An Optimized Cache Coherence handling Scheme on DSR Routing Protocol for (MANETS)

Ms. Samiksha Suri

Abstract— **Mobile ad hoc networks (MANETS) are self-created and self organized by a collection of mobile nodes, interconnected by multi-hop wireless paths in a strictly peer to peer fashion. DSR (Dynamic Source Routing) is an on-demand routing protocol for wireless ad hoc networks that floods route requests when the route is needed. Route caches in intermediate mobile node on DSR are used to reduce flooding of route requests. But with the increase in network size, node mobility and local cache of every mobile node cached route quickly become stale or inefficient. In this paper, for efficient searching, we have proposed a generic searching algorithm on associative cache memory organization to faster searching single/multiple paths for destination if exist in intermediate mobile node cache with a complexity (Where n is number of bits required to represent the searched field). The other major problem of DSR is that the route maintenance mechanism does not locally repair a broken link and Stale cache information could also result in consistencies during the route discovery /reconstruction phase. So to deal this, we have proposed an optimized cache coherence handling scheme for on -demand routing protocol (DSR).**

***Index Terms*—About four key words or phrases in alphabetical order, separated by commas.**

I. Introduction

Mobile ad hoc networks (MANETS) are self-created and self organized by a collection of mobile nodes, interconnected by multi-hop wireless paths in a strictly peer to peer fashion [1]. Caching is an important part of any on-demand routing protocol for wireless ad hoc networks. In mobile ad hoc network (MANETS) [2],[3],[4] all node cooperate in order to dynamically establish and maintain routing in the network , forwarding packets for each other to allow communication between nodes not directly within wireless transmission range. Rather than using the periodic or background exchange of routing information common in most routing protocols , an on-demand routing protocols is one that searches for the attempts to discover a route to some destination node only when a sending node originates a data packet addressed to the node. In order to avoid the need for such a route discovery to be performed before each data is sent, an on demand routing protocol must cache routes previously discovered. Such caching then introduces the problem of proper strategies for managing the structure and contents of this cache, as nodes in the network move in and out of wireless transmission range of one another, possibly invalidating some cached routing information. Several routing protocols for wireless ad hoc networks have used on-demand mechanisms, including temporally ordered routing algorithm (TORA) [8], Dynamic source Routing protocols (DSR) [5]. Ad hoc on demand distance vector (AODV) [6], Zone routing protocol (ZRP) [7], and Location-Aided Routing (LAR) [9]. For example, in the Dynamic Source Routing protocol [5] in the simplest form, when some node S originates a data packet destined for a node D to which S does not currently know a route, Initiates a new route discovery by beginning a flood a request reaches either D or another node that has a cached route to D, cache with a complexity) (n O this node then returns to S the route (Where n is number of bits required discovered by this request. to represent the searched field).The Performing such a route discovery other major problem of DSR is that can be an expensive operation, since the route maintenance mechanism it may cause a large number of does not locally repair a broken link request packets to be transmitted, and Stale cache information could and also add latency to the also result in inconsistencies during subsequent delivery of data packet the route discovery/reconstruction that initiated it. But this route phase. So to deal this, we have discovery may also result in the proposed an optimized cache of a large amount of information about the current state of network that may be useful in future routing decision. In particular, S may receive a number of route proposed an optimized cache coherence handling scheme for on -demand routing protocol (DSR). Inthis paper, Section 2, we describes the Dynamic Source Routing Protocol (DSR), Section 3, we replies in response to its route describe related work, Section 4, we discovery flood, each of which returns information about a route to D through a different portion of the network. In high-mobility environment the performance degrades rapidly of this protocol because the route maintenance mechanism does not locally repair a broken link. Stale cache information could also result in inconsistencies during the route reconstruction

phase. In this paper, for efficient searching, we have proposed a generic searching algorithm on associative cache memory organization to faster searching single/multiple paths for destination if exist in intermediate

Manuscript received May 07, 2013.

Ms. Samiksha Suri, Lecturer, Department of Computer Application, Udhampur, J & K, India.

mobile node describe associative searching Flowchart, Algorithm and their implementation with example, Section 5, we describe proposed an optimized cache handling scheme and Section 6,we had concluded the paper and future works.

II. Overview of the Dynamic Source Routing Protocols (DSR)

Dynamic source routing protocol (DSR) is an on-demand protocol designed to restrict the bandwidth consumed by control packets in ad hoc wireless networks by eliminating the periodic table-update messages required in the table-driven approach [10]. The major difference between this and other on-demand routing protocols is that it is beaconless and hence does not require periodic hello packet (beacon) transmission, which are used by a node to inform its neighbors of its presence. The basic approach of this protocol (and all other on-demand routing protocols) during the route construction phase is to establish a route by flooding Route Request packets in the network. The destination node, on receiving a Route Request packet, responds by sending a Route Reply packet back to the source, which carries the route traversed by the Route Request packet received. Consider a Source node that does not have a route to the destination. When it has data packets to be sent to that destination, it initiates a Route Request packet. This Route Request is flooded throughout the network. Each node, upon receiving a Route Request packet, rebroadcasts the packet to its neighbors if it has not forwarded already or if the node is not the destination node, provided the packets time to live (TTL) counter has not exceeded. Each Route Request carries a sequence number generated by the source node and the path it has traversed. A node, upon receiving a Route Request packet, checks the sequence number n the packet before forwarding it. The packet is forwarded only if it is not a duplicate Route Request. The Sequence number on the packet is used to prevent loop formations and to avoid multiple transmissions of same Route Request by an intermediate node that receives it through multiple paths. Thus, all nodes except the destination forward a Route Request packet during the route construction phase. A destination node, after receiving the first Route Request packet, replies to the source node through the reverse path the Route Request packet had traversed. In Figure 1, source node 1 initiates a Route Request packet to obtain a path for destination node 15. This protocol uses a route cache that stores all possible information extracted from the source route contained in a data packet. Nodes can also learn about the neigh boring routes traversed by packets if operated in the promiscuous mode (the mode of peration in which a node can receive the packets that are neither broadcast nor addressed to itself). This route cache is also used during the route construction phase. If an intermediate node receiving a Route Request has a route to the destination node in its route cache, then it replies to the source node by sending a Route Reply with the entire route information from the source node to the destination node.

A. Optimizations:

Several optimization techniques have been incorporated into the basic DSR protocol to improve the performance of the protocol. DSR uses the route cache at intermediate nodes. The route cache is populated with routes that can be extracted from the information contained in the data packets that get forwarded.

This cache information is used by the intermediate nodes to reply to the source when they receive a Route Request packet and if they have a route to the corresponding destination. By operating in the Promiscuous mode, an intermediate node learns about route breaks. Information thus gained is used to update the route cache so that the active routes maintained in the route cache do not use such broken links. During network partitions, the effected nodes initiate Route Request packets. An exponential back off algorithm is used to avoid frequent Route Request flooding in the network when the destination is in another disjoint set. DSR also allows piggy-backing of a data packet on the Route Request so that a data packet can be sent along with the Route Request.

III. Related Works.

A. Cache data and cache path

The cache data scheme considers the cache placement policy at intermediate nodes in the routing path between the source and destination. The node caches a passing by data item locally when it finds that the data item is popular,i.e., there were many requests for data item, or it has enough free cache space. Since cache data needs extra space to save the data, it should be used prudently. A conservative rule is proposed as follow: A node does not cache the data if all requests for the data are from the same node. However, it uses cooperative caching protocol among mobile node. Each mobile node does not independently perform the caching tasks such as placement and replacement. Cache path is also proposed for redirecting the requests to the cache node. In MANETS, the network the network topology changes fast and thus, the cached path may become invalid due to the movement of mobile nodes [11].

B. Neighbor Caching Technique

The concept of neighbor caching (NC) is to utilize the cache space of inactive neighbors for caching tasks. The basic operations of NC are as follow. When a node fetches a data from remote node, it puts the data in its own caching space for reuse. This operation needs to evict the least valuable data from the cache based on a replacement algorithm. With this scheme, the data that is to be evicted is stored in the idle neighbor nodes storage. In the near future if the node needs the data again, it requests the data not from the far remote source node but from the near neighbor that keeps the copy of data. The NC scheme utilizes the available cache space of neighbor to improve the caching performance. However, it lacks the efficiency of the cooperative caching protocol among the mobile nodes [12].

C. Node caching schemes

This is a novel approach to constrain route request broadcast based on node caching. The Intuition used is that the nodes involved in recent data packet forwarding have more reliable information about its neighbors and have better locations (e.g., on the intersection of several data routes) than other MANET nodes. The nodes which are recently involved in data packet forwarding are considered as cache nodes, and only they are used to forward route requests. The modified route request uses a fixed threshold parameter H. The first route request is sent with the small threshold H. When a node N receives the route request, it compares the current time T with the time T(N) when the last data packet through N has been forwarded. If T-H > T(N), then N does not belong to the current cache and ,therefore, N will not propagate the route request. Otherwise, if T-H <= T(N), then N is in the node ache and the route request is propagated as usual[13].

D. Group Caching

There are some challenges and issue such as mobility of mobile nodes, power consumption in battery, and limited wireless bandwidth when caching techniques are employed in MANETs for data communication .Due to the movement of mobile nodes, MANETs may be partitioned into many independent networks. Hence, the requester cannot retrieve the desired data from the remote server (data source) in another network. The entire data accessibility will be reduced. Also, the caching node may be disconnected from the network for saving power. Thus, the cached data in a mobile node may not be retrieved by other mobile nodes and then usefulness of the cache is reduced. The mobile nodes also decide the caching policy according to the caching status of other mobile nodes. However, the existing cooperative caching in a MANET lack an efficient protocol among the mobile nodes to exchange their localized caching status for caching tasks.

In this work a novel cooperative caching scheme called Group caching (GC) which maintains localized caching status of 1 hop neighbors for performing the tasks of data discovery, caching placement, and caching replacement when a data request is received in a mobile node. Each mobile node and its 1 hop neighbors form a group by using the "Hello" message mechanism. In order to utilize the cache space of each mobile node and its 1 hop neighbors form a group by using space of each mobile in a group, the mobile nodes periodically send their caching status in a group. Thus, when caching placement and replacement need to be performed, node selects the appropriate group member to execute the caching task in the group; this reduces redundancy of cached data objects [14].

Another work is intelligent caching a technique in which, a node not only saves the path discovered during route discovery for itself but also for others who are located close to it.

This technique reduces the number of route request packets unnecessarily circulating in the network, when the path it requires is present in its neighbourhood.[15].

Authors of [16] in order to share internet contents among mobile users by utilizing low cost wireless connectivity, a content delivery framework with a new content perfecting strategy (AGCS).Another work in which cache management, cooperative caching increase the effective capacity of cooperative caches by minimizing Proposed an Optimized Cache Coherence Handling Scheme Ad hoc networks (MANETS) are self- created and self organized by a collection of mobile nodes, interconnected by multi-hop wireless paths in a strictly peer to peer fashion. Caching is an important part of any on-demand routing protocol for wireless ad hoc networks. In mobile ad hoc network (MANETS) all nodes cooperate in order to dynamically establish and maintain routing in the network, forwarding packets for each other to allow communication between nodes not directly within wireless transmission range. For hat Several optimization techniques have been incorporated into the basic DSR protocol to improve the performance of the protocol. DSR uses the route cache at intermediate nodes. The route cache is populated with routes that can be extracted from the information contained in the data packets that get forwarded. This cache information is used by the intermediate nodes to reply to the source when they receive a Route Request packet during Route Discovery Phase. Due to presence of private cache for each mobile node in an ad hoc network necessarily introduces problems of cache coherence, which may result in data inconsistency. Clearly, the cache coherence problem cannot be solved by a memory Write-through policy. If a Write-through policy is used, the main memory location is updated, but the possible copies of the routing information in other caches are not automatically updated by the write-through mechanism. So "Write-through: is neither necessary nor sufficient for cache coherence. For that in this paper, we have proposed a dynamic coherence check scheme for cache coherence in routing table of mobile nodes for MANETs.

In this existing scheme, called dynamic coherence check, multiple copies are allowed. However, whenever a mobile node moves and modifies routing information in its local cache, it must check the other caches to invalidate possible copies.

This operation is referred to as a cross-interrogate (XI). In other words, when a mobile node writes into a shared block X in its cache, the node sends a signal to all the remote caches to indicate that the "data at memory address X has been modified." At the same time, it writes through memory. Note that, to ensure correctness of execution, a mobile node which requests an XI must wait for an acknowledge signal from all other mobile nodes before it can complete the write operation. The XI invalidates the remote cache location corresponding to X if it exists in that cache. When the other mobile node references this invalid cache location, it results in a cache miss, which is serviced to retrieve the block containing the updated information. In this approach, for each write operation, (n – 1) XIs result, where n is the number of mobile nodes. Note that the two sources of inefficiency for this technique are the necessity of a write-through policy, which increases the network traffic, and the redundant cache XIs which are performed. In the latter case,a cache is purged blindly whether or not it contains the data item X.

In our proposed scheme, our objective is to optimize cache coherence handling scheme .In this scheme, we focus on a more refined technique filters the cross interrogate (XI) requests before they are initiated on reactive routing protocol DSR for mobile ad hoc network. For that, we have ad hoc network in which every mobile node having own local cache and there is one mobile node having the centralized shared main memory. This main memory contains the memory control element (MSC) maintains a central copy of the directories of all the caches. We will elaborate on a similar scheme called the presence flag technique, which assumes a write-back main memory update policy. There are two central tables associated with the blocks of main memory (MM) as shown in Figure5. The first table is a two-dimensional table called the Present table. In this table, each entry contains a present flag for the ith block in MM and the cth cache. If, 1] , [= c i P then the cth cache has a copy of the ith block of MM, otherwise it is zero. The second table is the Modified table and is one-dimensional. In this table, each entry M[i] contains a modified flag for the ith block of MM. If, 1] [= i M it means that there exists a cache with a copy of the ith block more recent than the corresponding copy in MM. The Present and Modified tables can be implemented in a fast random-access memory. The philosophy behind the cache coherence check is that an arbitrary number of caches can have a copy of a block, provided that all the copies are identical. They are identical if the Mobile node associated with each of the caches has not attempted to modify its copy since the copy was loaded in its cache. We refer to such a copy as read only (RO) copy. In order to modify a block copy in its cache, a mobile node must own the block copy with exclusive read only(EX) or exclusive read-write (RW) access rights. A copy is held EX in a cache if the cache is the only one with the block copy and the copy has not been modified. Similarly, a copy is held RW in a cache if the cache is the only one with the block copy and the copy has been modified. Therefore, for consistency, only one mobile node can at any time own an EX or RO copy of a block.

III. References

[1]Jeremy Pitt, Pallapa Venkatarram, and Abe Mamdani, "QoSManagement in MANETS Using Norm-Governed ESAW 2005, LNAI 3963, Page(s): 221-240,2006.

[2]Internet Engineering Task Force MANET Working Group. Mobile Ad hoc networks (Manet) Charter Available at http://www.ietf.org/html.charters/manet-charter.html.

[3]Asis Nasipuri, Mobile Adhoc networks,Department of Electrical and Computer Engineering, The university of North Carolina at charlotte.Charlotte,NC 28233-0001.

[4]C.E. Perkins, Ad hoc networking, Addison-Wesley, 2001.[5] D.Johnson, Rice University; Y. Hu,UIUC and D.Maltz,Microsoft Research The Dynamic Source Routing Protocol (DSR) for mobile ad hoc networks for IPV4, February 2007 http://www.ietf.org/rfc/rfc4728.txt.

[5]C. Perkins and S. Das, Ad hoc On Demand Distance Vector (AODV) Routing IETF, Internet Draft, draft- ietf-manetaodv-13,RFc 3561, 17,February -17- 2003.

[6]Z. Haas, M. Pearlman nad P. Smar, Zone routing protocol (ZRP), Internet Draft, Internet Engineering Task ForceJan 2001, http://www.ietf.org/internet- drafts-ietf-manetzoneierp-00.txt.

[7] S. Bradner , temporally-ordered routing algorithm (TORA) Routing IETF, Internet draft-ietf-manet-tora-spec04.txt,RFC 2026,July 2001.

[8]Y. Kuo, and N.H. Vaidya, "Location –Aided Routing (LAR) Mobile Ad Hoc Networks," In proceedings of the International Conference on Mobile Computing and Networking (MobiCom'98),Oct.1998.

[9]David .B Johnson, David. A. Maltz, and Josh Broch, "Dynamic Source Routing protocol for Multihop Wireless Ad Hoc Networks," In Ad Hoc Networking, edited by Charles E. Perkins, chapter5, pages 139-172. AddisonWesley,2001

[10] LiangZhong Yin and Guohong Cao, " Supporting Cooperative caching in ad hoc networks ," IEEE Transactions on Mobile computing, Vol. 5, Issue 1,pages 77-89,jan. 2006.

[11] Joonho Cho, Seungtaek Oh, Jaemyoung Kim, Hyeong Ho Lee, and Joonwon Lee, " Neighbor caching in multi-hop wireless ad hoc networks," IEEE Communications Letters,Vol 7,issue Nov. ,pages 525-527,2003.

[12] Sunlook Jung,Nisar Hundewale, and Alex Zelikovsky, "Node Caching Enhancement of Reactive Ad hoc routing," IEEE Wireless Communications and

[13] NetworkingConference, 2005.

[14] Yi-Wei Ting and Yeim-Kuan Chang, " A novel Cooperative Caching Scheme for Wireless Ad hoc Networks;GroupCaching," International Conference on Networking Architecture and storage ,NAS 2007.

[15] Shobha.K.R., and K. Rajanikanth "Intelligent caching in On-Demand Routing Protocol for Mobile Adhoc Networks"World Academy of Science Engineering and Technology 56 pages 413-420,2009.

[16] Yaozhou Ma, M. Rubaiyat Kibria, and Abbas Jamalipour "Cache-based Content Delivery in Opportunistic Mobile Ad hoc Networks" IEEE "GLOBECOM", 2008.

[17] Mrs. K. Shanmugavadivu and Dr. M. Madheswaran "CachingTechnique for Improving Data Retrieval Performance in Mobile Ad Hoc Networks" In International Journal of Computer Science and Information Technologies(IJCSIT),Vol. 1(4),pages 249-255,2010.

[18] Yih-Chun Hu and David B.Johnson "Ensuring Cache Freshness in On-Demand Ad hoc Network Routing Protocols" In POMC'02, October 30-31, Toulouse, France,2002.