International Journal of Computer Science Engineering,
and Information Technology Research (IJCSEITR)
ISSN(P): 2249-6831; ISSN(E): 2249-7943
Vol. 6, Issue 5, Oct 2016, 29-36

COOPERATIVE LEASES USING IMPROVED DELAUNAY TRIANGULATION FOR DATA CONSISTENCY IN CONTENT DELIVERY NETWORKS

GADIRAJU MAHESH[1] & V. V. R. MAHESWARA RAO[2]

[1]*Associate Professor, Department of Computer Science and Engineering,*
S.R.K.R. Engineering College, Andhra Pradesh, India
[2]*Professor, Department of Computer Science and Engineering,*
Shri Vishnu Engineering College for Women, Andhra Pradesh, India

ABSTRACT

Maintaining data consistency is ever lasting issue in online applications where data is replicated at many places for better data accessibility. In this work the problem of maintaining data consistency in Content delivery networks is considered. Previous techniques for maintaining data consistency like push technique using hierarchical network of proxies suffer from the disadvantage of less resilience to failures. For getting better resilience to failures, the cooperative leases technique using single level Delaunay triangulation is explored and it is extended to multilevel Delaunay triangulation. A novel technique for effectively notifying the data updates at sources to proxies namely cooperative leases using multilevel Delaunay triangulation in Content delivery networks is proposed. This technique achieves higher resilience to failures than the push technique using hierarchical network of proxies. The proposed technique improves the performance of the cooperative leases technique using single level Delaunay triangulation for maintaining data consistency in content delivery networks.

***KEYWORDS:** Content Delivery Networks, Dynamic Data Dissemination Networks, Delaunay Triangulation, Data Aggregator*

Received: Aug 21, 2016; **Accepted:** Sep 13, 2016; **Published:** Sep 16, 2016; **Paper Id.:** IJCSEITROCT20164

Original Article

INTRODUCTION

Effective delivery of dynamic content to the web users is the need of the present day. Many applications including scientific applications and business applications require dynamic data. Examples of the dynamic data are different parameters of the atmosphere like pressure, temperature, humidity and share prices of different companies in stock markets. They should be delivered to the users as soon as they are updated within no time. Otherwise the data becomes stale. It is like informing a user about possible rain after it rained or informing an older value of a company share prices.

Content delivery networks especially dynamic data dissemination networks proposed by Shetal Shah et al. [8] deliver such dynamic data effectively to the users. Dynamic Data Dissemination Networks (DDDNs) are content delivery networks in which proxy servers are called Data Aggregators(DAs) where the frequently used dynamic data is cached. Dynamic data dissemination network is a hierarchical network of data aggregator. Whenever a request for data arrives for the first time, the data is obtained from the source and delivered to the user and a copy of the data is cached at DA for further use. In the event of further requests for the same data, the cached data at the data aggregators is delivered to the user requesting the data. The data required by the users is delivered by the data aggregator which is nearer to the users effectively. In such networks, server load is distributed to proxies or data aggregators.

Dynamic data dissemination networks use a network of data aggregators for progressively conveying the data to users. As discussed in [2, 5, 7], a separate tree of data aggregators is constructed for every data item by using source of the data item as the root of the tree in Dynamic data dissemination networks. A dynamic data dissemination graph as shown in Figure 1 that is constructed by applying union of all the data dissemination trees. In the Figure 1, SA and SB are sources of A and B data items, C1, C2, C3, C4, C5 are clients requesting the data. All the other ellipses are data aggregators. To insert a new node in data dissemination graph, there are two techniques proposed by Shetal Shah et al. [6]. The first technique is level at a time algorithm and another one is data at a time algorithm. A DDDN can be constructed using one of the techniques. The higher level data aggregators get the data from sources and lower level data aggregators get the data from higher level data aggregators. The clients could get the data items from data aggregators. If more than one data aggregator replicates the same data item, then the client gets the required data item from the optimal data aggregator as discussed in [3, 4].

The optimal data aggregators serve the data to users much efficiently within no time. since the same data item is duplicated on many data aggregators, the values of different copies of the data may not agree with each other. This creates the problem of data inconsistency. The values of a data item at the data aggregator and at the data source should agree with each other for the data to be consistent. This data consistency is difficult to be maintained and it is costly. But, the users may not require data items to be 100 percent consistent. They require the data at some specified allowable deviated value from its original value. The maximum allowable tolerance of data value error is called data incoherency bound. So, in data dissemination networks one of the main tasks is to propagate the data to clients confirming to the coherency requirements of the data. This task is done by propagating the data updates from source to data aggregators frequently that keeps the data coherent. Propagating all the updates may be costly and not necessary. So, if the incoherency of data item at a DA exceeds the incoherency bound then only the updates are propagated from source of the data to the DAs. These updates are propagated from source of the data item to the DAs through the network of data aggregators as shown figure 1.

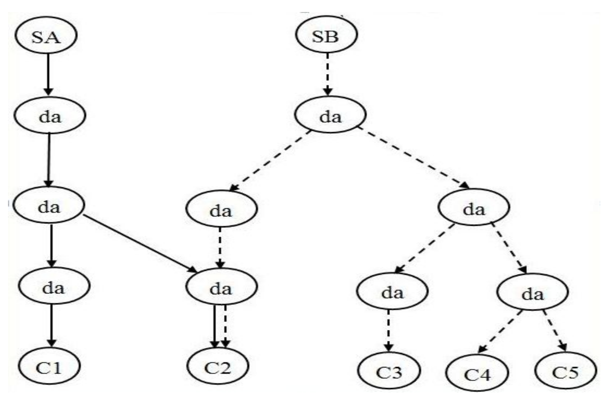

Figure 1: Hierarchical Network of Data Aggregators in Dynamic Data Dissemination Networks

The techniques for maintaining the data consistency at the proxies in content delivery networks are discussed and compared by PavanDeolasee et al. [10] and AnoopNinan et al. [9]. They are listed below.

- **Push:** Pushing the data updates from source to proxies /data aggregators (DAs) whenever the value of data at source changes and incoherency at the DA is greater than incoherency bound.

- **Pull:** DAs pull the updates from source whenever the DAs expect the updates to data.
- **Push-or-Pull:** Data updates may be pushed from sources or clients may pull the data according to client requirements.
- **Push-and-Pull:** Pull policy is followed initially. But, in the event of frequent data updates, the policy may be changed to push.
- **Leases:** Clients lease the sources for pushing all the required data updates for an agreed time interval.

COOPERATIVE LEASES TECHNIQUE USING DELAUNAY TRIANGULATION IN CDNS

The different techniques like pull, push, push-or-pull and push-and-pull can be used for maintaining data consistency in CDNs. All these approaches have low scalability. The cooperative leases approach proposed by AnoopNinan et al. [2] overcomes this problem. This method is based on leases for distributed file cache consistency proposed by Cary Gray and David Cheriton [12]. The meaning of leases in dynamic data dissemination networks is that there is an understanding between data aggregators and the server for the server to push the updates of a data item to the data aggregators in the period of lease. The aggregators could request for a lease whenever it requires consistent data. The basic lease includes the following values.

- Object for which lease is requested
- The data aggregator that requests the lease
- The lease time

In such type of basic lease technique all data updates are notified to the data aggregators from source. The disadvantage is that the load on the server increases for notifying all the updates to all the data aggregators taking the lease and all the data aggregators that require the same data item have to take a separate lease. To avoid these problems, cooperative leases approach of maintaining data consistency is proposed by AnoopNinan et al. [9]. This reduces the load on the server and also data aggregators. Cooperative leases also support delta consistency which provides flexibility. This policy uses application level multicast to notify data updates that reduces the server load and improves performance and scalability. The delta consistency means that the data updates are notified for every delta time instead of all updates in the lease period ensuring the data at the client is never incoherent by more than delta time from its server version of the data. Here delta is notification rate.

In cooperative leases approach, instead of granting lease to each and every node, groups of nodes are formed and leases are granted to leader node of each group on behalf of the group. The server interacts only with the leaders of the groups and transfers the updates only to leaders. It is the responsibility of the leader to propagate the notification to other nodes in the group. So, the leader shares the server load of propagating the notifications to all the data aggregators reducing the over head of the server. The cooperative leases tuple contains fields O, G, L, t and r. Here O is the Data object of interest., G is the group taking the lease, L is the leader of the group, t is time for which lease is taken and finally notification rate is r.

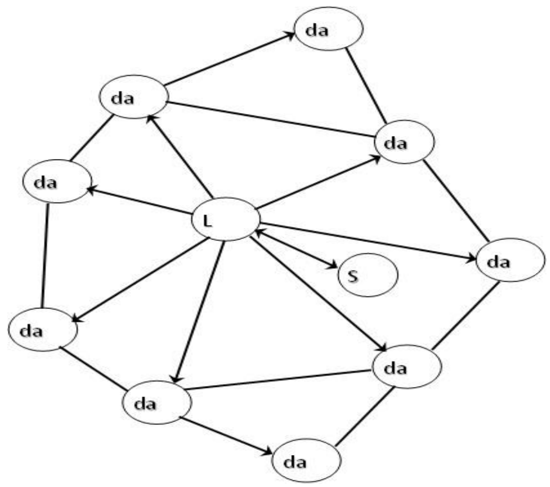

Figure 2: Illustration of Delaunay Triangulation for Data Aggregators of a Group

The cooperative leases using minimum spanning tree explained in the preceding section has low resilience to failures of nodes. In order to improve the resilience to failures, cooperative leases technique using application layer multicast with Delaunay triangulation in CDNs was proposed by Gadiraju Mahesh and V. ValliKumari [1]. In cooperative leases, the updates at the source are informed to the leader of the group only. Then It is the duty of the leader to notify the data updates to the data aggregators that are members in the group. The notifications may be propagated using application layer multicast with Delaunay triangulation effectively than using a hierarchical network. The Delaunay triangulation for the data aggregators in a group is shown in figure 2. In the figure, L is the leader node of the group. The members of the group are represented by ‘da’. S is the source of the data. The Figure 2 shows an example of Delaunay triangulation. A triangulation of a set of points in which all points are within the circumscribing circle of any triangle is known as Delaunay triangulation. In application layer multicast using Delaunay triangulation, a data aggregators group is formed for each data item and leader is designated. Triangulation for each group of DAs is constructed by keeping the triangulation equiangular and considering a DAs of each group as a node in the triangulation. Compass routing [11] could be used for finding the route to a members of the group from the leader node. compass routing does not require the absolute positions of all the data aggregators. It requires the relative positions of leader and adjacent data aggregators only. In this routing a destination data aggregator could find its parent for a source out of optional adjacent data aggregators by choosing line path which makes least angle with line joining destination and the source. In the example shown in Figure 3, let L be the leader DA and A be the DA whose parent DA is to be established. Let O1 and O2 be the optional parents. The angle between AL and AO2 is less than the angle between AL and AO1(i.e., θ2< θ1). So, O2 is chosen as an optimal parent of A.

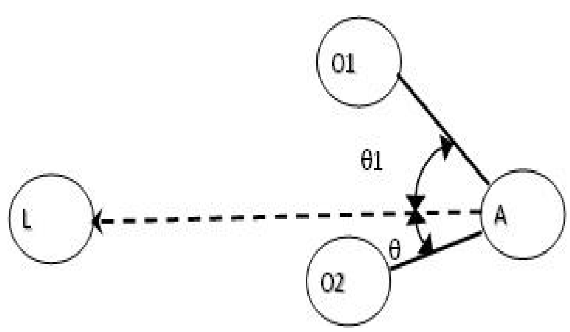

Figure 3: Illustration of Compass Routing

COOPERATIVE LEASES TECHNIQUE USING MULTILEVEL DELAUNAY TRIANGULATION IN CDNS

The advantage of the Cooperative leases technique using Delaunay Triangulation in CDNs is that it has more resilience to failures than the other techniques like push technique using hierarchical network of proxies. For optimally constructing the Delaunay overlay the geographical location is taken as the coordinates of the nodes. This network is advantageous only if the nodes in a group are located geographically in the same area. But the Cooperative leases technique using Delaunay Triangulation in CDNs proposed in [1] collected all the data aggregators requiring the same data item into one group. But all such data aggregators may not be geographically located in the same area. They may also span in different continents also. So, it is not a good idea to make a group of data aggregators which are geographically far away.

So, in this paper, a multilevel Delaunay triangulation instead of a single Delaunay triangulation of DAs for each data item is proposed. A two level Delaunay triangulation is considered in this work. In this scheme one Delaunay triangulation is constructed with DAs disseminating the same data item for each geographical area. So, many Delaunay triangulations are constructed for propagating a single data item instead one Delaunay triangulation for each data item. So, the DAs in each group are geographically nearer. A leader is established for each of the groups. The leaders at each group again form another top level triangulation. This top level triangulation has all the bottom level leaders as its members. For the top level group source acts as leader. Then all the leases are taken by the members of top level group that are also leaders of bottom level groups. So, the source transfers the update notifications to the top level group members. Then the notifications are consequently multicast to the DAs in the bottom level group.

In this cooperative leases scheme, the leader of each bottom level group is determined by the formula H(U)% N. Here H is hash function, U the URL of the data object, the number of members of the group is N and % is the modulus operator. The source is selected as top level group leader without requiring leader selection. As explained in the previous section, the lease record contains attributes O, I, L_i, t_i and r_i. Here O is the Data object for which data updates are required, I is the group id, Li is the leader of the group I that takes lease on behalf of all its group members, t_i indicates the time for which lease is required by the group I and r_i is rate at which updates to O are required to be notified to group I within the lease time t_i.

The leaders of each bottom level groups that are also members of top level group take the lease from the source. Then the updates to the object O are notified to the group leader L_i during the specified period t_i. In this period t_i, all the updates are notified to the leader L in ordinary leases. But in this work cooperative leases technique is used. So, the updates to O during lease period t_i are notified at the rate r_i only. The cooperative Leases technique provides such flexibility.

The scheme is illustrated in Figure 4. In the figure there are five areas. So, five groups are formed and Delaunay triangulation is constructed for each bottom level group. L1, L2, L3, L4, L5 are leaders. The source is S that is also leader of the top level group. The required updates are notified by the source to the leaders L1, L2, L3, L4 and L5. Then each leader forward the notifications to the group members. The figure also shows the path established from leader to members in each group using compass routing explained in previous section. In compass routing a node decides its parent out of the optional nodes by selecting path that makes least angle with the line joining the destination node and the source node. In this scheme the DAs in a group are geographically closer to each other making the Delaunay triangulation network more efficient. Another advantage is that the task of transmitting the notifications is distributed among all the leaders of bottom level groups. This reduces the load on the source that is top level group leader. This technique also has the advantages of scalability and flexibility as it uses cooperative leases instead of ordinary leases.

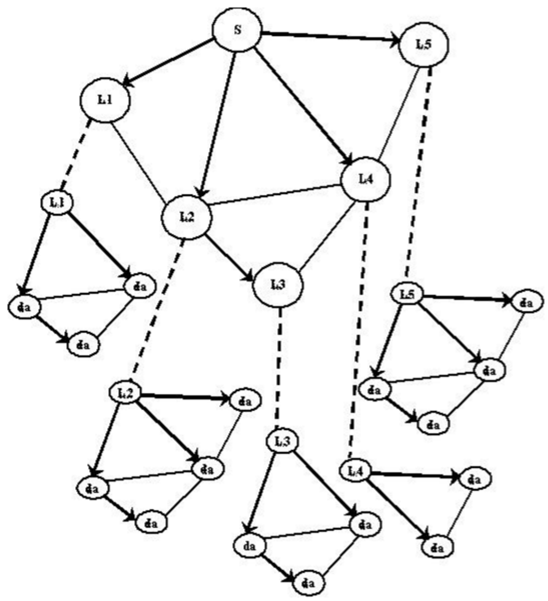

Figure 4: Illustration of Multilevel Delaunay Triangulation

RESULT ANALYSIS

The performance of single level Delaunay triangulation and multilevel Delaunay triangulation are compared. For comparing the two topologies, networks of 100 to 1000 nodes are taken. For the single level Delaunay triangulation, all the nodes are grouped together into a single group and one of the nodes is taken as leader. For the two level Delaunay triangulation the nodes are divided into ten groups according to their geographical proximity and a separate leader is assigned to each the groups. These groups form the bottom level Delaunay triangulations. The leaders of these groups form the top level group and source is the group leader. By assuming the failures of nodes starting from 0% failure upto 50 %failures of nodes, the loss of number of notifications are observed for both the networks. The failures are distributed following random distribution. For both the networks the notifications lost are observed. Then average percentage of notification failures is calculated and tabulated.

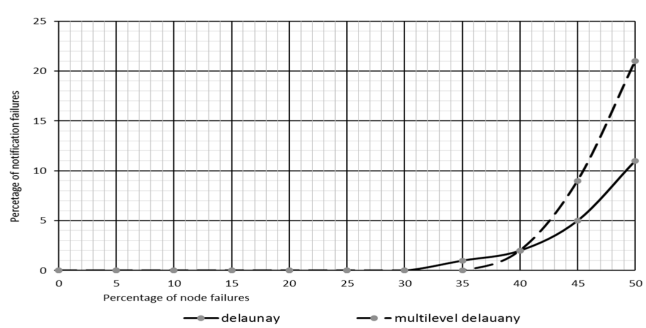

Figure 5: Graph Showing the Comparison of Single Level Delaunay Triangulation and Multilevel Delaunay Triangulation

A graph is plotted as shown in figure 5 with percentage of node failures on the axis and percentage of notification failures on the y-axis for both single level and two level Delaunay triangulation networks. The two level Delaunay triangulation has better resilience to failures than that of single level. The reason for better performance of the multilevel triangulation is that the failures in one group is localized to that group only. Then the proposed algorithm is also compared with Minimum Spanning Tree(MST) applying the same setup with 100 to 1000 nodes. Again 10 groups are assumed for the multilevel triangulation. The failures are assumed to be randomly distributed and results are observed and plotted in figure 6. As observed in the graph the proposed multilevel Delaunay triangulation network has far better performance than Minimum spanning tree.

Two level Delaunay triangulation is considered for multilevel triangulation. In these results, 10 groups are considered and the nodes and node failures are assumed to be randomly distributed. The results may vary depending on the number of groups formed and the distribution of node failures. Whatever may be the distribution of node failures, surely the proposed multilevel Delaunay triangulation has better resilience to failures than the MST and single level Delaunay triangulation. Another advantage of multilevel Delaunay triangulation is that the load of multicasting update notifications is distributed to leaders of each group.

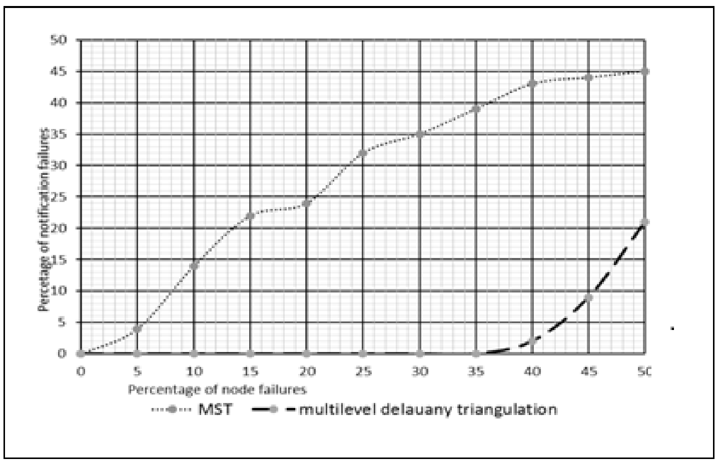

Figure 6: Graph Showing the Comparison of Minimum Spanning Tree and Multilevel Delaunay Triangulation

CONCLUSIONS

A novel technique namely Cooperative Leases using Multilevel Delaunay triangulation in content delivery networks for maintaining data consistency is proposed in this paper. The results show that the proposed multilevel Delaunay triangulation has better resilience to failures than the Minimum spanning tree and single level Delaunay triangulation. The proposed consistency maintenance technique is better than the previous techniques like push, pull for recovery from the node failures. The proposed technique distributes the load on a single leader node to each of bottom level group leaders providing better scalability. The technique for maintaining data consistency in CDNs proposed in our previous work[1] is extended here and proved that the extension is more effective. The work may be extended to cloud computing platform more specifically to CDNs using cloud infrastructure.

REFERENCES

1. *Mahesh, G., &Kumari, V. V. "Data Consistency in Dynamic Data Dissemination Networks", InInternational Journal of Engineering Research and Technology Vol. 3, No. 2 February-2014. ESRSA Publications.*
2. *Mahesh, G., &Kumari, V. V. "Optimal Policy of Data Dissemination in CDNs", International Journal of Computer Applications, 73(17) 2013.*
3. *Mahesh, G., &Kumari, V. V. "Primal-dual parallel algorithm for continuous aggregate query dissemination", In International Conference onAdvances in Computing, Communications and Informatics (ICACCI), pp. 545-549 IEEE, August 2013.*
4. *Mahesh, G., &Kumari, V. V. "Distribution of continuous queries over data aggregators in dynamic data dissemination networks", In International Conference on Advances in Information and Communication Technologies pp. 584-589, Springer Berlin Heidelberg, September 2010.*
5. *Gupta, R., &Ramamritham, K. "Optimized query planning of continuous aggregation queries in dynamic data dissemination networks", InProceedings of the 16th international conference on World Wide Web pp. 321-330, ACM, May 2007.*
6. *Shah, S., Ramamritham, K., &Shenoy, P. "Resilient and coherence preserving dissemination of dynamic data using cooperating peers", IEEE Transactions on Knowledge and Data Engineering, 16(7), pp: 799-812, 2004.*
7. *Shah, S., Dharmarajan, S., &Ramamritham, K. "An efficient and resilient approach to filtering and disseminating streaming data", In Proceedings of the 29th international conference on Very large data bases-Volume 29 (pp. 57-68). VLDB Endowment, September 2003.*
8. *Shah, S., Ramamritham, K., &Shenoy, P. "Maintaining coherency of dynamic data in cooperating repositories", In Proceedings of the 28th international conference on Very Large Data Bases (pp. 526-537). VLDB Endowment, August 2002.*
9. *Ninan, A., Kulkarni, P., Shenoy, P., Ramamritham, K., &Tewari, R. "Cooperative leases: Scalable consistency maintenance in content distribution networks", In Proceedings of the 11th international conference on World Wide Web (pp. 1-12). ACM, May 2002.*
10. *Deolasee, P., Katkar, A., Panchbudhe, A., Ramamritham, K., &Shenoy, P. "Adaptive push-pull: disseminating dynamic web data", InProceedings of the 10th international conference on World Wide Web pp. 265-274, ACM, April 2001.*
11. *Kranakis, E., Singh, H., &Urrutia, J. "Compass routing on geometric networks", In in Proc. 11 th Canadian Conference on Computational Geometry, 1999.*
12. *Gray, C., &Cheriton, D. "Leases: an efficient fault-tolerant mechanism for distributed file cache consistency", Operating systems review, 23(5), 202-210, 1989.*