The International Journal Of Engineering And Science (IJES)
||Volume|| 3 ||Issue|| 2 ||Pages|| 06 - 11 || 2014 ||
ISSN (e): 2319 – 1813 ISSN (p): 2319 – 1805

A Light Weight Multi-Copy Geocache Maintenance in Mobile Disconnected networks

[1,] M.Subalakshmi, [2,] Dr.M.Prabakaran,
[1,] Research Scholar of Bharathidasan University
Head of the Department, Department Of Computer Science
Sri Saradha College for Women, Perambalur, Tamilnadu, India
[2,] Research advisor,
Asst, Professor, department of computer science
Government Arts College, ariyalur-621713, Tamilnadu, India

---ABSTRACT--
We present a light weight Geocache maintenance algorithm to maintain the information at a particular location in mobile disconnected networks. In mobile disconnected networks the mobile nodes within the coverage of particular location can carry the data for little while and pass to some other node when it moves away from the location. The selection of nodes to carry the data by means of movement of the node and returning the data to the location when it moves away from location doesn't make sense. Because it increases the traffic in the network and the traffic introduced in sending the data back to the location will increase the network overhead. To challenge this issue we introduce a multi-copy Geocache maintenance algorithm, where the cache or data will be maintained in few nodes around the location. In the proposed methodology the data will not be returned to the origin but will be handover to some other node which is closer in origins perimeter. This reduces the unnecessary routing of packets and data towards the origin and removes the failure introduced by the node failure.

INDEX TERMS: *Geocache Maintenance, Mobile Disconnected Networks, Location based information.*

Date of Submission: 11 January 2014 ⟺ Date of Acceptance: 15 February 2014

I. INTRODUCTION:

The peoples started using mobile phones and PDA's in their daily life. Due to the technology development these mobile devices could be used as a sensing device. Unlike few automatic special purpose sensing machines, above said devices can sense anytime anywhere, so that the people can watch audio, video, pictures and other format data's independent of amount of data volume. These data can potentially bring great convenience to the society as they can serve as traces of our lives and logs of the physical world.

Fully utilizing these data, however, demands the establishment of channels between data producers and consumers. We have seen several methods that were used to establish such channels in earlier systems. In many web applications, data producers upload their data to servers, and consumers can either directly contact the server or locate the server through a search engine; in many peer-to-peer data sharing applications, directories are used to map data names to their locations. Though these methods have proven success in their intended systems, they are unsuitable for the anytime-anywhere personal sensing. In personal sensing, there is no fixed relationship between data producers and consumers. Data are more likely to be produced unintentionally than purposefully, and the value of the data is discovered postfacto. Consequently, we may end up having much more data than what will be needed later, and uploading these data can place a huge burden on the underlying network. In addition, privacy can be a serious concern in a server-centric solution as well. This relationship, i.e., having many more producers than consumers, is opposite from what we have observed in other systems, and thus calls for a new data sharing architecture.

For example suppose you have lost your laptop somewhere around a location in Chennai (Guindy), what we will do is we simply file a complaint in the police station and later we check for the updation in the police station. This form of query becomes location based query and to facilitate this we have designed a container where the information is stored and the information about the container is called Geocache. The Geocache are maintained around the location in few mobile nodes, so that it could be fetched easily at the time of location based query arises.

In this paper we address the following challenges

1. Maintaining the Geocache persistent even at natural disaster and temporary node disconnection and node failure.
2. Minimizing the network overhead induced by routing the Geocache towards location of interest.

II. BACKGROUND:

The area of interest covers mobile computing, sensor networks and vehicular networks. There are various methodologies have been discussed to support location based queries in vehicular networks. We discuss few of them here and analyze the techniques proposed for the support of location based queries.

GHT: A Geographic Hash Table for Data-Centric Storage [1], have been proposed, which specifies hash table mechanism to store geographic data in Data centric Storage. In this GHT hashes keys into geographic coordinates, and stores a key-value pair at the sensor node geographically nearest the hash of its key. The system replicates stored data locally to ensure persistence when nodes fail. A data object is associated with a key and each node in the system is responsible for storing a certain range of keys. A name-based routing algorithm allows any node in the system to locate the storage node for an arbitrary key. This enables nodes to put and get files based on their key, thereby supporting a hash-table-like interface and it uses the GPSR geographic routing algorithm as the underlying routing system.

In [2], an energy efficient computing for wildlife tracking is discussed with Zebranet. In this they place sensors on zebras to collect valuable zoology data. In under water sensor network [3], mobile nodes are robots that collect data from regions of interest. Several projects target specifically at vehicular sensing. CarTel [4], for example, is a comprehensive distributed mobile computing system used to collect, process, and visualize data from sensors located on mobile units. It aims at exploring in-network computing on individual mobile units, as we do, but it does not use intervehicle communication, which in our project, is a main focus to enable distributed aggregation of sensor readings from multiple cars. Another vehicular sensor network:

In vehicular delay tolerant networks Maxprop [5] is proposed, for the delivery of messages in delay tolerant networks. It is a protocol for the effective routing of DTN messages. It is based on prioritizing both the packets to be schedule for forwarding and dropping. It works based on the history of event like previous intermediaries, head-start, and acknowledgements. In [6], mobile hosts actively modify their trajectories to transmit messages. We develop algorithms that minimize the trajectory modifications under two different assumptions: (a) the movements of all the nodes in the system are known and (b) the movements of the hosts in the system are not known.

In carnet [8], it places radio nodes in the cars which communicate with grid. It uses a novel scalable routing system for the delivery of messages. Geographic routing uses geographic forwarding and scalable distributed location service to route packets from car to car without flooding the network. Both [7,8] transmit messages to a predefined geographical region. They are suitable for location-based services such as position-based advertising and publish-and-subscribe.

Repeated geocasts or time stable geocasts [10] could also be used to maintain Geocache in a certain area and bear similarities to our baseline scheme. It is different in concept though in that it requires the definition of a geographic region, which is not needed in our case. Most geocast schemes concentrate on routing messages to the areas of interest, or distributing messages to all nodes [9], [11], while Geocache is established close to the anchor location and needs only be known to very few nodes. Further, time-stable geocasts continuously remain in the region of interest, while Geocache can travel away from the anchor location.

In [12], it mentions some trajectory concepts, but it fails to take into account the peculiarities of vehicular networks and still only forwards data to a node that is physically closer to the destination. Geopps [13], [14] are maybe the most similar works to ours; however, it requires each mobile node to have full topology information which is not feasible in realistic scenario.

The Boomerang [15], has proposed a new technique to tying data to the geographic location, which uses neighbor nodes for the maintenance of data. The node selection is based on the direction and distance from anchor node. Whenever a carrier moves away from anchor node it sends back the Geocache to the anchor node. This challenges the network conditions like node failure, because the data has to be route back to the anchor node. The routing of packets of Geocache data introduces the traffic in the network and there is no solution specified for the maintenance of Geocache when there is a carrier failure, for the support of location based query processing.

We propose a solution for the identified problem with the following methodology, to support the maintenance of Geocache even at natural disaster, carrier failures in order to execute the location based queries. The proposed system contains the following components named Hook- the location of interest and the Carrier- the mobile node which keeps the data of Geocache and travelling nodes.

III. PROPOSED SYSTEM:

The proposed system maintains four numbers of carriers at four quarters of the perimeter of the coverage of the Hook. It divides the coverage area into four quarters with each 90 degree and selects four carriers at each part to maintain the Geocache.

The carrier sense for any update in the hook with a broadcast message in the network. Whenever it moves away from the hook it broadcast a message in the network to handover the Geocache data. When it receives reply for the handover message it analyses the replies. The selection of carrier is done by the previous carrier using the location and direction and speed metrics. Based on the spatial and geometric metrics it identifies a carrier which is moving towards the Hook. Each carrier is responsible for selection of its own next coming carrier.

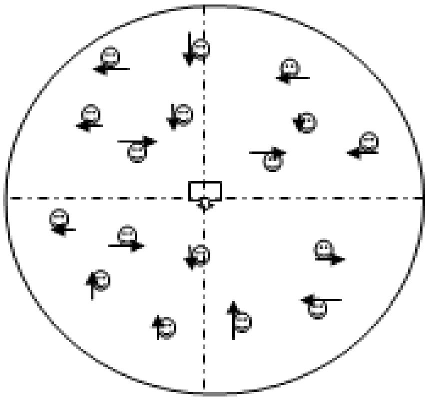


Fig1: shows the four quarters of hook with nodes with different direction at each.

The carrier is selected by a mobile vehicle based on the following metrics.

- Number of forward vehicles (moving towards Hook).
- Number of junctions present in available paths between vehicle and Hook.
- The distance tolerant between vehicle and Hook.

Based on these metric a cumulative spatial closure factor is computed to select the next carrier to keep the Geocache of the location of interest.

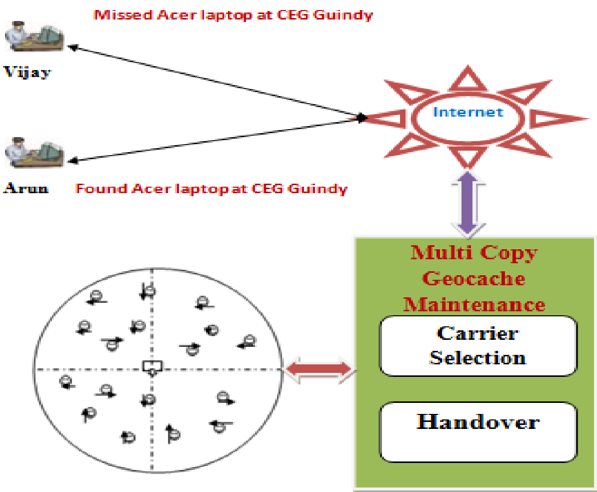


Fig2: architecture of the proposed system.

Multi-Copy Geocache maintenance:

In the multi copy Geocache maintenance algorithm, the hook maintains multiple copies of Geocache, one at each quarter of the perimeter of coverage. It will select a carrier initially at each of its perimeter quarter and stores the data in the node for initialization. At later stage when the vehicle moves out of hook then the vehicle initializes the handover process with its neighbor to exchange the Geocache and selects a new carrier for the cache what it has.

Carrier Selection:

The process of carrier selection done using various inputs like neighbor matrix, spatial and geometric data. The node which wants to handover the Geocache will initiate the carrier selection process. First it sends the spatial data request SREQ packet to all its neighbor and wait for the reply. When it receives the reply from its neighbors it extracts the location loc, speed s, direction d from the reply and store in the spatial matrix sp. It computes the spatial closure for each of its neighbor and sort the neighbor based on calculated sc. It verifies the presence of neighbor in its quarter by computing quadratic closure and if the neighbor is not present in its quarter it simply removes the nb id from the list. So that the neighbor will not be selected as a carrier.

Algorithm:

Step1: start
Step2: initialize neighbor matrix nb.
Step3: initialize spatial matrix sp.
Step3: for each neighbor nb_i from nb
    Send spatial data request SREQ.
    Receive spatial data reply SREP.
    Sp_i=SREP(loc,speed,direction).
  End
Step4: for each nb_i from nb
    Compute spatial closure sc.
    Sc=(1/nd)×(sqrt(l+(s×t))×log(n)).
    Nd =no of diversions
    L –location metric
    S – speed
    T – time slot
  End
Step5: sort nb_i according to sc.
Step6: for each nb_i from nb
    Compute quadratic closure cs.
    Cs = loc€qa©
    If cs == true
        continue
    Else
        Remove nbi from nb.
        Nb=ℂ(nb(nb_i)).
    end

Step7: select node with top closure and direction towards location of interest.
Step8: stop.

Handover:

The process of handover is initiated by the carrier node, the decision of handover is taken by the carrier when it moves away from location of interest. It is not necessary that the carrier should be going out of reach of the location of interest (hook), it will be initiated even if it moves between quarters of perimeter of the hook. It will be initiated because there may be a immediate diversion in the next quarter towards what the carrier approaches.

From fig3 it is clearly shows that there are two inner circles, which shows the carrier selection boundry. The selected carrier should be between the boundry so that to avoid frequent handover and carrier selection. To improve the performance of the proposed system the distance between both circles can be reduced. The red vertical line shows the boundry based on which the carrier will initiate handover process.

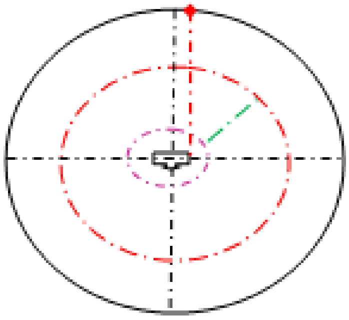
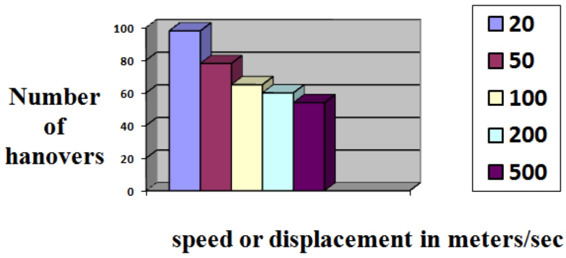


Fig3: shows the handover situation

Algorithm:
Step1: start
Step2: compute distance from hook (hd) and quadratic distance (qd).
$Hd= sqrt((hd_x-c_x)^2 \times(hd_y -c_y)^2)$
$Qd = sqrt((qb_x-c_x)^2\times(qb_y-c_y)^2)$
Step3: if qd>= l©
Initiate carrier selection .
C_n = cs(nb).
Start handover with C_n.
End
Step4: if no carrier found forward data to hook.
Step4: stop.

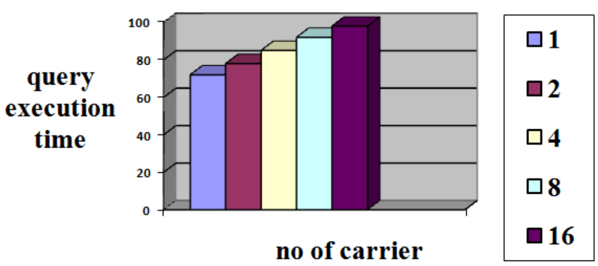
Results and Discussion:

The proposed system has implemented and tested with various simulation parameters and setting with NS2. It has produced very good results with the settings like 500 nodes with traffic pattern of 3000 from the road maps of Chennai. It produced various efficient results with the parameter settings.



Graph1: shows the handover frequency

From figure 1, it is clear that the number of handover is higher when the speed and displacement is higher.



Graph 2: query execution efficiency

From figure 2, it is clear that the query execution efficiency is proportional to the number of carrier used in simulation. So that if we increase the number of carriers around the hook it will increase the efficiency of the overall system.

CONCLUSION:

The proposed multi-copy Geocache methodology maintains the multiple copy of location information around the location of interest. The maintenance of multiple copies reduces the failure of location based query processing, because if a carrier failure occurs another copy of cache will be used for processing the query. This increases the overall efficiency of the system. Keeping multi copy of cache also reduces the query processing time and reduces the overall network overhead generated by data transfer between outgoing carrier to the hook.

REFERENCES:

[1] S. Ratnasamy, B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan, and S. Shenker, "GHT: A Geographic Hash Table for Data-Centric Storage," Proc. First ACM Int'l Workshop Wireless Sensor Networks and Applications, pp. 78-87, 2002.

[2] P. Juang, H. Oki, Y. Wang, M. Martonosi, L.S. Peh, and D.Rubenstein, "Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with Zebranet," ACM SIGOPS Operating Systems Rev., vol. 8, pp. 96-107, 2002.

[3] I. Vasilescu, K. Kotay, and D. Rus, "Data Collection, Storage, and Retrieval with an Underwater Sensor Network," Proc. Third Int'l Conf. Embedded Networked Sensor Systems, pp. 154-165, 2004.

[4] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden, "CarTel: A Distributed Mobile Sensor Computing System," Proc. Fourth Int'l Conf. Embedded Networked Sensor Systems, pp. 125-138, 2006.

[5] J. Burgess, B. Gallagher, D. Jensen, and B. Levine, "Routing for Vehicle-Based Disruption-Tolerant Networks," Proc. 25th IEEE Int'l Conf. Computer Comm., pp. 1-11, 2006

[6] Q. Li, D. Rus, M. Dunbabin, and P. Corke, "Sending Messages to Mobile Users in Disconnected Ad-Hoc Wireless Networks," Proc. Sixth Ann. Int'l Conf. Mobile Computing and Networking, pp. 44-55, 2000.

[7] T. Small and Z.J. Haas, "The Shared Wireless Infostation Model: A New Ad Hoc Networking Paradigm (or Where There Is a Whale, There Is a Way)," Proc. ACM MobiHoc, pp. 233-244, 2003.

[8] R. Morris, J. Jannotti, F. Kaashoek, J. Li, and D. Decouto, "Carnet: A Scalable Ad Hoc Wireless Network System," Proc. Ninth ACM SIGOPS European Workshop, pp. 61-65, 2000.

[9] Y.B. Ko and N.H. Vaidya, "Flooding-Based Geocasting Protocols for Mobile Ad Hoc Networks," Mobile Networks and Applications, vol. 7, pp. 471-480, 2002.

[10] C. Maihofer, T. Leinmller, and E. Schoch, "Abiding Geocast: Time-Stable Geocast for Ad Hoc Networks," Proc. Second ACM Int'l Workshop Vehicular Ad Hoc Networks, pp. 20-29, 2005.

[11] R. Morris, J. Jannotti, F. Kaashoek, J. Li, and D. Decouto, "Carnet: A Scalable Ad Hoc Wireless Network System," Proc. Ninth ACM SIGOPS European Workshop, pp. 61-65, 2000.

[12] J. LeBrun, C.-N. Chuah, D. Ghosal, and M. Zhang, "Knowledge- Based Opportunistic Forwarding Invehicular Wireless Ad Hoc Networks," Proc. 61st IEEE Conf. Vehicular Technology, pp. 2289-2293, 2005.

[13] I. Leontiadis and C. Mascolo, "GeOpps: Geographical Opportunistic Routing for Vehicular Networks," Proc. IEEE Workshop Autonomic and Opportunistic Comm., pp. 1-6, 2007.

[14] I. Leontiadis and C. Mascolo, "Opportunistic Spatio-Temporal Dissemination System for Vehicular Networks," Proc. First Int'l MobiSys Workshop Mobile Opportunistic Networking, pp. 39-46, 2007.

[15] Tingting Sun and Bin San, "The Boomerang Protocol: Tying Data to Geographic Locations in Mobile Disconnected Networks" , IEEE transactions on mobile computing, vol. 11, no. 7, july 2012.

[16] U. Lee, E. Magistretti, B. Zhou, M. Gerla, P. Bellavista, and A. Corradi, "MobEyes: Smart Mobs for Urban Monitoring with a Vehicular Sensor Network," IEEE Wireless Comm., vol. 13, no. 5, pp. 52-57, Oct. 2006.

[17] V. Park and M. S. Corson. A highly adaptive distributed algorithm for mobile wireless networks. In Proceedings of INFOCOM'97, Kobe, Japan, April 1997.

[18] C. E. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. In Computer Communication review, volume 24, pages 234 –244, October 1994.

[19] Elizabeth Royer and C-K. Toh. A review of current routing protocols for ad hoc mobile wireless networks. In IEEE Personal Communication, volume 6, pages 46– 55, April 1999