ISSN:2348-2079

Volume-6 Issue-2

International Journal of Intellectual Advancements and Research in Engineering Computations

Accessing the Location Based Details

[1]Mr. S. Jagadeesan, M.Sc., M.C.A., M.Phil., ME., Assistant Professor/MCA
[2]Ms. K. Nandhini, Final MCA
Department of MCA, Nandha Engineering College (Autonomous), Erode-52.
E-Mail ID: jagadeesan12398@gmail.com, nandhukrish20@gmail.com

***Abstract*—In this paper is solving the severe issues of current Internet architecture, such as complex usage, poor resource utilization, inefficient mapping, scalability, location dependency, for Named data wireless network. Communication in NDN is based on content names decoupling from their locations. In existing system on-demand protocol called OEFS for NDN-based mobile ad hoc networks is implementing. The OEFS results outperforms other conventional schemes in terms of average content download time, the number of Interest retransmissions, the duplication of Data packets, and the total number of sInterest propagations in the network. In reality, however, some nodes may selfishly decide only to cooperate partially, or not at all, with other nodes. These selfish nodes could then reduce the overall data accessibility in the network. This project examines the impact of selfish nodes in a mobile ad hoc network from the perspective of replica allocation. This is termed as selfish replica allocation. In particular, a selfish node detection algorithm is developed that considers partial selfishness and novel replica allocation techniques to properly cope with selfish replica allocation.**

***Index Terms*— location, selfish node, replica.**

I. INTRODUCTION

THISdocumentDelay Tolerant Network (DTN) is an intermittently connected wireless ad-hoc network that allows the communication between the wireless nodes in the state where end to end connectivity can never be achievable or when the delay associated in relaying data could be very high. The DTN is also known as opportunistic networks. The DTN's are suitable for work in the infrastructure-less environment. It also allows the communication between wireless nodes in the heterogeneous environments as well. The selection of routing protocol in DTN depends on the application environment in which it is to be used. In this study, they describe an attempt to review the existing DTN routing protocols in the literature. Also made a comparison of the existing routing protocols of DTN for the different performance metrics such as number Of data message generated, message delivery ratio and average delay.

In traditional networks, the routing of packets between a pair of nodes aims to select the optimal path with minimum cost incurred. In these networks an optimal route needs to be established before the actual transmission of message. As in DTN the end to end route can never be achieved so hop byhop, in which the selection of next hop is done dynamically as per the application scenario as well as the algorithm used. In general, when a node receives any bundle (or message then as per the algorithm, that node will search the good relay node to which it can forward the bundle. The transmission of message in DTN can either be done by replicating the message or forwarding it, that depends on the type of algorithm used. In the protocols are categorized into two broad categories as.

- Flooding Based
- Forwarding Based

II. RELATED WORK

S. Sangeetha and B. Ananthi [1] describe the problem and content searching in mobile disconnected networks has been well studied in previous research articles and the authors discussed many approaches towards the problem of content search in mobile disconnected networks. Still the approaches suffer with the problem of latency and poor searching quality. To overcome the issue of content search in MDN, the author proposes a region based random deployment approach to improve the search quality. The propose method splits the entire network region into different region and maintains Meta data about the content search. Also, the method monitors the query and computes content popularity from the search query. Based on the content popularity, the method analyses the query and the location of the client nodes. Using all these, the method performs random deployment of cache nodes to support content searching in MDN with reduced latency and improved throughput

Copyrights © International Journal of Intellectual Advancements and Research in Engineering Computations, www.ijiarec.com

Jivodar B. Tchakarov, Nitin H. Vaidya [2] describe the advances in wireless networking have enabled new paradigms in computing. An abundance of information and services provided by remote servers is expected to become available to wireless users. A fundamental issue in this environment is efficiently locating needed content. Such content may be in the form of files, services, or any other kind of data. In this paper, they describe an algorithm for efficient content location in location-aware ad hoc networks.The Geography-based Content Location Protocol (GCLP) makes use of physical location information to lower proactive traffic while reducing query cost

Cheng-Chang Hoh and Ren-Hung Hwang [3] describe a dynamic nature of MANET causes many challenges in designing robust and scalable P2P system. Although flooding-based techniques are shown to be robust in highly dynamic network, it leads to poor efficiency in terms of bandwidth usage and scalability. In this paper, they propose an efficient and scalable P2P file sharing system based on Swarm Intelligence for MANET, referred to as P2PSI. By applying the behavior of the real ant colonies, P2PSI owns the capability of adaptive learning and is able to cope with mobility problem without flooding

Thomas Repantis and VanaKalogeraki [4] propose adaptive content-driven routing and data dissemination algorithms for intelligently routing search queries in a peer-to-peer network that supports mobile users. In their mechanism nodes build content synopses of their data and adaptively disseminate them to the most appropriate nodes. Based on the content synopses, a routing mechanism is being built to forward the queries to those nodes that have a high probability of providing the desired results. Their simulation results show that their approach is highly scalable and significantly improves resources usage by saving both bandwidth and processing power.

Ying Huang, Yan Gao, KlaraNahrstedt and Wenbo He [5] describe a WiFi-based content distribution community infrastructure (CDCI), file servers are deployed in diverse locations around cities, caching popular files interesting to a community. They serve file download requests from mobile users in proximity via WiFi. In this paper, they study the optimal caching strategy for file servers in CDCI so that file retrieval probability within deadline is maximized, subject to storage constraint of each file server. The optimal caching strategy depends on many factors, such as users' mobility patterns, access point topology, file popularity, etc. They have formalizedthis content management problem as a mixed integer optimization problem. Because of NP-hardness and a large variable space of this optimization problem, they propose a heuristic algorithm MobaSsign to allocate file blocks

In addition, reducing traffic on the outgoing link will save bandwidth for other downloads which must go through Internet. Finally, workload at content origins reduces by caching near clients. WiFi-based content distribution community relies on the fact that users within a community share similar interests. If no two users request for the same file, there is no benefit of caching filesat file servers. Fortunately, in a community, it is the usual case that many users are interested in a small set of contents, such as traffic reports and community videos.

III SYSTEM METHODOLOGY

A. Introduction

The strategy existing system does not consists of three parts:

- Detecting selfish nodes,
- Allocating replica. At a specific period, or relocation period, each node executes the following procedures:
- Each node detects the selfish nodes based on credit risk scores.
- Each nodemakes its own (partial) topology graph and builds its own SCF-tree by excluding selfish nodes.
- Based on SCF-tree, each node allocates replica in a fully distributed manner.

The existing system has following disadvantages.

- The existing system is not working on the impact of different mobility patterns.
- Does not identify and handle false alarms in selfish replica allocation.
- Query delay is not reduced.

B. Replication Model

The proposed system devises novel replica allocation techniques with the developed selfish node detection method. They are based on the concept of a self-centered friendship tree (SCF-tree) and its variation to achieve high data accessibility with low-communication cost in the presence of selfish nodes. The SCF-tree is inspired by our human friendship management in the real world.

In the real world, a friendship, which is a form of social bond, is made individually. For example, although A and B are friends, the friends of A are not always the same as the friends of B. With the help of SCF-tree, we aim to reduce the communication cost, while still achieving good data accessibility. The technical contributions can be summarized as follows:

Copyrights © International Journal of Intellectual Advancements and Research in Engineering Computations, www.ijiarec.com

- Recognizing the selfish replica allocation problem: The existing system views a selfish node in a MANET from the perspective of data replication, and recognizes that selfish replica allocation can lead to degraded data accessibility in a MANET.

- Detecting the fully or the partially selfish nodes effectively: It devises a selfish node detection method that can measure the degree of selfishness.

- Allocating replica effectively: It proposes a set of replica allocation techniques that uses the self-centered friendship tree to reduce communication cost, while achieving good data accessibility.

- Verifying the proposed strategy: The simulation results verify the efficacy of the proposed strategy.

The following advantages are present in the proposed system.

- Identifies and handles false alarms in selfish replica allocation.
- Peers are classified as high-capability non-selfish peers that handle search or routing, and ordinary peers that act as their clients.
- Time threshold is given to treat the node as stable node.
- Rating the super peer nodes is possible. This helps to add more capability to that node.
- Multimedia content retrieval is efficient and data availability is more.
- Most offline peers can be tracked easily.

C. Access Frequency of Nodes

In this module, to find selfish replica allocation, first access frequency of nodes is given as input. The example data is given below.

Data	Nodes					
	N_1	N_2	N_3	N_4	N_5	N_6
D_1	0.65	0.25	0.17	0.22	0.31	0.24
D_2	0.44	0.62	0.41	0.40	0.42	0.46
D_3	0.35	0.44	0.50	0.25	0.45	0.37
D_4	0.31	0.15	0.10	0.60	0.09	0.10
D_5	0.51	0.41	0.43	0.38	0.71	0.20
D_6	0.08	0.07	0.05	0.15	0.20	0.62
D_7	0.38	0.32	0.37	0.33	0.40	0.32
D_8	0.22	0.33	0.21	0.23	0.24	0.17
D_9	0.18	0.16	0.19	0.17	0.24	0.21
D_{10}	0.09	0.08	0.06	0.11	0.12	0.09

Table 1: Access Frequency Of Nodes

D. Selfish Replica Allocation

In this module, selfish replication allocation is shown. Fig. 1 illustrates an existing replica allocation scheme, DCG, where nodes N1, N2, ... ,N6 maintain their memory space M1, M2, ... ,M6, respectively, with the access frequency information in Table 1 (In Fig. 1, a straight line denotes a wireless link, a gray rectangle denotes an original data item, and a white rectangle denotes a replica allocated.

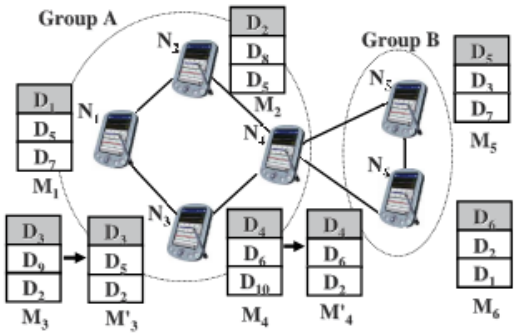

Figure 1: Selfish Replica Allocation

In Table 1, the gray colored area shows three data items that are accessed frequently by N3 and N4). As shown in Fig. 1, DCG seeks to minimize the duplication of data items in a group to achieve high data accessibility. Let us consider the case where N3 behaves "selfishly" by maintaining M'3, instead of M3, to prefer the locally frequently accessed data for low query delay. In the original case, D3, D9, and D2 were allocated to N3.

However, due to the selfish behavior, D3, D5, and D2, the top three most locally frequently accessed items, are instead maintained in local storage. Thus, other nodes in the same group, i.e., N1, N2, and N4, are no longer able to access D9. This showcases degraded data accessibility, since N1, N2, and N4 cannot fully leverage N3's memory space as intended in cooperative replica sharing.

As another example, a node may be only "partially selfish" in a MANET. For instance, node N4 may want to locally hold D2, one of the locally frequently accessed data items. In this case, N4 uses only a part of its storage for its own frequently accessed data, while the remaining part is for the benefit of overall data accessibility. Thus, N4 may decide to maintain M'4, instead of M4. Even with only partial selfishness, data accessibility is still degraded, since the other nodes in the same group, i.e., N1, N2, and N3, cannot access D10.

E. Algorithm: Detect Selfish Nodes

In this module, the algorithm to detect selfish nodes is executed. Algorithm 1 describes how to detect selfish nodes. At each relocation period, node N_i

Copyrights © International Journal of Intellectual Advancements and Research in Engineering Computations, www.ijiarec.com

detects selfish nodes based on nCR^k_i. Each node may have its own initial value of Pk_ias a system parameter. Interestingly, the initial value of Pk_i can represent the basic attitude toward strangers.
For instance, if the initial value equals zero, node N_i always treats a new node as a nonselfish node. Therefore, Ni can cooperate with strangers easily for cooperative replica sharing. Replicas of data items are allocated by allocation techniques. After replica allocation, N_i sets ND^k_i and SS^k_i accordingly. Recall that both ND^k_i and SS^ki are estimated values, not accurate ones. The estimated values are adjusted at query processing time, according to Algorithm 2.

Algorithm 1

00: At every relocation period
01: /* N_i detects selfish nodes with this algorithm */
02: detection(){
03: for (each connected node N_k){
04: if ($ŋCR_i^k$<δN_k is marked as non-selfish;
05: else N_k is marked as selfish;}
06: wait until replica allocation is done;
07: for (each connected node N_k){
08: if (N_i has allocated replica to N_k){
09: ND_i^k= the number of allocated replica;
10: SS_i^k = the total size of allocated replica;}
11: else{
12: ND_i^k = 1;
13: SS_i^k= the size of a data item;
14: } } }

F. Algorithm: Update Selfish Features

In this module, the algorithm to update selfish features is executed. In Algorithm 2, N_i maintains its ND^k_i, SS^k_i, and P^k_iduring each query processing phase. When N_i issues a query, Ni awaits the response from the expected node N_k during the predefined wait time w, where w is the expected maximum time taken to exchange one round of request-response message across the entire network. Whenever Ni detects the selfish behavior of N_k, it modifies P^k_i, ND^k_i, and SS^k_iaccordingly.

If N^k serves the query as expected, however, only P^k_iwill be decreased, while ND^k_iandSS^k_iremainunchanged.Note that, in case an unexpected node $N_j$replies to N_i's request, N_i will modify ND^j_iandSS^j_iaccordingly, while not affecting P^j_i, P^k_i, ND^k_i, and SS^k_i. That is, the reply from unexpected nodes does not affect the selfish features of expected nodes. Note also that N_i may receive multiple replies from unexpected and/or expected nodes. In this case, N_i modifies P^k_i, ND^k_i, and/or SS^k_iaccordingly for each reply based on Algorithm 2. If N_i does not receive any reply from expected node N_k during w, it observes N_k's selfish behavior andmodifiesP^k_i,ND^k_i, and SS^k_iaccordingly.

Algorithm 2

00: At every query processing time
01: /* When Ni issues a query */
02: update_SF(){
03: while (during the predefined time ω){
04: if (an expected node N_k serves the query)
05: decrease P_i^k;
06: if (an unexpected node N_j serves the query){
07: ND_i^j =ND_i^j + 1;
08: SS_i^j= SS_i^j+ (the size of a data item);
09: } }
10: if (an expected node N_k does not serve the query){
11: increase P_i^k;
12: ND_i^k= ND_i^k - 1;
13: SS_i^k = SS_i^k - (the size of a data item);
14: } }

G. Flooding Based Method

This module shows the basic structure of the proposed broadcasting format. Each node schedules a broadcast. Clearly, each message is broadcast once at most by a node. A broadcast schedule can be set at any time. For example, a message can be dropped after the first reception but scheduled for broadcast the second time. The node will broadcast the data to maximum number of nodes nearest to it.

H. Advertisement Based Method

1. Super Node And Normal Node Addition

In this module, the node details in the P2P network are added. The node id, computer name and IP address details are saved in 'Nodes' table. The super node details in the P2P network are added. The node id is selected from 'Nodes' table and are updated as super peer type.

2. Assign Super Node As Parent To Normal Node

In this module, the network is constructed such that 'n' number of nodes is interconnected; among which super peer nodes become parent to normal nodes. So the environment resembles the real P2P environment.

3. Update Document Details To Super Nodes

In this module, the node id is selected from 'Nodes' table and file is selected, the search phrase (part of the query) to show this file (when the phrase is given as search text) and uploaded in the shared path of that node. The details are saved in 'Files' table. In this module, all the nodes update the document/resource list and search phrase one or more super peer nodes to which it is connected.

4. Normal Node Send Query To Super Node

In this module, the nodes creates a query and sends to all the super peer nodes to which it connects and retrieve results in which the peers connected with that super peer and having the required resource.

Copyrights © International Journal of Intellectual Advancements and Research in Engineering Computations, www.ijiarec.com

5. Resource Download From Selected Nodes

In this module, the node selects the node which is having the required resource and downloads the content.

I. Social Networking Based Method

1. Intracommunity File Searching and Retrieval

Algorithm 1 (Section 4.3.3 A) shows the pseudocode of the intracommunity searching algorithm which describes intracommunity file searching for query Q conducted by node Ni.

2. Intercommunity File Searching and Retrieval

Algorithm 2 (Section 4.3.3 B) shows the pseudocode of the intercommunity searching algorithm.

IV CONCLUSION

The new system eliminates the difficulties in the existing system. It is developed in a user-friendly manner. The aim of the project is to generate data templates with minimum input. The system is very fast and any transaction can be viewed or retaken at any level. Error messages are given at each level of input of individual stages.

This software is very particular in developing web templates. More number of customized templates can be designed. In addition, tags can also be created. Any node with .Net framework installed can execute the application.

The difficulty in distributing the content in the server is eliminated by using this application. It reduces the server bandwidth to consistent amount. The end users need not wait for server in downloading the content since the P2P application gets the content from available clients.

- A good documentation of user-friendly features had been incorporated in the system.
- The system has been introduced to eliminate human error.
- To minimize the time consumption and design & development work.

V SCOPE FOR FUTURE DEVELOPMENT

It is believed that almost all the system objectives that have been planned at the commencement of the software development have been net with and the implementation process of the project is completed. A trial run of the system has been made and is giving good results the procedures for processing is simple and regular order.

The process of preparing plans had been a new experience, which was found use full in later phases of the project is completed. Efforts had been taken to make the system user friendly and as simple as possible. However at some points some features may have been missed out which might be considered for further modification of the application. The new system become useful if the below enhancements are made in future.

- The statistical analysis of download data if prepared, can be used for further development.
- The P2P application if developed as web site can be used from anywhere.
- The user portal if developed can assist in maintaining download history for end users.

The new system is designed such that those enhancements can be integrated with current modules easily with less integration work

REFERENCES

[1] Y. Huang, Y. Gao, K. Nahrstedt, and W. He, "Optimizing File Retrieval in Delay-Tolerant Content Distribution Community," Proc. IEEE 29th Int'l Conf. Distributed Computing Systems (ICDCS '09), 2009.

[2] J. Reich and A. Chaintreau, "The Age of Impatience: Optimal Replication Schemes for Opportunistic Networks," Proc. Fifth Int'l Conf. Emerging Networking Experiments and Technologies (CoNEXT '09), 2009.

[3] V. Lenders, M. May, G. Karlsson, and C. Wacha, "Wireless Ad Hoc Podcasting," ACM SIGMOBILE Mobile Computing and Comm. Rev., vol. 12, pp. 65-67, 2008.

[4] F. Li and J. Wu, "MOPS: Providing Content-Based Service in Disruption-Tolerant Networks," Proc. IEEE 29th Int'l Conf. Distributed Computing Systems (ICDCS '09), 2009.

[5] C. Boldrini, M. Conti, and A. Passarella, "ContentPlace: Social-Aware Data Dissemination in Opportunistic Networks," Proc. 11th Int'l Symp. Modeling, Analysis and Simulation Wireless and Mobile Systems (MSWiM '08), 2008.

[6] M. Papadopouli and H. Schulzrinne, "A Performance Analysis of 7DS: A Peer-to-Peer Data Dissemination and Prefetching Tool for Mobile Users," Proc. IEEE Sarnoff Symp.

Copyrights © International Journal of Intellectual Advancements and Research in Engineering Computations, www.ijiarec.com

Digest Advances in Wired and Wireless Comm., 2001.

[7] J.B. Tchakarov and N.H. Vaidya, "Efficient Content Location in Wireless Ad Hoc Networks," Proc. IEEE Int'l Conf. Mobile Data Management (MDM '04), 2004.

[8] A. Fast, D. Jensen, and B.N. Levine, "Creating Social Networks to Improve Peer-to-Peer Networking," Proc. 11th ACM SIGKDD Int'l Conf. Knowledge Discovery in Data Mining (KDD '05), 2005.

[9] A. Iamnitchi, M. Ripeanu, and I.T. Foster, "Small-World File-Sharing Communities," Proc. IEEE INFOCOM, 2004.

[10] M. Mcpherson, "Birds of a Feather: Homophily in Social Networks," Ann. Rev. Sociology, vol. 27, no. 1, pp. 415-444 , 2001.

[11] J. Eriksson, H. Balakrishnan, and S. Madden, "Cabernet: vehicular content delivery using wifi," in MobiCom '08: Proceedings of the 14th ACM international conference on Mobile computing and networking, 2008.

[12] D. T. J. Kangasharju, K.W. Ross, "Optimizing file availability in peer-to-peer content distribution," in INFOCOM 2007, 26th IEEE International Conference on Computer Communications, 2007.

[13] TIER Project, http://tier.cs.berkeley.edu/wiki/Home.

[14] RSS Advisory Board. RSS 2.0 Specification. http://www.rssboard.org/rss-specification, June 2007.

[15] M. Nottingham and R. Sayre (Eds.). The Atom Syndication Protocol. IETF RFC 4287, December 2005.

Copyrights © International Journal of Intellectual Advancements and Research in Engineering Computations, www.ijiarec.com