checks the query reply and compares with the previous result. If the results are not satisfactory that particular node is fixed as selfish node and other path is chosen.

C. Allocating replica

Replica allocation is done with the help of SCF tree. Every node allocates replica at every remotion period. Each node asks its neighboring node to hold replica when it is not having enough space to hold *replica* in its local memory. As each and every node maintains its own SCF-tree, it accomplishes replication at its discretion without communicating with other nodes.

RESULTS AND DISCUSSIONS

Simulation of the project is done using Network Simulator 2 (NS2) tool taking into consideration 17 nodes and the selfish nodes are specified at run time. The results show that delay is reduced and packet loss is minimized. Table 1. Shows the different simulation parameters used in the project.

TABLE 1 Parameters used in simulation

Parameters	Values
Simulation time	10 sec
No. of nodes	17
Packet size	500
Network area	1600 m
Routing protocol	AODV
Transport protocol	UDP

Performance is one major metric and here it is evaluated by using packet delivery function. Packet delivery function is the ratio of number of packets delivered to the destination. In the project selfishness alert is added so that delay can be reduced and quality can be increased. The graph below compares the packet delivery function of the existing and proposed model. In “Figure 3”, the pdf of existing model is shown and in “Figure 4”, the pdf function of proposed model is shown. The results show that the proposed packet delivery function yields better results than existing model. Packet loss and unnecessary delay can be reduced whenever selfish ness is detected and quick action is taken.

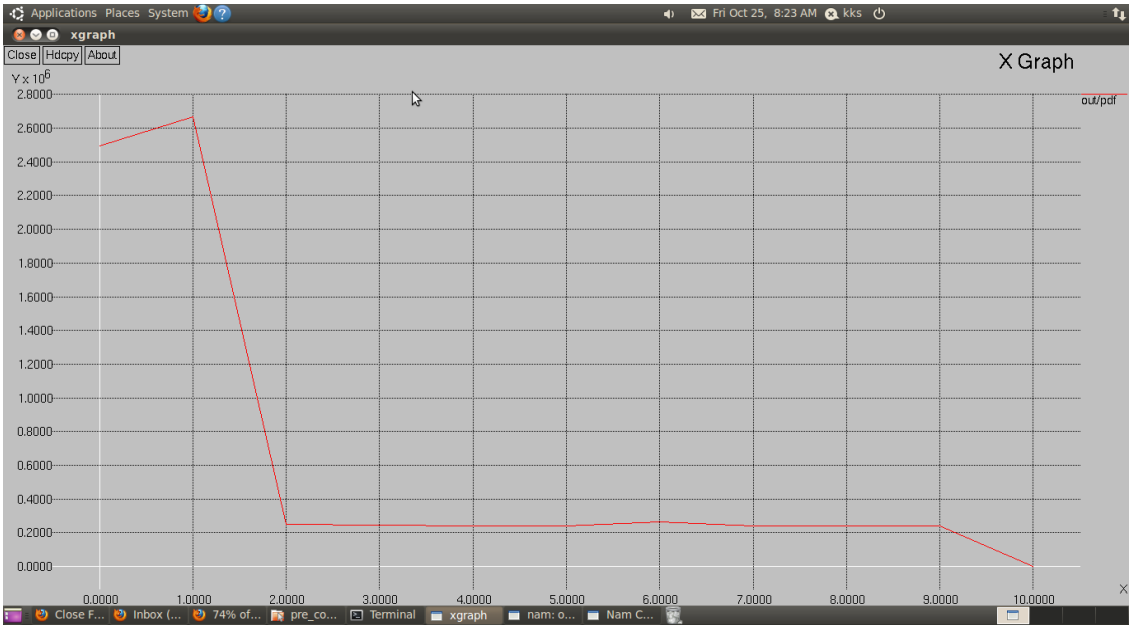

Figure 3. Packet Delivery Function of existing model

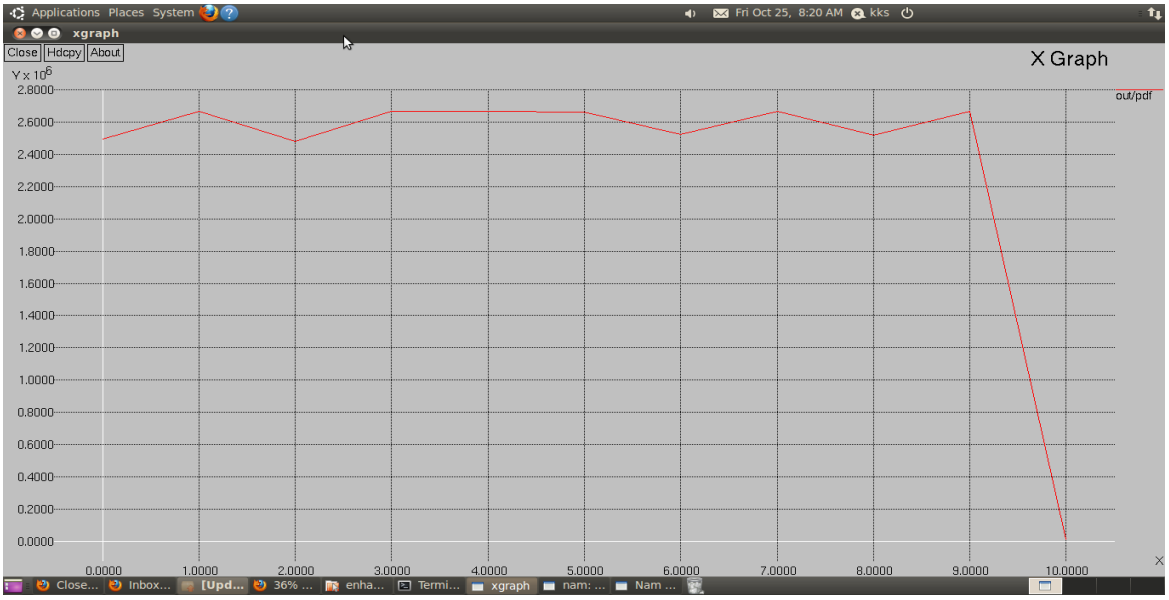

Figure 4. Packet Delivery Function of proposed model

CONCLUSION

The area of mobile ad hoc networking has been an active research area over past few years due to its widespread applications. Such a network requires that all the nodes cooperate but here some nodes act selfishly these nodes are termed as selfish nodes. In this paper a selfish node identification method and novel replica allocation technique is proposed in order to increase packet delivery and decrease delay.

REFERENCES

[1]. L Anderegg, S. Eidenbenz, “Ad hoc-VCG: A Truthful and Cost-Efficient Routing Protocol for Mobile Ad hoc Networks with Selfish Agents,” proceedings, 9th annual international conference on Mobile Computing and Networking, New York USA ,pp. 245-259,September 2003

[2]. E. Atsan, "A Scalable and Reactive Replication Framework for Mobile Ad-hoc Networks," Koc University, September 2007

[3]. K. Balakrishnan, J. Deng, P.K. Varshney, "TWOACK: Preventing Selfishness in Mobile Ad Hoc Networks," Wireless Communications and Networking Conference IEEE, New Orleans USA, pp. 2137-2142, 2005

[4]. J. Broch, D.A. Maltz, D.B. Johnson, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," proceedings, 4th annual ACM/IEEE international conference on mobile computing and networking, Texas USA, pp. 516-521, 1998

[5]. G. Ding, B. Bhargava, "Peer-to-Peer File-sharing over Mobile Ad hoc Network, proceeding," 2nd IEEE annual conference on pervasive computing and communication workshops, West Lafayette, USA, pp. 104-108, 2004

[6]. S.R. Divya, K.R. Ciji, "Evaluation of Overall Selfishness and Data Accessibility in Replica Allocation over MANET" International Journal of Engineering Research & Technology, vol 2, March 2013

[7]. T. Hara, "Effective Replica Allocation in Ad Hoc Networks for Improving Data Accessibility, "proceedings, 20th annual joint conference of IEEE computer and communications, Alaska, pp. 1568-1576, 2001

[8]. M. Li, W. Lee, A. Sivasubramaniam, "Efficient Peer-to- Peer Information Sharing over Mobile Ad Hoc Networks," proceedings, Second Workshop on Emerging Applications for Wireless and Mobile Access (MobEA II), in conjunction with the World Wide Web Conference (WWW), Pennsylvania, 2004

[9]. H. Miranda, L. Rodrigues, "Friends and Foes: Preventing Selfishness in Open Mobile Ad Hoc Networks" proceedings, 23rd International conference on Mobile Distributed Computing, Rhode island USA, pp. 440-445,2003

[10]. P. Padmanabhan, L. Grunewald, A. Vallur, M. Atiquzzamen, "A survey of data replication techniques for mobile ad hoc network databases", The VLDB Journal Springer, pp. 1143-1164, 2008

[11]. V. Srinivasan, P. Nuggehalli, C.F. Chiasserini, R.R. Rao, "Cooperation in Wireless Ad Hoc Networks," proceedings,22nd annual joint conference of the IEEE computer and communications,San Francisco, pp. 808-817, 2003

[12]. W. Wang, Xiang-Yang Li, Yu Wang, "Truthful Multicast in Selfish Wireless Networks," proceedings,10th annual international conference on Mobile Computing and Networking, New York USA , pp. 245-259, September 2004

[13]. Y. Zhang, L. Yin, J. Zhao, G. Cao, "Balancing the Tradeoffs between Query Delay and Data Availability in MANETs," IEEE Journal, vol 23, 2012