Int. J. Advanced Networking and Applications
Volume: 09 Issue: 05 Pages: 3588-3595(2018) ISSN: 0975-0290

Consistent Replicated Mobile Data Broadcasting for Read-Write Mobile Clients (CRMB)

Dr. S. Abirami
Asst. Prof, Dept.of CA&IT
Thiagarajar college(Autonomous)
Madurai-9,Tamil Nadu
viswaa2005@gmail.com
Dr. K. Chitra
Asst.Prof, Dept.of Computer Science
Govt.Arts college,Melur
Madurai, Tamilnadu
Dr. K. Senthamarai kannan
Professor, Dept. of Statistics
Manonmaniam Sundaranar University
Tirunelveli, Tamilnadu

ABSTRACT

Miraculous growth in mobile technology increases the mobile usage day by day. Mobile data traffic and the limitation of power backup are still the major problem in disseminating consistent data on mobile database environment. In this paper, the author propose a consistent replicated mobile data broadcast algorithm(CRMB) to improve the performance of broadcast approach by providing high data availability, consistency and currency with the minimum data access delay and uplink communication. When an update transaction is executed on replicated mobile database environment, it preserves the consistency among the fixed host server, replica servers and the read-write mobile clients. The implementation results show that the performance of proposed approach is better in most cases. Key features of the proposed approach are:

- **It disseminates consistent data quickly to all mobile clients in the same order in which they were updated through minimum bandwidth communication**
- **It allows the mobile clients to get broadcasted data items from any nearest replica server without contacting the server.**

Key Words: **Mobile computing, Data Replication, Consistency, Data Dissemination, Temporal database**

Date of Submission: Feb 10, 2018 Date of Acceptance: March 07, 2018

1. INTRODUCTION

With the enhancement in mobile technology most of the e–commerce applications rely on broadcast based mobile database applications [6][7]. Broadcasting consistent data is a significant requirement for mobile database applications. Delivery of fresh data to the mobile clients leads the business to a good place. If the mobile application broadcasts outdated data (inconsistent data) then it may collapse the entire system [8][9][13].The data inconsistency may occur due to the overlapped execution of update and broadcast transactions.

To provide high data availability and quick data access, the server database can be duplicated among several replica servers [5][15][20]. Since most of the mobile database applications rely on asymmetric communication environment [16][17][21][24], the implementation of replicated database environment should not affect the consistency and bandwidth requirements. The mobile database on server can be duplicated using passive or active replication scheme. In passive replication, mobile clients can get the fresh updated data only from the master server. If the clients get the updates directly from any replica server then it is called active replication scheme. Whenever the master database server is updated then updates can be applied to all replicas immediately (Eager update propagation) otherwise updates can be given to all replicas after the commitment of update transactions on master server(Lazy update propagation) [1][10][12].Whenever the master server is updated, the master server can transfer the entire updated content to all replicas (Content delivery) otherwise the server can transfer the description[2][3][4] about the update transaction to all replicas (Log based delivery) and there should be a proper synchronization among replicas to achieve consistency. Still the existing replicated data broadcast approaches efficiently deliver the consistent data to the mobile clients, the order of updated data item, size of update transactions and network communication for special cases of database were not considered.

In this paper, the author propose an algorithm for broadcasting consistent, current and ordered data to the read-write mobile clients using replicated mobile database environment. It guarantees maximum data availability and dissemination of current data with minimum bandwidth requirement. The following section-2 describes the related work and their limitations. Section-3 presents the recommended replication system model. The algorithm of

Volume: 09 Issue: 05 Pages: 3588-3595(2018) ISSN: 0975-0290

proposed system is described in section-4. Section-5 discusses the experimental results of the proposed approach. Conclusion and future enhancements are specified in section-6.

2. RELATED WORK

In [23], they proposed an optimistic single master replication scheme. It was derived from MS_RDA (Remote Data Access). The proposed ERDA method employs pull based data communication between master and replica servers. Similar to RDA, ERDA provides synchronization and eventual consistency among replicated servers. But RDA allows the data flow in one direction at a time and it increases the bandwidth consideration. ERDA implements log based data transfer instead of content transfer. ERDA avoids the overlapping of data sets between replicas by partitioning and normalizing database to be replicated. But there can be update propagation delay due to the pull based delivery and each replica is responsible for their consistency. Update transaction requests of mobile clients were not considered.

In [18], they have addressed the approaches for replica relocation, consistency management and location management for peer to peer MANETS. The three approaches (SAF, DAFN, and DCA) were proposed for replica relocation. SAF allocates the data in descending order of their access frequency. But there is a low data access if more mobile hosts have the same access characteristics. The DAFN allows the mobile hosts to share among neighboring hosts. DCA allows the mobile hosts to share replicas among larger set of mobile hosts. It increases the data accessibility. But there is a network overhead due to exchange of information among larger group.

For consistency management, they proposed Update Broadcast and Connection Rebroadcast methods. The host containing original data broadcasts the IR (Invalidation Report) every time when it updates the data. Then replicas update the data items by comparing their TS (Time Stamp) with TS in IR. In connection rebroadcast method, it spreads the IR among the large set of mobile hosts even though they were not directly connected with mobile server host. For location management, they proposed AL (Access Log) and GM (Gateway) methods. These methods provide the data structures for finding the location of our requested data. These proposed approaches well suitable for MANET emphasis on data access frequency but their performance would be poor for different data access patterns and they need additional data structures for dynamic allocation of replicas.

In [22], they proposed the methods for maintaining eventual consistency and for propagating updates. To distribute the updates among the replicas, they proposed pull based single master system using time stamp. Second is pull based multi-master log transfer method. But these pull based approaches could not provide the complete synchronization among replicas instantly. In order to order the updates, they proposed total and partial update ordering methods. In total update ordering method, all replicas accept the updates in same order. In partial update ordering, the replicas can receive the updates in different order. To detect data conflict among replicas timestamp method is proposed but this approach might increase the number of aborted transactions.

In [11], they suggested the two tier based replication scheme. Here, the mobile nodes have two versions of data: Master and Tentative. The master version holds the most updated value; tentative version has the local updates. The consistency is enforced through two transactions (Tentative, Base). Tentative transactions processed on local data at mobile node. A Base transaction processed on master server database. Whenever the base transaction is in execution, if there is any disconnection then the tentative transaction would be processed. If the tentative transaction is failed to meet consistency constraints then it would be aborted to retain consistency. Since it allows the tentative transactions, it is preferable for frequently disconnected environment but it could not give better performance for the dissemination of dynamic data to read-write mobile clients.

Bayou proposed an approach [19], it allows the mobile clients to read and write any replica. Here, primary server maintains write log. Whenever it is updated, it propagates the write log to all the replica servers. Each replica severs have two versions of data: tentative, committed. When replica is locally updated then the data is tentative. Replica server propagates the updated data to other replica using version vector. When a write is added to primary copy write log, it can be aborted or re executed based on dependency checks. Still this protocol handles read-write requests of mobile clients, but this type of eventual consistency is not suitable for temporal database systems.

DREAM is proposed in [14] improves the data accessibility by replicating hot data items before the cold at servers. It handles the real time transactions by giving higher priority for replicated data that are accessed frequently by firm transactions instead of soft transactions. Read-only data are accessible from any replica based on the type of transactions. When the data is updated, it would be forwarded to primary copy server. It re-executes or cancels the updates based on type of transaction. Primary copy server does the synchronization with other replica using time stamp mechanism. This approach is opted for real time transactions executed in Ad-Hoc network but it could not give balanced performance for different data access patterns.

3. SYSTEM MODEL

The author described the proposed replica system to answer the following criteria, who can issue update? How the server propagates updates? Who broadcast the consistent data to the mobile clients? The proposed algorithm applies the single master- active replication scheme with the lazy update propagation mechanism. Here, the database server considered as master and it is responsible for propagating updates either from external sources or from mobile clients. In order to reduce the bandwidth requirements, the proposed system pursue the log based data delivery to get the updates from the master server to replica servers. To enhance the performance of consistent data broadcast approach with the high data

Volume: 09 Issue: 05 Pages: 3588-3595(2018) ISSN: 0975-0290

availability and accessibility, the authors designed the system so that mobile clients can get the broadcasted data from any nearest replica server with in their cell region and the update transactions can execute only on master server. The following Figure.1 shows the recommended system model.

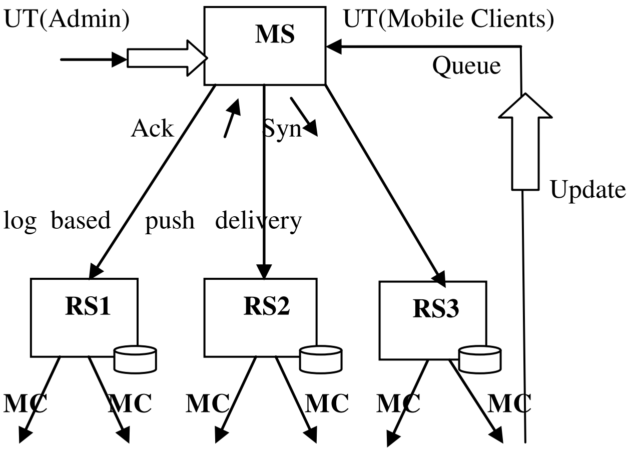

Figure.1 System Model

In CRMB approach, whenever the master server is updated it is responsible for propagating updates to all replica servers then replica servers have to broadcast the updated data items to the mobile clients using synchronization mechanism.

3.1 Properties of the proposed CRMB Algorithm

The proposed approach delivers the newly updated data to the mobile clients with the following properties

- The update transactions executed only on master server and the replica servers disseminate the consistent data items
- A server update transaction can be able to update any number of data items and the mobile clients' update transaction can update only one data item
- The proposed approach executes the update transactions differently based on type of database users (Admin user ,Mobile clients) and the type of data (Dynamic, Temporal)
- It maintains the consistency over multiple broadcast cycles on replicated environment through synchronization mechanism with the minimum network communication
- Updated data items would be re-broadcasted in the next immediate broadcast cycle to provide high data accessibility
- Since this approach uses index based broadcast mechanism, it is suitable even for large mobile databases as well as larger size of data items
- This approach propagates the updated data items from MS to RSs only once through Ack, Syn mechanisms.

4. Replication Algorithm

It maintains two phases of consistency between fixed master server to replica servers (using Ack and Syn signals) and replica servers to the mobile clients(using serialization mechanism).This algorithm provides consistency on replicated mobile database environment to the read-write mobile clients through three phases,

- Master server(MS)
- Replica server(RS)
- Mobile client(MC)

Since the preemptive execution of update transactions on MS may possibly cause data inconsistencies, we have designed the queuing system to receive update requests. Whenever the concurrent update transactions are given to MS for execution, they would be considered through queue. Based on consistency checks, update transaction would be executed or cancelled by the MS.

4.1 Master server phase

This phase is responsible for executing update transactions and propagates the fresh data to the replica servers. The MS could be updated either by administrator or by the mobile clients. The author suggested four cases to achieve consistency based on type of users and data. The following Figure.2 shows the process for each case.

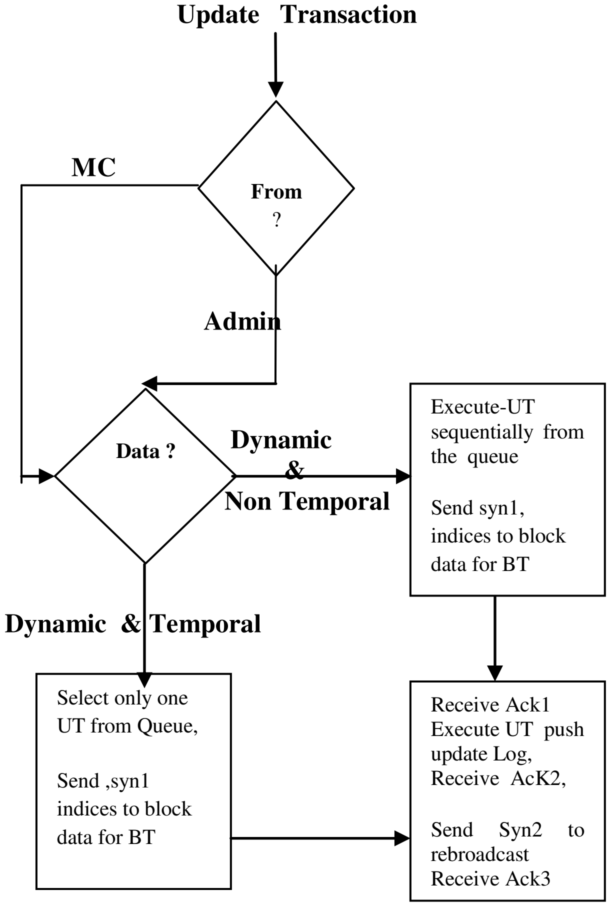

Figure.2 Process of Master server

Case 1: When the dynamic (data which are frequently updated) and non-temporal data (their value not depend on time) stored on database, if it is updated by administrator, they would be executed by the master server sequentially from the queue. When the MS retrieves any update transaction for execution, it blocks the data items (to be

Volume: 09 Issue: 05 Pages: 3588-3595(2018) ISSN: 0975-0290

updated) from broadcast transaction then it does the following,

- After Receiving Ack1,master server executes the update transaction
- Master server pushes update log transaction to all replica server (RS)
- After Receiving Ack2 from all Replica servers, the Master server sends Syn2 signal to all the replica servers for rebroadcast of updated data items.
- If the Ack3 has received then MS would consider next update transaction from queue

Example
When the master database contains the weather database (Dynamic, Non-Temporal), it would be updated by the external device frequently. The weather value of individual entity is not a temporal data. It can be executed after a short delay. Whenever the weather value is updated by administrator/external sources, it would be executed by the master server sequentially through queue then MS sends Syn1 signal along with indices of data (to be updated) to block those data from broadcast until the dissemination of fresh data to the MCs. After receiving Ack1 from all RSs, it executes the update transaction, sends the update log transaction to the RSs. By receiving Ack2 from all RSs, MS sends Syn2 for all RSs to rebroadcast the fresh data. After receiving Ack3, MS would ready to execute next update transaction from the queue.

Case 2: Whenever the administrator/external device issues the update transactions for the dynamic and temporal data on database, the updated data items are supposed to broadcast instantly for preserving consistency. The MS consider the update requests from the queue as follows

- MS guarantees the consistency by retrieving one update request from the queue by doing application specific comparison (MIN,MAX,>,<,=) between queue of requests
- When a request is taken for execution, all other update requests in queue for the same indices would be cancelled
- Then the MS would enter into the update transaction execution process and block the data from broadcast transaction
- After the selection of update transaction from the queue, the MS executes the process as that of case1.

Example
Once the stock trading database is updated by the administrator frequently, updated data items have to be broadcasted straight away. If an update transaction request (which is going to update the data items x1, x2) is taken from the queue, it compares all the update requests for x1 or x2.Then it retrieves the update transaction with the highest stock value from the queue, all other update requests for the same indices would be cancelled from the queue.

Case 3: When the MS is updated by the mobile clients, if it has the highly dynamic and temporal data (their value depends on time). Delayed broadcast of these data items (Outdated data) might collapse the entire system. Here, each update on a data item depends on its' previous value. MS does the following,

- MS consider the clients update requests from the queue as that of in case 2.

Example
When the auction database is stored on MS, it is frequently updated by MCs. In this case, whenever the current bidding value of an item is updated by MCs, their requests would be considered by the MS through queue by blocking data from broadcast transaction. Whenever the first request is considered by MS, it compares the new value of that transaction with other mobile client's update requests on queue for same index. Then it retrieves the update request with maximum value as new bidding value. Then all other update requests of mobile clients for the same indices would be cancelled by MS.

Case 4: If the MS is updated by mobile clients, MS has non-temporal data then it does the following,

- MS executes the update requests of all MCs one by one from the queue by blocking the data items (to be updated) for broadcast transaction. Then MS does the process as that of case 1.

Example
When the polling database is stored on MS, it is frequently updated by MCs. Here, the mobile users can submit their blind writes to the server. In this case, whenever the number of votes for a contestant is updated by the MCs, their requests would be considered by the MS sequentially through queue by blocking the data items (which are in current update transaction) from broadcast transaction.

4.2 Replica Server Phase

This phase is responsible for broadcasting consistent data pushed by MS. This phase described with the following data sets,
BD → Set of data to be broadcasted
BI → Set of indices of data to be broadcasted
UD →Set of updated data
LI → Set of indices of data to be blocked.

This phase broadcasts the updated data items to the MCs in the same order. It disseminates the replicated database content in the series of regular intervals (broadcast cycle). If the RSs have received Syn1 along with the indices set then they have to block those data from the broadcast transaction by making the indices set as LI. Then all RSs have to send back Ack1 to MS. Whenever the RSs received update log transaction (update transaction description), they set the updated data items on UD. After executing and committing update log transaction, RSs send Ack2 to MS for giving update-completion signal. Whenever the RSs have received Syn2 signal from MS then all RSs would rebroadcast the updated data items

Volume: 09 Issue: 05 Pages: 3588-3595(2018) ISSN: 0975-0290

from UD in same order by comparing UD with the data to be broadcasted in current cycle. At end, RSs reset LI and Ack3 would be given to MS to signal about the completion of data rebroadcast. When there is no update transaction, all RSs broadcast the unblocked data from BD with their indices.

The Ack, Syn signals enforce the synchronization between MS and RSs to disseminate the consistent data. After receiving the Syn2 from the MS, the replica server compares the LI with BI. If there is any overlap, then newly updated data items with their indices would be broadcasted and they would be removed from current broadcast cycle list of BD to avoid duplicated broadcast. Otherwise, the updated data items may previously broadcasted then it rebroadcasts the data items. Since we have used index as control information to identify the data inconsistency, all RSs broadcast the updated data items in the same order to MCs. At any time, if a replica server failures then the MCs can receive the consistent data from any nearest replica on current cell region.

RS Algorithm

```
While (true) {
 For each bcycle set  BD,BI

        If(RS received Syn1 ){
        Set LI=set of Indices of the data  items to
        be updated by MS

         Block them from broadcast
         Send Ack1 to MS.
        }

        If(Update log transaction signal)
        {
        Set UD, execute Update Log        transaction
          Send Ack2 to MS.
        }

        If(RS received Syn2)
          {  For each  x  in LI{
         If (x  є  BI) // x to be Broadcasted  in      current
        Bcycle
           {
        Broadcast D[x]  from UD,LI
          Remove D[x]  from BD,BI
          //To avoide duplicated broadcast
             }
        Else  // x may already broadcasted in previous
        Bcycles
          Rebroadcast D[x] from UD, LI
          }
         LI={}
         Send Ack3 to MS.
           }
       Else  //No Update Log
Broadcast D[x]  from  BD,BI    where  x  not in LI
}//while
```

4.3. Mobile Client Phase

The mobile clients can receive consistent broadcasted data from any nearest available replica server independent of their location and they can send update requests to the MS through uplink channel. Whenever the mobile clients try to update any data item on the MS, the mobile clients have to send index of the data item to be modified with the new value. The mobile clients can get back the updated data item from the broadcast channel. The MS would execute or cancel the update requests of mobile clients based on the result of consistency checks. To save the broadcast bandwidth, the proposed approach is not sending any acknowledgement to the update transactions of mobile clients.

5. Performance Evaluation

As in depth analysis of the performance of CRMB system makes it crystal clear that extended simulation has done for replication environment with various system loads. The proposed scheme is implemented by using Java with the components: one master server, two replica servers, queuing system and the mobile clients. The following factors that help in assessment of effectiveness of the proposed approach are: average throughput for different size of update transactions, throughput for a range of database load, broadcast transaction prolong ratio for the update transactions executed in various intervals, rebroadcast overhead and the stale access rate. The following notations are used in this section,

D : Set of all data items in database
DS : Database Size
d_i : Data Item where i ={1,2,…...DS}, ∀ $d_i \in D$ there is d_i [I,V]
NBC : Number of broadcast cycles to broadcast D
TBC : Time to broadcast a cycle in Seconds
NDT : Network Delay Time in Seconds
THR : Throughput of Update Transaction
QL : Queue Latency
EST : Execution Starting Time of UT
RT : Re-broadcasting Time of updated data items
UTA : Update Transaction Arrival time
NUT : Number of Update transactions executed when broadcasting D
LUT: Length of Update Transaction

By using CRMB approach, the average tuning time (waiting time in broadcast channel) to receive desired data item(x) from the read-only database is calculated as follows:

When the database is **non-updatable**, the average tuning time of mobile clients depend on size of database and number of broadcast cycles. If the desired data item is in the first segment of database,

Average Tuning Time:

$$TT(d_i[x]) = (TBC * (NBC/2)) + NTD \quad \text{where } d_i[x] < DS/2. \quad \textbf{(1)}$$

If the desired data item is in the second portion of database,

$TT (d_i[x]) = (TBC * NBC) + NTD$ where $d_i[x] > DS/2$. **(2)**

The **throughput of update transaction** (THR) is calculated by the proposed system as shown below:

Since the response time of update transaction depends on queue latency and rebroadcast time, the throughput of lengthy update transactions and the update transactions executed in very short interval are higher.

Queue Latency: $QL (UTi) = UTAi – ESTi$ **(3)**
THR(UT$_{i)}$ $= (QL(UTi)+(RTi-ESTi) +NDT)$ **(4)**

Another factor called **Broadcast Transaction Prolong Ratio (PR)** is defined as the ratio of additional time needs to broadcast the updated data items.

$$PR= \left(\sum_{i=1..NUT} (THR(UT_{i}) / (TBC+NTD) * NBC) \right) * 100 \quad \textbf{(5)}$$

In CRMB, the percentage of bandwidth needs to broadcast the updated data items is defined as **Broadcast Overhead Ratio (BO)** of an update transaction which is calculated as follows,

Size of a broadcast cycle (SZ)= $Sizeof (d_i[I][v]) * Len(Bcycle)$ bytes **(6)**

BO= $((Sizeof (d_i[I][v]) * LUT) –$ Sizeof (removed $di[I][v]$ from current broadcast cycle if any) $/SZ) * 100$ **(7)**

5.1. Experimental Setup

In the simulation system, the master server is designed to execute the update transactions via queuing system and the replica servers were modeled to broadcast consistent data. The server update transactions can randomly update any number of data items, while the clients update transactions can update only one data item. In case of dynamic and temporal databases, the cancelled update requests would not be restarted to preserve the absolute consistency level. The following Figure.3 shows the implementation of CRMB. Table.1 shows the sample parameters and their default values. The size of update transaction length indicates the number of data items updated by an update transaction.

Table.1 Model Parameters

Parameters	Default Values
No. of Data items in the Database	1000
Broadcast rate	20 items per
No. of Replica Servers	2
Simulation Time	5000Sec
Server Update Transaction Length	1-15
Clients Update Transaction	1
Update Transaction inter arrival	0.5 Sec- 5sec
No. of Mobile Clients	100

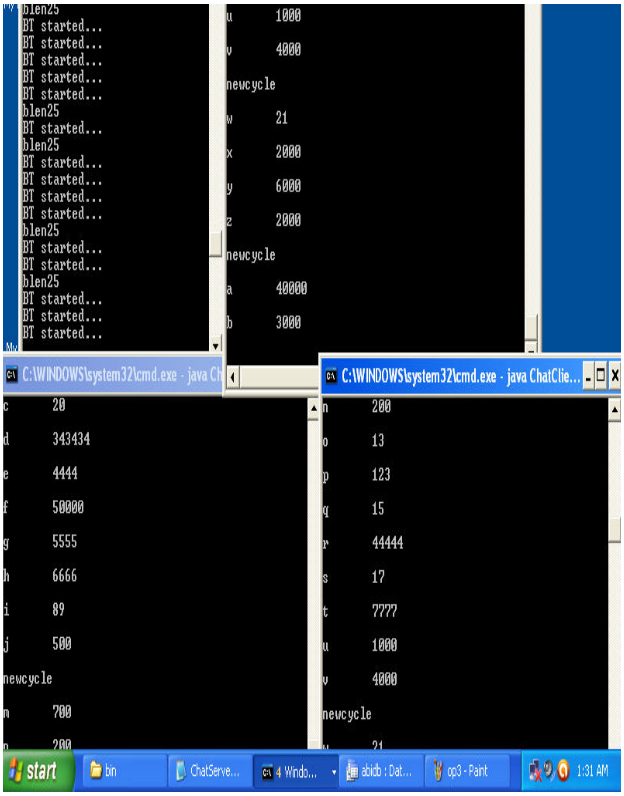

Figure.3 Broadcast Server to the Clients

5.2. Simulation Results

5.2.1. Impact of Server Update Transaction

Length

The impact of different length of update transactions of various database sizes is shown in figure 4. As the proposed approach broadcast all the updated data items instantly, very lengthy update transactions for larger database need high response time. The response time gradually increases with the database size because of the fixed size of broadcast cycles, which lengthens broadcast transaction of bulky databases. However, most of the applications are in need of the update transactions of maximum length of four or five. The proposed approach can yield faultless performance for such lengths with high data consistency, currency and availability.

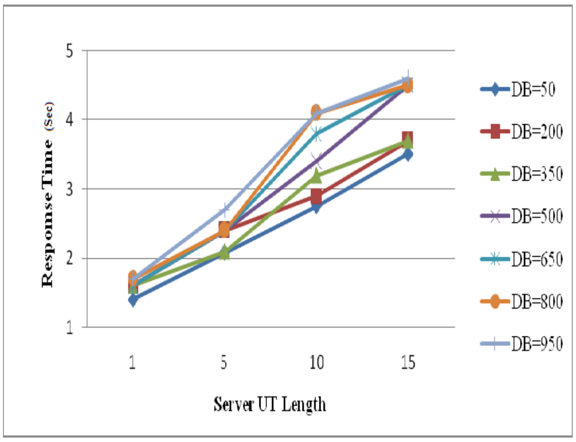

Figure.4 Server Update Transaction Length Vs Response Time (Sec)

5.2.2. Impact of Update Transactions with Short Interval

The frequency of update transactions affects the system performance because it increases the data conflicts and the rebroadcast rate. Hence, the response time is increased for the update transactions executed at short intervals. When the inter-arrival time of update transaction is minimal, the update prolong ratio is high (22%) in the suggested approach as shown in figure 5. Still the rebroadcast overhead and prolong ratio are higher than other methods, the proposed replicated broadcast approach provides other benefits such as consistency, high currency level, data availability through minimum throughput.

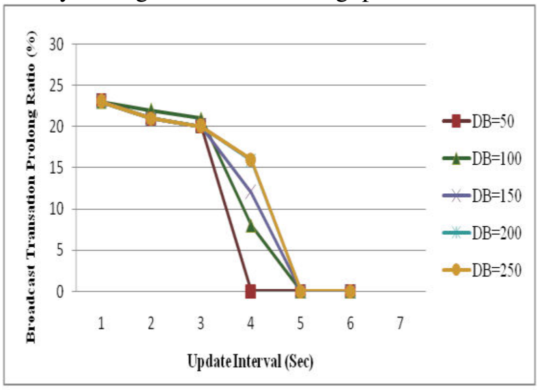

Figure.5 Update Interval Vs Broadcast transaction Prolong Ratio (%)

6. Conclusion and Future Enhancements

The proposed algorithm controls the simultaneous execution of broadcast and update transactions to achieve consistency and currency on replicated database environment using index based broadcast method. To ensure the consistency, currency and high performance on replicated mobile database dissemination, CRMB needs additional logics on MS and RS. All replica servers broadcast the fresh data in the same order in which they were updated in MS. Through the usage of replicated database environment, this algorithm can provide high data accessibility by the lowest amount network communication. Since the proposed algorithm rebroadcasts only the updated data using indices, it avoids the redundant data broadcast and it saves bandwidth requirement of mobile clients by re-broadcasting the updated data item in the current or next immediate broadcast cycle without IR. Though the rebroadcast overhead is higher in CRMB, it can give better performance by proving strict consistency, currency, high data availability and accessibility through minimum size of control information and network communication. But there may be delay for executing multiple clients' update transactions. However, high data availability, accessibility, consistency and currency are the major criteria for data dissemination, several other issues such as disconnection of mobile clients, tuning time to get their desired data and node failure are still remain to discuss.

REFERENCES

[1]Abdelkrim Beloued, JeanMarieGilliot, MariaTeresaSegarra,Franc¸oise Andr´e. Dynamic Data Replication and Consistency in Mobile Environments,2nd International Doctorial Symposium on Middleware '05, November 28-December 2,2005 France.

[2]Aekyung Moon, Han Namgoong, Hyoungsun Kim, Hyun Kim , A data replication based on primary copy for ensuring data consistency in mobile Ad-hoc network, 01/2006, Proceeding of the 2006 International Conference on Communications in Computing, CIC 2006.

[3]Anita Vallur, Mohammed Atiquzzaman, A survey of data replication techniques for mobile ad hoc network databases, The VLDB Journal (2008) 17:1143–1164.

[4]Arkady Zalslavsky,Mohamed Faiz, Bala srinivasan, Amir rashieed, simon lai. Primary Cpy Method and its Modification for Database Replication in Distributed Mobile Computing Environment, proceeding of SRDS '96 15th Symposium on reliable distributed systems.

[5]Ashraf Ahmed, P.D.D.Dominic Azween Abdullah and Hamidah Ibrahim. A New Optimistic Replication Strategy forLarge-scale Mobile Distributed Database Systems, Proceedings of International Journal of Database Management Systems (IJDMS), Vol.2, No.4, November 2010.

[6]Dr.K.ChitraManikandan,Dr.K.Senthamarai Kannan, S.Abirami. Index Based Ordered Broadcast with Status (IOBS) Algorithm for Consistent Data Broadcast, proceedings of First International Conference, CCSEIT 2011-Springer, Tirunelveli, Tamil Nadu, India, September 23-25, 2011, Communications in computer and Information Series , Vol. 204,1st Edition 2011, pp 275-285.

[7]Dr.K.Chithra,R.Varadarajan,Reducing latency through efficient channel allocation methods for multiiversion data dissemination in mobile computing environment, International journal of Internet Technology and Secured transactions, vol.1, nos.3/4,2009.

Volume: 09 Issue: 05 Pages: 3588-3595(2018) ISSN: 0975-0290

[8]Dr.K.ChitraManikandan,S.Abirami,Time Stamping Method For Consistent Data Dissemination To Read Write Mobile Clients, Proceedings of International Conference on Computer and Informatics (ICCCI-2012), Coimbatore, India ,Jan10-12,2012-IEEE, ,Vol2,1 edition 01-03-2012, pp 335-339.

[9]Dr.K.ChitraManikandan,S.Abirami, Efficient algorithm for disseminating consistent data to read –write mobile clients, International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol.2, No.4, July 2012,pp21-25.

[10]EvaggeliaPitoura,PanosK.Chrysanthis,Caching and Replication in Mobile Data Management, IEEE Data Eng. Bull, 13-27, 2007.

[11] J. Gray, P. Helland, P. O' Neil, and D. Shasha, The Dangers of Replication and a Solution. In Proceedings of the ACM SIGMOD Conference, pages 173–182, Montreal, Canada, 1996.

[12]G. Herman, et. al The Data cycle Architecture for Very High Throughput Database Systems Proceedings of the ACM SIGMOD Conference, 1987.

[13]Lam, K.Y., Chan, Edward and Au, Mei-Wai. Broadcast of Consistent Data to Read-Only Transactions from Mobile Clients, Proceedings of 2nd IEEE Workshop on Mobile Computing Systems and Applications, New Orleans, Louisiana, USA, Feb. 1999(3) .

[14]Padmanabhan,Prasana, Managing Data replication in mobile AD-HOC network databases, proceedings of International conference communication, Networking & Broadcasting Computing & Processing (Hardware/Software) Nov 2006,pp-1-20.

[15]Shanmugasundaram,J., Nithrakashyap, A., Sivasankaran, R. and Ramamritham, K., Efficient Concurrency Control for Broadcast Environments, in Proceedings of ACM SIGMOD International Conference on Management of Data, Philadelphia, June 1-3, 1999. (8).

[16]Shiow-Yang Wu,Yu-Tse Chang,An active scheme for mobile data management, proceeding of 6th international conference on advanced systems for advanced applications, April 1999,pp-143-150.

[17] Supeng Leng, Shumao Ou, Guanhua Qiao, Energy harvesting and computing enabled data broadcasting in mobile social networks, Communication and computation cooperation: principles, algorithms and systems,Vol 31, Issue 4 August 2017.

[18]Takahiro Hara. Data Replication Issues in Mobile Ad Hoc Networks, Proceedings of 26th International workshop on Database and Expert Systems Applications (DEXA' 05).

[19]D. B. Terry, M.MTheimer, K. Petersen, A. J. Demers, M. J Spreitzer, and C. H. Hauser. Managing Update Conflicts in Bayou, a Weakly Connected Replicated Storage System. Proceedings of 15th ACM Symposium on Operating Systems Principles, December 1995.

[20]Vahideh Baradaran Sorkhbai,Hadi Shahamfar Detecting appropriate Replication approach considering client and server parameters in mobile environment, Proceedings of IEEE International Conference on Educational Technology and Computer(ICETC),2010.

[21]Weihan Wang and Cristiana Amza. An Optimal Concurrency Control for Optimistic Replication, Proceedings of 29th IEEE International Conference on Distributed Computing Systems,2009.

[22]Yasushi Saito. Consistency Management in Optimistic Replication Algorithm, International Journal of Database Systems , June 15,2001,pp 1-17.

[23]Ziad Itani,Hassan Diab,Hassan Artail,Efficient Pull Based Replication And Synchronization for Mobile Databases, Proceedings of IEEE International Conference ,2005.

[24] Zaixin Lu, Weili Wu, Wei Wayne Li, Efficient scheduling algorithms for on-demand wireless data broadcast, Proceedings of INFOCOM Apr 2016 - The 35th Annual IEEE International Conference on Computer Communications, IEEE, 2016.