ENERGY-EFFICIENT REPLICATION EXTENDED DATABASE STATE MACHINE IN MOBILE AD-HOC NETWORK

Aekyung Moon
Software Robot Research Team, ETRI
Gajung-dong, Yusong-gu, Taejon 305-350, KOREA
akmoon@etri.re.kr

Haengrae Cho
Department of Computer Engineering, Yeungnam University
Kyongsan, Kyongbuk 712-749, KOREA
hrcho@yu.ac.kr

ABSTRACT

A *mobile ad-hoc Network* (MANET) is a collection of mobile servers and clients that can communicate with each other directly via wireless link in the absence of fixed wired infrastructure. MANET needs to consider specially data replication that is traditionally seen as a way to increase the availability, reliability and performance of transaction processing. The reason why is that additional energy for data broadcasting can be consumed to maintain consistency if the database is fully replicated among all mobile servers. In this paper, we propose an eager replication scheme, named E-DRM (*Eager replication extended Database State Machine*) for MANET where both server and client have energy restrictions. That is, E-DRM extends database state machine scheme suitable to MANET with taking energy into consideration. According to E-DRM, tentative transactions are processed locally at single *small mobile host* (SMH) and are forwarded to one of *large mobile hosts* (LMH)s as soon as broadcast cycle starts. After that, LMHs certify tentative transactions as correct through only one message broadcasting. As a result, the proposed scheme can validate all conflicts in LMHs locally; hence, it can improve the performance by reducing additional energy consumed to maintain consistency and the number of message broadcasting.

KEYWORDS

Ubiquitous, Sensor Network, Energy, Replication, DBMS

1. INTRODUCTION

The characteristic of *mobile ad-hoc network* (MANET) is different from that of existing wireless network. A traditional wireless network consists of a fixed network of servers and clients with a collection of mobile clients that move within the geographic area of the network. In this kind of network, servers have unlimited electric energy; hence, related to the issue is the energy consumption of clients. In contrast, MANET is a collection of mobile servers and clients that can communicate with each other directly via wireless link in the absence of fixed wired infrastructure [7]. Consequently, in addition to the issue associated with a traditional wireless network, the energy consumption of server must also be considered in MANET.

In this paper, we propose an energy-efficient eager replication scheme to achieve consistency data replication for MANET. The proposed scheme is focused on how to reduce additional energy consumption and the number of message broadcasting to maintain consistency. Traditionally, data replication is seen as a way to increase the availability, reliability and performance of transaction processing in the area of distributed computing. MANET needs to consider specially data replication [5]. The reason why is that additional energy for data broadcasting can be consumed to maintain consistency if the database is fully replicated among all mobile servers. However, little work has been done to address the issue of both data broadcasting and replication in MANET. Moreover, eager replication has received very little attention in wireless network. The result was that the probability of occurring deadlock is inherently high [6]. Atomic broadcast revealed excellent result lately. Holliday [8] suggested a way to avoid deadlocks, pre-order transactions. Pre-ordering of transactions can be viewed as atomic broadcast [4]. After that, Alonso [9] and Pedone [11] showed an eager replication using atomic broadcast that can resolve the problem of deadlock. Both approaches rely on atomic broadcast to propagate transactions between database servers. However, with atomic commitment messages Alonso's approach may suffer from the overhead of aborted transactions and broadcasting for the result of certification test to all replication nodes. On the other hand, the scheme proposed by Pedone alleviates the need for atomic commitment differently from that of Alonso.

With regard to this viewpoint, we propose eager replication scheme, named E-DRM (*Eager replication extended Database State Machine*) for MANET. That is, E-DRM extends database state machine presented by Pedone suitable to MANET. According to E-DRM, tentative transactions are processed locally at single *small mobile host* (SMH) that has cache to store portions of database and are forwarded to one of *large mobile hosts* (LMH)s that has complete database as soon as broadcast cycle starts. After that, LMH reprocesses and certifies tentative transactions as correct through only single message broadcasting. The proposed scheme validates all conflict in LMHs locally. As a result, it can improve the performance by reducing additional energy consumed to maintain consistency and the number of message broadcasting.

This paper is organized as follows. Section 2 presents the related works. In section 3 the details of our scheme is discussed. The performance analysis is in section 4. Finally, section 5 summarizes the main conclusion of this study.

2. RELATED WORK

Existing replication scheme in wireless network is either eager or lazy replication [6]. The eager replication ensures that any change to copies should happen within the transaction boundaries - i.e. when a transaction commits, all copies have the same value. The lazy replication propagates changes only after the transaction commits, thereby allowing copies to have different values.

Gray [6] showed that in some situations the probability of deadlock is directly proportional to n^3, where n is the number of replica. After that, most of replication schemes assume lazy replication [1,10] and eager replication has received very little attention. Holiday [8] suggested a way to avoid deadlocks, pre-order transactions. Pre-ordering of transactions can be viewed as atomic broadcast. An atomic broadcast primitive enables to send messages to several nodes, with the guarantee that all nodes agree on the set of messages delivered and the order according to which the messages are delivered [4].

The representative schemes that use atomic broadcast are *Replication with Serizability* (SER) [9] and *Database State Machine* (DSM) [11]. The SER, which executes read operations locally and broadcasts only write operations, is achieved by deferring write operations The sender node S broadcasts its deferred write operations of transaction T_i to all nodes. Upon receiving the write operations, the lock manager of receiver nodes including S grants all write locks on T_i atomically. The S node broadcasts an abort message to all nodes if any conflict happened in T_i for ensuring global serializability. The receiver node undoes all write operations and releases all locks on T_i when it receives abort message from the sender node. Therefore, SER may suffer from the message traffic of abort transactions and the overhead of the execution of abort transactions at all receiver nodes except for S node. This is due to the fact that the sender node certifies related transactions as correct and needs atomic commitment message for the result notification of certification test. On the other hand, DSM alleviates the need for atomic commitment differently from SER. The basic idea of DSM is that every receiver node receives, certifies and processes the same sequence of write operations requests. In this case, atomic broadcast constitutes a sufficient order mechanism to implement a DSM. However, DSM may lead to large transaction abort rate because of delayed broadcasting after read and write operations are executed at sender node.

3. EAGER REPLICATION EXTENDED DATABASE STATE MACHINE

The nodes in a MANET can be classified by their capabilities as [5]. SMH is a node with reduced processing, storage, communication, and energy resources. LMH is a node having larger resources. With large capacity

LMH contain the entire database and the database is fully replicated. Therefore, LMHs bear primary responsibility for data broadcasting and satisfying client update transactions as well as read-only. SMHs typically have resources to cache portions of the database. We assume MANET has tentative transactions. Tentative transactions execute update operations on local cached data in SMH.

For simplification, we consider a particular client C_a that sends requests to a SMH on behalf of a transaction T_k. The update on the local data cached in SMH is performed by the tentative transactions T_k. At broadcast cycle, SMH sends commit request to LMH. LMH reprocesses and certifies T_k as correct in broadcast cycle. LMH manages *update information table* (U-TBL) to validate the global serializability. U-TBL registers the updated data item in this broadcast cycle. It consists of [data_id]. When reprocessing T_k, LMH broadcasts information including [*read data set* (RS_k), *write data set* (WS_k)] to all LMHs for certifying tentative transaction T_k. At this time, E-DRM broadcasts information after execution of read operations differently from DSM. When a broadcast cycle starts, the LMH broadcasts the tentative transaction's information.

Step 1: Request Commit Phase: The SMH sends the information of tentative transaction T_k to corresponding LMH. Then state of transaction T_k becomes the executing state as figure 1.

Step 2: Information Broadcasting Phase: The LMH reprocesses read operations of T_k and broadcasts the information [RS_k, WS_k, WO_k] of T_k. The WO_k is write operations of $T_{k.}$

Step 3: Certification Test Phase: Every receiver LMH including sender LMH passes the state of T_k to the committing state and certifies T_k as correct. The certification test takes into account every transaction those conflicts with RS_k of T_k. The LMH validates d_r in case of all $d_r \in RS_k$ in U-TBL. If update of d_r isn't happened before, U-TBL(d_r) is $\varnothing$ in this broadcast cycle.

(1) U-TBL(d_r)=$\varnothing$: This implies that the data d_r has not been updated recently; hence, T_k is correct.

(2) U-TBL(d_r)$\neq \varnothing$: This implies that the data d_r has been updated in this broadcast cycle; hence, read data d_r is invalid. It means that T_k isn't correct if a tentative transaction T_{k-1} is before T_k and $d_r \in (WS_{k-1} \cap RS_k)$.

If the result of the validation of d_r is case (1) then LMH registers [d_w] in all $d_w \in WS_k$ into U-TBL. Then LMH reprocesses WO_k and T_k passes to the committed state. However, if the result is case (2), the LMH decides to abort a transaction T_k. After all, T_k passes to the aborted state. In case that T_k is committed state, write operations WO_k of T_k can be processed the same sequence of request at every LMH for reduction in abort rate. That is, all LMHs can process tentative transactions as same order. As a result data consistency can be ensured. Naturally, the ratio of aborted transactions can be reduced too.

Step 4: Data Broadcasting Phase: LMH broadcasts updated data including the result of related tentative results. SMH accepts notice of the abort or commit of T_k from LMH and sends this result to C_a.

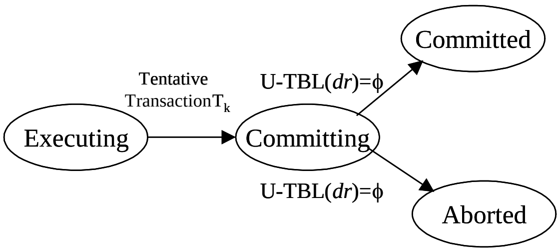

Figure1. The State of Tentative Transaction in LMH

4. ANALYSIS

This section analyzes the performance of SER and E-DRM by experiments. Figure 2 shows the simulation model, which is implemented using the CSIM discrete-event simulation package [12]. The simulation parameters for specifying the resources, overheads of the system, and the settings will be used for experiments. Many of the detailed parameter values are adapted from [2] and [3]. The number of node LMH is changed from 5 to 40. The number of disk per LMH is 1. Each disk has a FIFO queue for I/O requests. The percentage of operations on data residing in main memory is 80 percent. The disk access time is 20 milliseconds. The network manager is implemented as a FIFO server with 10 Mbps bandwidth. The CPU cost to send or to receive a message via the network is modeled as a fixed per-message instruction count plus an additional per-byte instruction increment. The periodic broadcasting time is 10sec. The performance metrics used in the experiments is the response time. The response time is measured as the difference between when a transaction is submitted and when the transaction successfully commits. The time includes any time spent in the queue and time spent due to restarts.

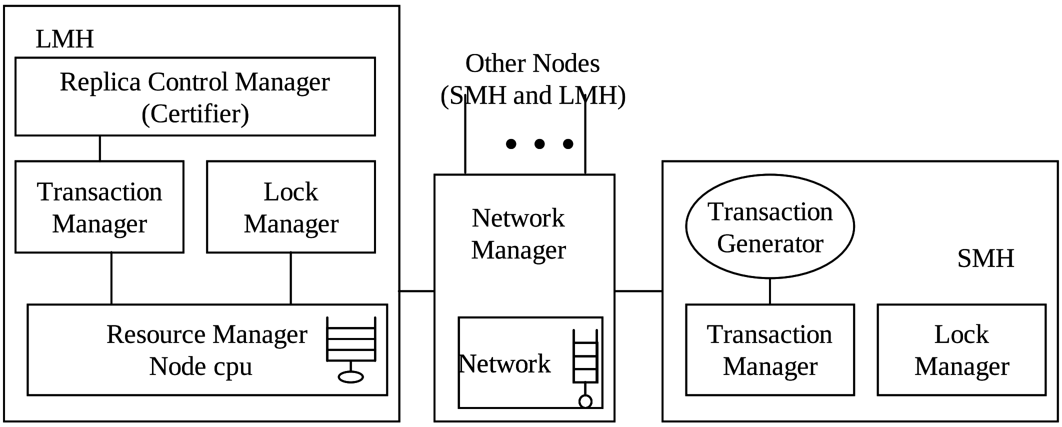

Figure2. Simulation Model

To analyze the performance of these schemes, their evaluation will be examined under high contention workloads [3]. This workload models an application where all nodes have the same data access skew and the degree of data contention is vary high as a result. For every transaction, most of the database accesses go to the hot set. Specifically, 80 percent of every transaction's accesses go to about 20 percent of database. This workload is very important since the high contention application is one of the major applications of distributed computing environment. Figure 3-(1) shows the transaction response time at this workload for various numbers of LMHs, where the write probability was 30 percent. As the number of nodes increase, E-DRM performs better than SER. Note that the performance of E-DRM improves about maximum of 1.7 times when the number of nodes is 40. If there are large the number of nodes, more transactions can be executed concurrently. It leads that the number of transactions accessing the same data is increased, and thus large number of transaction must abort in proportion to data contention. As a result, SER can be increased the overhead due to aborted transactions.

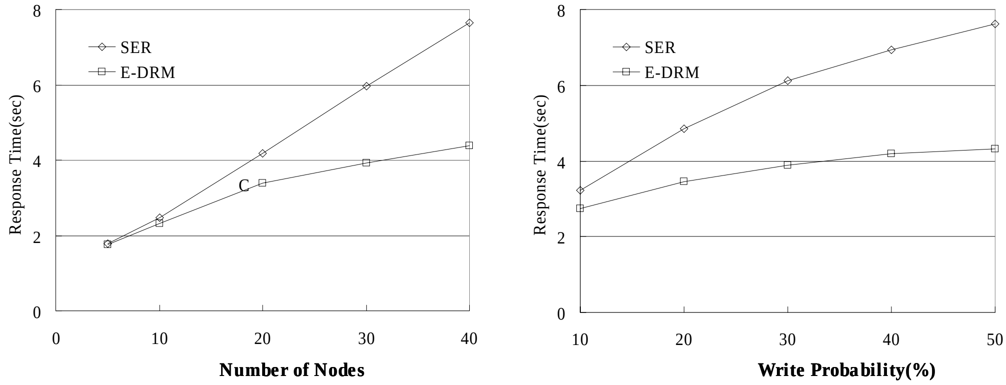

Figure 3.Response Time –(1) Number of Nodes (2) write Probability

Figure 3-(2) shows the response time at this workload when probability of updating a record (WriteOpPct) is varied when the number of node is 30. If the WriteOpPct is low, updated records are few and data contention is negligible. As WriteOpPct is increased, E-DRM outperforms SER. When WriteOpPct is 50

percent, the performance of ERGC improves about 1.8 times better than SER. If WriteOpPct is high, SER suffers from aborted transactions maximum. The probability of data contention is $(1-s^2)k^2N/D$ [13]. s, k, N and D are the probability of read operation, the number of locks, multiprogramming level (number of nodes) and the number of data respectively. This is due to the fact that the higher WriteOpPct rises the higher the rate of data contention rises. As a result, the number of aborted transactions is increased.

5. CONCLUSION

We proposed the eager replication scheme named E-DRM considering energy consumption. If the database is fully replicated among all mobile servers, additional energy in data broadcasting can be consumed to maintain the consistency of databases. E-DRM extends database state machine scheme used in existing distributed computing suitable to MANET with reducing the number of message broadcasting. The basic results obtained from the experiments can be summarized as follows. As the number of nodes is increased, E-DRM exhibits substantial performance improvements about 1.7 times compared to SER in high contention workload. When probability of updating a record is 50 percent, the performance of E-DRM improves about 1.8 times better than SER. This result is very encouraging because (1) E-DRM reduces the overhead of additional commitment messages to notify the results of certification test, and (2) reduces the overhead of aborted transactions. On the other hand, SER has the execution overhead of aborted transaction as well as the message traffic of commitment messages, and thus the higher data contention rises the more the overhead of aborted transactions increases.

REFERENCES

[1]T. Anderson, Y. Breitbart, H.F. Korth and A. Wool, 1998. Replication, Consistency, and Practicality: are These Mutually Exclusive?. *Proc. ACM SIGMOD*, pp. 484-495.

[2]D. Barbará and T. Imielinski, 1994. Sleepers and Workaholics: Caching Strategies in Mobile Environment. *Proc. ACM SIGMOD*, pp. 1-12.

[3]M.Carey, M. Franklin and M. Zaharioudakis, 1994. Fine-Grained Sharing in Page Server DBMS. *Proc. ACM SIGMOD*, pp. 359-370.

[4]K.P. Birman, A. Schiper and P. Stephenson, 1991. Lightweight Casual and Atomic Group Multicast. *ACM Trans.Computer Syst*. Vol. 9, No. 3, pp. 272-314.

[5]L. Fife and L. Gruenwald, 2003. Research Issues for Data Communication in Mobile Ad-Hoc Network Database Systems. *SIGMOD Record*, Vol. 32, No. 2, pp. 42-47.

[6]J. Grey, P. Helland, P.O'Neil and D. Shasha, 1997. The Dangers of Replication and a Solution. *Proc. ACM SIGMOD,* pp. 173-182.

[7]L. Gruenwald, M. Javed and M. Gu, 2002. Energy-Efficient Data Broadcasting in Mobile Ad-Hoc Networks. *Proc. International Database Engineering and Applicaions Symp*., pp. 64-73.

[8]J. Holliday, D. Agrawal and A. Abbdai, 2000. Using Multicast Communication to Reduce Deadlock in Replicated Databases. *Proc. IEEE Symp. Reliable Distributed Syst*., pp. 196-205.

[9]B. Kemme and G. Alonso, 2000. A New Approach to Developing and Implementing Eager Database Replication Protocols. *ACM Trans. Database Syst.,* Vol. 25, No. 3, pp. 333-379.

[10]F. Pedone and R. Guerraoui, 2003. The Database State Machine Approach. *Distribiuted and Parallel Databases*, 14, pp. 71-98.

[11]E. Pacitti, P. Minet and E. Simon, 1999. Fast Algorithms for Maintaining Replica Consistency in Lazy Master Replicated Databases. *Proc. VLDB*, pp. 126-137.

[12] H. Schwtman, 1992. *CSIM User Guide for use with CSIM Revision 16,* MCC.

[13] Y.C. Tay, N. Goodman and R. Suri, 1985. Locking Performance in Centralized Databases, *ACM Trans. Database Syst*., Vol. 10, No. 4, pp. 415-462.