

A Review of Checkpointing Strategies for Mobile Networks

Anjin Xiong, Guoqiong Liao, Guoqiang Di

School of Information Technology

Jiangxi University of Finance and Economics

Nanchang, China

e-mail: xiaosanhill@126.com, liaoguoqiong@163.com,

dylydgq@21.cn.com

Jiali Xia

School of Software and Communication Engineering

Jiangxi University of Finance and Economics

Nanchang, China

e-mail: xiajl65824@263.net

Abstract—Checkpoints are used to store the specific states of processes in stable storage. When distributed systems fail, the systems can recover to a global consistency state from specific checkpoints rather than the beginning states of all involved processes. There are many kinds of checkpointing strategies suggested for distributed systems. However, the existing checkpointing protocols proposed for the traditional distributed systems are not suitable for mobile networks due to the unique characteristics such as low storage capacity, limited bandwidth, frequent disconnection, etc. Therefore, some researchers have proposed many new checkpointing strategies for mobile networks. This paper intends to give a summary of them.

Keywords- *checkpointing; mobile computing; ad hoc network*

I. INTRODUCTION

Checkpoint technologies are proposed for improving the fault tolerance ability of distributed systems primitively. A checkpoint is a record of certain state of a process. When a distributed system fails, it can be recovered to a global consistent state by the checkpoints stored in the stable storages of all involved processes. For a general checkpointing strategy, it mainly takes recovery overhead and storage overhead into account.

According to the relationship among checkpoints, the checkpointing technologies can be divided into two categories: synchronous checkpointing [1,2] and asynchronous checkpointing [3,4]. What is more, the asynchronous checkpointing could be further classified into blocking [1] and unblocking checkpointing [2] based on whether the execution of a process can be interrupted. There are two ways to trigger a checkpoint operation: the period of a timer is coming [1] or a process receives a certain amount of messages [5,6].

As a new kind of distributed network, mobile networks have been widely applied to different kinds of application situations. But the MHs in mobile environment are much liable to fail. The failures of MHs can be divided to two types. The first one is called hard failures including all irreparable failures such as the falling, breaking, or lost of MH. The second one is called soft failures such as power down, temporal software failures. Therefore, the checkpoint strategies should be able to tolerate these two kinds of failures. However, the mobile networks have brought many new challenges to the checkpointing:

- The storage spaces of mobile hosts (MH) are limited, so the storage spaces occupied by the checkpoints should be cleaned up regularly.
- The wireless bandwidths of mobile networks are limited, so the messages generated by checkpoints should be reduced.
- The disconnection is frequent in mobile networks, so it should decrease the effect to the consistency of checkpoints due to disconnection as far as possible.
- The locations of MHs are changeable, so the checkpoint methods should consider location consistency of MHs.

As a special kind of mobile networks, the mobile Ad hoc network is getting more and more attention. Besides the features mentioned above, it has also following unique characteristics to be considered for checkpointing [7].

- Lack of support of fixed network. In regular mobile network, the checkpoint information of MHs is mostly stored in the mobile support stations (MSSs) in fixed networks because of the unreliable storage of MHs. However, there are no any fixed hosts like MSSs in mobile Ad hoc network. Hence, the existing checkpointing methods in mobile computing environment depending on MSS are not suitable for this kind of network anymore.
- Multi-hops routing. Because of lack of support of fixed networks, the messages among MHs in the network are delivered by wireless network completely, that is, the communication among MHs in different clusters needs to be transmitted through multiple intermediate nodes. It will increase the number of wireless messages and prolong the time of checkpointing.

Therefore, the checkpointing methods for traditional distributed systems cannot be applied to the mobile networks very well. Aiming at above challenges, some researchers proposed a lot of new checkpointing methods for the mobile networks. This paper intends to present a review of the checkpointing strategies in the mobile networks. The rest of this paper is organized as follows. In section 2, the checkpointing strategies for regular mobile networks are summarized. Section 3 describes the checkpointing strategies for mobile Ad hoc networks. Finally, the paper is concluded.

II. CHECKPOINTING STRATEGIES FOR REGULAR MOBILE NETWORKS

A. Checkpointing Strategies Considering QoS of Wireless Networks

Neves and Fuchs [8] put forward an adaptive checkpointing strategy which considers the quality of service (QoS) of wireless networks. The strategy proposes the new concept of soft checkpoints and hard checkpoints. The soft checkpoints are stored in the MHs, while the hard checkpoints are stored in the stable storages of the hosts like MSSs in fixed networks. Once taking maxSoft soft checkpoints, a process will take a hard checkpoint. It means that the MH in which the process runs needs to transmit the soft checkpoints to its MSS by wireless channels. The value of the maxSoft can be changed with the QoS adaptively. As the QoS becomes bad, the maxSoft should be larger, that is, the processes will take more soft checkpoints before taking a hard checkpoint, since the MH should reduce wireless transmission of checkpoint messages if the QoS is bad. The advantage of the method is that the value of maxSoft can be changed with QoS of the network adaptively. But the MHs will store so many soft checkpoints and the checkpoints will occupy much storage space as the QoS becomes worse.

B. Checkpointing Strategies Based on Mobile Agents

The mobile agent [9,10] is used to manage the checkpoints and message logs of MHs. When one MH leaves the cell controlled by the MSS, the mobile agent will take charge of transmitting the checkpoints and message logs from MSSs to MHs. Cao et al. [9] put forward a checkpointing and rollback strategy for wide-area distributed applications using mobile agents. In the strategy, the mobile agents can execute at one host, and then are dispatched to another. When one process needs to recover, it could get the related checkpoints and message logs from the mobile agents.

Chowdhury and Neogy [10] proposed a checkpointing algorithm based on mobile agents. In the algorithm, the mobile agents are divided to checkpointing agents (CPA) and log agents (LGA). Each MSS has one LGA. The LGA is used to store the message logs of MHs, while the CPA usually stays in the MSS with which the MH keeps connection currently. By the time the MH moves into another wireless cell, the CPA will send the id of the next target MSS to the current LGA and so the LGA could track the location of the CPA. The CPA stores the list of MSSs which contain the message logs of MHs as a mobility profile. When the MH takes new checkpoint, the previous message logs and checkpoint will be cleaned up and the mobility profile will be initialized to the MSS that the current checkpoint stays. If the MH fails or disconnects with the network, the CPA goes back to the initial location of mobility profile. This algorithm could reduce the communication overhead of MSS. However, it brings the augment of total device overhead and increases the delay of wireless transmission obviously.

C. Checkpointing Strategies Considering the Mobility of MHs

Considering location movement of MHs, George and Chen [11] proposed a checkpointing method based on movement and logs. In mobile computing environment, the area controlled by an MSS is called a cell. The process that an MH moves from one cell to another is called a handoff. In the method, an MH will take a checkpoint when it crosses a fixed number of handoff and uses the cp_seq to indicate the sequence number of checkpoints. Whenever an MH enters a new cell, it will record the ID of its MSS and writes the ID into the logs (i.e., MSSlogs). In order to record the ID of the MSS, the MH uses the cp_loc to indicate the ID of the MSS. Thus, when the MH fails, it will connect with one MSS. Then it sends the cp_seq and cp_loc to the MSS. On the one hand, the MSS will require the MSS related to cp_loc to send the checkpoint of MH. Moreover, it will obtain the logs from the MSSs which belong to the MSSlogs. Finally, it sends the checkpoint and logs to the MH. The limitation of the method is that the recovery time becomes very long if the failed MH has multiple handoffs.

D. Checkpointing Strategies Based on Transmission of Checkpoints

Considering the overhead of wireless transmission, Cao and Singhal [12] present a coordinated checkpointing algorithm using mutable checkpoint stored in MHs. In the method, the process appends current checkpoint sequence number (CSN) with the message. The process which receives the message will compare the CSN of message sender that is known by itself with the CSN carried by the message. If the CSN carried by the message is larger and the process has sent messages after the last checkpoint, the process will take mutable checkpoint and then receives the message. Otherwise, the process receives the message directly. When one process initializes a checkpointing process, the mutable checkpoints will be treated as the tentative checkpoint. The shortcoming of this strategy is that the number of mutable checkpoints induced by messages is large and most of these mutable checkpoints may become useless checkpoints.

There are also some other kinds of checkpointing schemes suggested for the mobile networks. Cao and Singhal [13] prove the impossibility of min-process non-blocking checkpointing firstly, and then proposed an efficient checkpointing algorithm for mobile computing systems. In the proposed method, the MSS is in charge of initializing the checkpointing procedure. It uses an $N \times N$ dependency matrix to find out all processes on which the initial process depends on. Park and Yeom [14] propose an asynchronous recovery scheme for mobile computing systems combining the optimistic message logging and checkpointing. The suggested strategy consists of two parts. The first part is designed for the systems with reliable MSSs, and the MSSs take charge of storing the checkpoints and message logs for MHs and recording the mobility of the MHs. The second part is designed for the systems with unreliable MSSs, and the MSSs is responsible for flushing the checkpoints and logs to

the stable storage, and recording the dependency relations among the states of processes.

III. CHECKPOINTING STRATEGIES FOR MOBILE AD HOC NETWORKS

A. Checkpointing Strategies Based on Channel Allocation

Based on the channel allocation, Meng et al. [15] put forward a checkpointing strategy for Ad hoc networks with cluster structure. In the strategy, the timing scheme is adopted and the wireless channels are assumed to be FIFO order. There are two checkpoint beacon intervals in one checkpoint. The cluster head (CH) in each cluster is responsible for initializing checkpointing, and sending an initial checkpoint request to all MHs in the cluster. If the channel can be allocated in the first beacon interval as the MH receives the request, the MH will take a checkpoint in the second beacon interval. Otherwise, the MH takes a checkpoint in the first beacon interval. The advantage of the method is that it designs a coordinated checkpointing algorithm by combining the channel allocation of mobile Ad hoc network very well. But the checkpointing operations are too frequent, resulting in heavy storage overhead.

B. Checkpointing Strategies Considering Delay of Wireless Communication

Considering the delay of wireless communication, some checkpointing strategies for mobile Ad hoc networks assume that the wireless channels are not in FIFO order. The stagger method is to be designed for delaying to record the current states of processes.

Jaggi and Singh [16] propose a staggered checkpointing and recovery method for mobile Ad hoc networks based cluster structure. In the method, the CH is also the initiator of checkpointing. At first, the CH sends a `take_chkpt` message to other clusters after taking a checkpoint. For any CH, if it receives the `take_chkpt` message for the first time, it will record the sender of `take_chkpt` as the PARENT and records the initiator of checkpoint as the LEADER. Otherwise, it sends a DENY message to the sender of `take_chkpt`. If the initiator is not the LEADER, the CH will add it to its boundary-set. After receiving the `take_chkpt` message, the CH will forward the message to the members in its own cluster one by one, and the members will take a checkpoint in a staggered manner. After taking the checkpoints, the member will send replies to its CH. If there is a CH which doesn't receive replies from all members after a period of time, it will delete the members without reporting from the set of active nodes. When all members in the cluster complete the checkpoints, the CH will send an ACK message to its upper CH according to the boundary-set. Finally, if the CH is the initial CH, it will find out the other initiators by the boundary-set and sends the `chkpt_taken` message to them. After receiving all replies from the other initiators, the initial CH will finish the checkpointing procedure and send the `chkpt_taken` message to its child nodes. Hereinto, the child nodes are obtained from a spanning tree built by the PARENT structure. The main

shortcoming of the method is that there are too many messages to be transmitted among the initiators.

Besides above methods, there are some other checkpointing methods proposed for Ad hoc networks by improving existing checkpointing strategies of the traditional distributed systems directly. Tuli and Kumar [17] put forward an asynchronous checkpointing strategy. In the method, in order to reduce the occupation of storage space in MHs, the message logs of MHs are stored in the cluster heads. Parveen and Kumar [18] present a synchronous checkpointing strategy. It assumes that the executions of the processes are deterministic, namely the same results will be gotten when the same messages sequence are executed twice from the same starting point.

IV. SUMMARY

The paper has summarized the checkpointing strategies for the mobile networks. In the regular mobile networks, the checkpointing strategies are mainly designed for reducing the storage and communication overhead of checkpointing, and can yield good performance in practice. In addition, the MSS is introduced to store checkpoints and message logs for MHs. In the mobile Ad hoc networks, because the wireless bandwidth is the most important factor, the checkpointing strategies for mobile Ad hoc networks mainly consider the augments which impact the wireless bandwidth such as channel allocation and delay of wireless communication.

However, for the checkpointing strategies proposed for the regular mobile networks, though they use the MSSs to help deal with checkpoint, the overhead of wireless message transmission is still high. Moreover, it is still difficult to process the failure due to disconnection. For the checkpointing strategies suggested for the mobile Ad hoc networks, they cannot handle the multi-hop routing and changeable topological structure very well. What is more, these strategies are just adapted to the unicast routing environment. Hence, it is still necessary to design new checkpointing methods supporting the unique features fully for the mobile networks.

ACKNOWLEDGMENT

This work is supported by Natural Science Foundation of China (No. 60863016, 61262009), Natural Science Foundation of Jiangxi, China (No. 2008GQS0019, 20122BAB201032), Science Foundation of Jiangxi Provincial Department of Education, China (No. GJJ10694, GJJ12259), Building Project of Superior Innovation Team of Science and Technology of Jiangxi Province, China (No. 20113BCB24008).

REFERENCES

- [1] K. M. Chandy, L. Lamport, "Distributed snapshots: Determining global state of distributed systems," *ACM Transaction on Computing Systems*, Vol. 3, No. 1, pp. 63-75, February 1985.
- [2] Y. Tamir, C. H. Sequin, "Error recovery in multi-computers using global checkpoints," *Proceedings of the 13th International Conference on Parallel Processing*, Bellaire, MI, pp. 32-41, 1984.

- [3] R. E. Strom, S. Yemini, "Optimistic recovery in distributed systems," *ACM Transactions on Computer Systems*, Vol. 3, No. 1, pp. 204-226, August 1985.
- [4] B. Bhargava and S. R. Lian, "Independent checkpointing and concurrent rollback for recovery in distributed systems-An optimistic approach," *Proceedings of 17th IEEE Symposium on Reliable Distributed Systems*, Columbus, OH, pp. 3-12, 1988.
- [5] Y. Wang, W.K. Fuchs, "Lazy checkpoint coordination for bounding rollback propagation," *Proceedings of the 12th Symposium on Reliable Distributed Systems*, Princeton, NJ, pp. 78-85, 1983.
- [6] R. Baldoni, J-M. H elary, A. Mostefaoui and M. Raynal, "A communication-induced checkpointing protocol that ensures rollback-dependency trackability," *Proceedings of the International Symposium on Fault-Tolerant-Computing Systems*, Seattle, WA, pp. 68-77, June 1997.
- [7] S. K. Soni, T. C. Aseri, "A review of current multicast routing protocol of mobile Ad hoc network," *Second International Conference on Computer Modeling and Simulation*, Sanya, China, pp. 207-211, 2010.
- [8] N. Neves, W. K. Fuchs, "Adaptive recovery for mobile environments," *Communications of ACM*, Vol. 40, No. 1, pp. 68-74, January 1997.
- [9] J. Cao, G. H. Chan, T. S. Dillon, W. Jia, "Checkpointing and rollback of wide-area distributed applications using mobile agents," *Proceedings of the 15th International Parallel and Distributed Processing Symposium*, San Francisco, California, pp. 1-6, 2001.
- [10] C. Chowdhury, S. Neogy, "Checkpointing using mobile agents for mobile computing system," *International Journal of Recent Trends in Engineering*, Vol. 1, No. 2, pp. 26-29, 2009.
- [11] S. E. George, I. Chen, "Movement based checkpointing and logging for failure recovery of database applications in mobile environments," *Distrib parallel Databases*, Vol. 23, No. 3, pp. 189-205, 2008.
- [12] G. Cao, M. Singhal, "Mutable checkpoints: A new checkpointing approach for mobile computing systems," *IEEE Transaction on Parallel and Distributed Systems*, Vol. 12, No. 2, pp. 157-172, 2001.
- [13] G. Cao, M. Singhal, "On the impossibility of min-process non-blocking checkpointing and an efficient checkpointing algorithm for mobile computing systems," *Proceedings of International Conference on Parallel Processing*, Minneapolis, MN, pp. 37-44, August 1998.
- [14] T. Park, H. Y. Yeom, "An asynchronous recovery scheme based on optimistic message logging for mobile computing systems," *Proceedings of the 20th International Conference on Distributed Computing Systems*, Taipei, China, pp. 436-443, April 2000.
- [15] C. Men, Z. Xu, X. Li, "An efficient checkpointing and rollback recovery scheme for cluster-based multi-channel Ad-hoc wireless networks," *2008 International Symposium on Parallel and Distributed Processing with Applications*, Sydney, Australia, pp. 371-378, 2008.
- [16] P. K. Jaggi, A. K. Singh, "Staggered checkpointing and recovery in cluster based mobile Ad hoc networks," *Advances in Parallel Distributed Computing Communications in Computer and Information Science*, Vol. 203, pp. 122-134, 2011.
- [17] R. Tuli, P. Kumar, "Asynchronous checkpointing and optimistic message logging for mobile Ad hoc networks," *International Journal of Advanced Computer Science and Applications*, Vol. 2, No. 10, pp. 70-76, 2011.
- [18] Praveen, P. Kumar, "A min-process checkpointing protocol for deterministic mobile Ad hoc networks," *International Journal of Engineering Science and Technology*, Vol. 3, No. 9, pp. 6923-6930, 2011.