A Proxy-based Uncoordinated Checkpointing Scheme with Pessimistic Message Logging for Mobile Grid Systems

Nomica Imran
Ubiquitous Computing Lab
Dept. of Comp. Eng.
Kyung Hee University
South Korea
nomica@oslab.khu.ac.kr

Imran Rao
NICTA Victoria Labs
Dept. of CSSE
The University of Melbourne
Australia
imran@csse.unimelb.edu.au

Young-Koo Lee*
Ubiquitous Computing Lab
Dept. of Comp. Eng.
Kyung Hee University
South Korea
yklee@khu.ac.kr

Sungyoung Lee
Ubiquitous Computing Lab
Dept. of Comp. Eng.
Kyung Hee University
South Korea
sylee@oslab.khu.ac.kr

ABSTRACT

Due to mobility, energy limitations, and unreliable wireless channels, applications running on mobile devices suffer from faults such as temporary disconnection and data loss. We, therefore, need a fault tolerance mechanism to guarantee their smooth working and performance. In this paper, we present a novel proxy-based uncoordinated checkpointing scheme with pessimistic message logging for efficient fault recovery in mobile Grid system. Simulation results show that this scheme is reliable, efficient and, at the same time, consumes less network traffic.

Categories and Subject Descriptors

C.2 [**Computer-Communication Networks**]: Distributed Systems—*Distributed applications*

General Terms

Reliability, Performance

1. INTRODUCTION

Many researchers have proposed different solutions for fault recovery in mobile computing. These solutions, however, fail to appropriately handle the failures with minimal processing and storage overhead on mobile hosts. Much of the literature on message logging and checkpointing in the past decade has been based on a so-called optimistic approach that places more emphasis on failure-free overhead than recovery efficiency. To overcome these issues we purpose a novel proxy based coordinated checkpointing scheme with pessimistic message logging for fault recovery in mobile Grid systems. The key idea we employ is that MHPs monitor and maintain MH's entire state. It communicates asynchronously with other MHs (through their respective MHPs) and logs the messages that can affect its state. In case of MH's disconnection or failure, the MHP sustains its state of the connection to hide its non-availability from the environment. Incurring the pessimistic message-logging (PML) reduces the storage overhead of the uncoordinated checkpointing on MSS and enhances the overall efficiency of the recovery line calculation process. Moreover, due to proxies, there is no overhead of piggybacking the sequence number with every message exchanged between MH and its corresponding proxy, as is in [2].

2. PROPOSED SCHEME

2.1 Background Information

We consider a mobile Grid system consisting of mobile hosts (MHs), mobile service stations (MSSs) and Grid resources. MHs are connected with Grid infrastructure through a mobile to Grid middleware MAGi [1]. And finally, MSS is defined as a process that resides on resourceful MAGi and communicates with MHs within its range. The static MSS provides various services to support a mobile host. When a MH goes outside the region of a MSS, known as cell, it connects to the other MSS within that range. The MHP asynchronously stores checkpoints on its stable storage, and hence subsequently participates in the process of recovery line calculation without direct involvement of the MH. As MHP is a static host and resides on the resourceful MSS, this delegation results in better performance and reliability as compared to existing techniques. We also introduce a message sequence number free failure recovery scheme as compare to [2] to mitigate the storage overhead.

2.2 System Model

We model our system as a collection of region based mobile cells. If there are n mobile hosts in a cell, then it can be modeled as

$$\{(MH_p, MH_p^a), MSS_a\}\forall p \in (1, 2, 3, \dots, n)$$

When a new MH_p enters in the region of a mobile service station MSS_a, it creates a new mobile host proxy MH_p^a for

*Corresponding author

HPDC'07, June 25–29, 2007, Monterey, California, USA.
ACM 978-1-59593-673-8/07/0006.

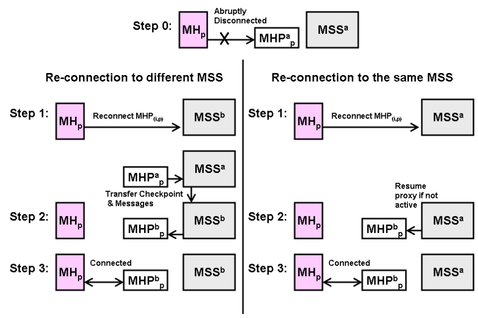


Figure 1: Reconnection and Hand-off management

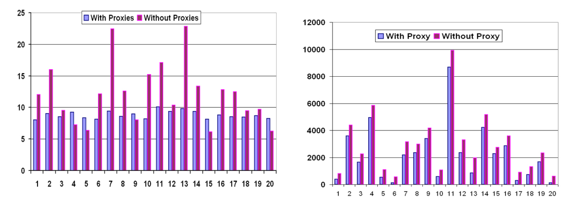
MH_p and send its address $curr\_proxy_p^a$ to MH_p. MH_p uses $curr\_proxy_p^a$ address pointer of its current mobile host proxy MH_p^a. $curr\_proxy_p^a$ is also used in mobility management to locate the last proxy MH_p has paired with. MH_p^a maintains a message queue $msg\_Q_p^a$, checkpoint data structure ckp_p^j which includes process states and function stack. Message queue $msg\_Q_p^a$ is a FIFO based message queue which records all the messages received since its last checkpoint by MH_p through MH_p^a. Theses messages are stored in order of reception and, hence, do not need to be numbered. We define ckp_p^j as the jth checkpoint taken by MH_p. We also define three control messages MH_p may send for connection management with MSS. Message *msg_join* is sent for a new connection with a MSS, *msg_disc* is for a graceful disconnection from its current MSS and *msg_recon* is for a reconnection after a graceful disconnection or failure with the same or a different MSS. Note the difference between *msg_join* and *msg_recon*. *msg_recon* is meant to restart its processing from where it left whereas *msg_join* is used to make a new connection. Moreover, *msg_recon* and *msg_disc* messages sent by MH will piggyback the address pointer of the last MH_p it corresponded with.

2.3 Message Logging and Checkpointing

Suppose that MH_p enters in the vicinity of MSS_a and sends *msg_join* message for a new connection. Upon receipt of this message, MSS_a will create a new mobile host proxy MH_p^a along with $msg\_Q_p^a$. MH_p^a logs all the messages sent to MH_p in $msg\_Q_p^a$ in the order of their reception. After a periodic but arbitrary time interval $\triangle_p$, MH_p takes a local checkpoint ckp_p^j of its processes, which includes process states, function stack, and sends them to its current proxy MH_p^a (Figure reffig:mh). The value of $\triangle_p$ can be adjusted by the administrator and depends upon the network availability. Moreover every MH_p may have its own checkpoint frequency. After receiving ckp_p^j, MH_p^a stores it to its personal stable storage which is readily available to it. Subsequently, MH_p^a will run the garbage collection routine defined in Garbage Collection. The Garbage Collection routine is also executed when MH_p willingly leaves MSS_a by sending a disconnection message *msg_disc*. After sending a *msg_disc* message, if MH_p wants to connect again to same or a different MSS, it needs to send *msg_join* message and it will be handled as a new MH. Figure reffig:mh illustrates the re-connection of MH_p with the same or a new MSS.

3. SIMULATION RESULTS

We simulated our model to evaluate the performance effects due to the inclusion of mobile host-proxies in the systems. We chose the communication cost and the time to recover as simulation metrics for 20 MHs with varying network bandwidths, number of messages exchanged and hand-offs.



(a) Recover efficiency from failure with and without using proxies

(b) Communication cost comparison with and without using proxies

Figure 2: Simulation Results

As shown in Figure 2(a), in our scheme there is a constant overhead of creating a proxy but such overhead is negligible. Cases 4, 5, 15 and 20 show that when a MH is not moving from its home cell and the number of message exchanged are also low, the system with the MHP is outperformed by the system without the MHP. In these cases, the role of MHP and MSS is almost the same and the time taken to create the MHP results in an overhead and gives poor recovery time.

In our scheme, due to the existence of MHP, we do not need to numerate messages as proposed by [2]. And hence there is less communication cost as shown in Figure 2(b).

4. CONCLUSION AND FUTURE WORK

In this paper we propose a mobile host proxies (MHPs) based uncoordinated checkpointing scheme. This scheme takes storage and processing overhead from low-power mobile hosts and delegates to their respective proxies. Our simulation results indicate that inclusion of mobile host proxies significantly improves the performance of checkpointing process, especially for more wandering mobile hosts. In future we plan to investigate the performance and storage overheads of the proposed scheme for data and computation intensive mobile grid applications.

5. ACKNOWLEDGMENTS

This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITFSIP (IT Foreign Specialist Inviting Program) supervised by the IITA (Institute of Information Technology Advancement).

6. REFERENCES

[1] A. S. et al. Magi - mobile access to grid infrastructure: Bringing the gifts of grid to mobile computing. In *Proceedings of NODe/GSEM*, pages 311–322, 2005.

[2] T. Park, N. Woo, and H. Yeom. An efficient recovery scheme for mobile computing environments. In *Proceedings of the 8th International Conference on Parallel and Distributed Systems*, pages 53–60, 2001.