6S-2

Consistent Checkpoint Protocol for Mobile Ad hoc Networks*

Masakazu Ono and Hiroaki Higaki ‡
Tokyo Denki University §

1 Introduction

Recently, wireless LANs composed of mobile computers such as notebook PCs within which a wireless communication protocol such as IEEE802.11 is implemented are so highly developed and widely available. Examples of applications in an ad hoc network environment which are consist of only mobile computers are temporally configured networks in conventions and for disaster rescue in an area where it is difficult to set base stations, sensor networks and so on. In an ad hoc network, mission-critical applications are also supported as in a conventional wired networks and checkpoint recovery is one of the important methods for achieving fault-tolerant environment. However, in traditional checkpoint protocols for fixed networks, it is assumed that all computers have some stable storages. In addition, network bandwidth is assumed to be so wide that inconsistent messages are detected by synchronization between sender and receiver computers with low communication overhead. Hence, it is difficult to apply these conventional protocols in a mobile ad hoc network since required cost for achieving stable storages in unstable mobile computers and overhead for achieving synchronization among mobile computers which are connected unreliable, unstable and narrow wireless communication links. Therefore, in this paper, we propose a novel checkpoint protocol which achieves a stable storage by cooperation among multiple mobile computers and avoiding communication overhead for synchronization between sender and receiver mobile computers.

2 Related Works

An ad hoc network $\mathcal{N} = (\mathcal{V}, \mathcal{E})$ is a network with a set $\mathcal{V}$ of mobile computers and a set $\mathcal{E}$ of bi-directional wireless communication links $\langle M_i, M_j \rangle$ between two mobile computers M_i and M_j between which messages are exchanged directly. Generally in a computer network both wired and wireless, a global checkpoint $C_{\mathcal{V}}$ which is a set of local checkpoints c_i each of which is taken by a mobile computer $M_i \in \mathcal{V}$ is consistent if the following condition is satisfied [1].

[Definition]

1) A message m which is transmitted from a source mobile computer M_s to a destination one M_d is a lost message for a global checkpoint $C_{\mathcal{V}}$ if *Send*(m) precedes to c_s in M_s and c_d precedes to *Receive*(m) in M_d. Here, *Send*() and *Receive*() are message sending and receipt events in an application layer, respectively.
2) A message m is a lost message for $C_{\mathcal{V}}$ if c_s precedes to *Send*(m) in M_s and *Receive*(m) precedes to c_d in M_d.
3) A global checkpoint $C_{\mathcal{V}}$ is consistent if there are no orphan messages and all lost messages are retransmitted after recovery. □

A checkpoint protocol in [3] is designed to store lost messages into a message log in a destination computer M_d for retransmission in recovery according to this definition of a consistent global checkpoint.

Almost all conventional checkpoint protocols are designed on an assumption that each lost and orphan message is detected in a destination computer M_d of the message. Hence, virtual synchronization among all computers in a system is required. However, in an ad hoc network, higher synchronization overhead is required than in a conventional wired network due to narrower bandwidth of wireless communication links, reduction of transmission power of wireless signal, contention and collision caused by multiple access and hidden terminal problem and longer transmission delay in multihop transmission.

3 Checkpoint Protocol

Our proposed protocol is designed under the following assumptions:

1) Any pair of mobile computers in an ad hoc network are mutually reachable with multi-hop message transmission during processing of the checkpoint protocol.
2) A wireless communication link between two mobile computers is dynamically connected and disconnected due to their mobility.
3) Each mobile computer keeps a list of neighbor mobile computers within its message transmission range up-to-date.
4) All wireless communication links between two neighboring mobile computers are bi-directional and communication along them is half-duplex.

Now, we show an outline of our checkpoint protocol. Any mobile computer in an ad hoc network initiates a checkpoint protocol. Transmission of request for taking local checkpoints and virtual synchronization of them are realized by flooding [2] of copies of a checkpoint request message *CReq* as show in Figure 1.

On receipt of *CReq*, a mobile computer M_i takes its local checkpoint c_i by storing its state information S_i and broadcasts copies of *CReq* to all its neighbor mobile computers within its message transmission range. By applying this method, all mobile computers in an ad hoc network take their local checkpoints due to assumption 1). Here, since it is difficult for each mobile computer alone to implement stable storage for storing S_i, M_i asks neighbor mobile computers of M_i to store S_i for achieving stable storage. Since each mobile computer broadcasts *CReq* after taking its local checkpoint, i.e. achieving its local state information,

*MANET における一貫性のあるチェックポイントプロトコルの実現手法
†小野 真和 桧垣 博章
‡{masa,hig}@higlab.k.dendai.ac.jp
§東京電機大学理工学部

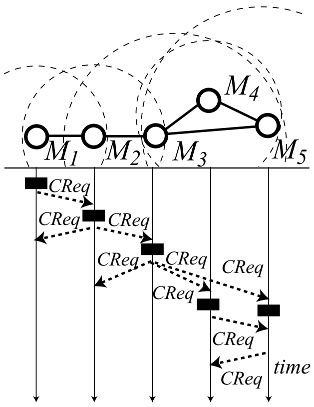

Figure 1: Checkpoint Protocol.

no additional messages are required to be transmitted for the state information by pigging back it to *CReq*.

Here, messages sent, forwarded and received during processing of a checkpoint protocol might be lost or orphan messages. Hence, each lost message is required to be stored in a certain mobile computer and to be retransmitted after recovery in order to keep a global state of an ad hoc network consistent. On the other hand, it is impossible to keep a global state in an ad hoc network consistent if there are some orphan messages which are not surely retransmitted after recovery. Hence, an orphan message should be avoided in an usual message transmission protocol.

According to relationship between local checkpoint and transmission of messages, an intermediate mobile computer along a message transmission route detects a message m_l which is possible to be a lost message. If this detection is realized only in a destination mobile computer M_d, M_d has already broadcasts *CReq* to all mobile computers within a message transmission range of M_d before *Receive*(m_l) where M_d receives m_l. In this case, m_l has to be broadcasted to all mobile computers within a message transmission range of M_d in order to be stored into these mobile computers. However, M_d is not able to decide number of messages which should be broadcasted and terminate execution of a checkpoint protocol.

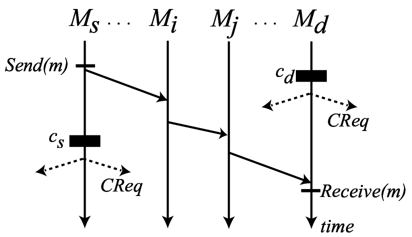

Figure 2: Delayed Lost Message Detection Problem.

Therefore, in our proposed checkpoint protocol, a message m_l which is possible to be a lost message (we call m_l a *pseudo lost message*) is detected by an intermediate mobile computer along a message transmission route of m_l before it sends *CReq* as shown in Figure 3.

In Figure 3(a), m_l is received before a local checkpoint c_j and has not yet forwarded before c_j. Hence, M_j detects m_l to be a pseudo lost message and m_l is pigged back to *CReq* for retransmission in recovery.

On the other hand, in Figure 3(b), before receipt of m_l, M_j has taken a local checkpoint c_j and broadcasted *CReq*. Since m_l is sent by M_i before a local checkpoint c_i, M_i detects m_l to be a pseudo lost message by receipt of an ack for m_l with information that m_l is received after c_j which is represented by * in Figure 3(b).

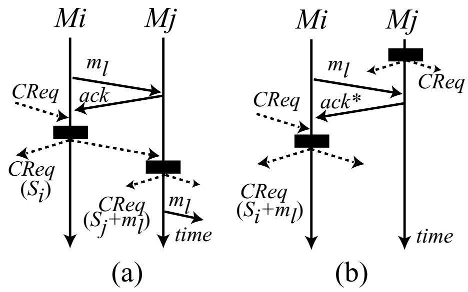

Figure 3: Lost Message Detection in Intermediate Mobile Computer.

An intermediate mobile computer only forwards a message that is possible to be an orphan message. By delaying acceptance of the message in an application layer in a destination mobile computer, i.e. *Receive*() event, it is achieved to avoid orphan messages.

4 Concluding Remarks

This paper has proposed a novel ad hoc checkpoint protocol in which detection of lost messages is not based on end-to-end but hop-by-hop, i.e. an intermediate mobile computer detects and stores pseudo lost messages which are possible to be lost messages. By introducing this method, each mobile computer is required to broadcast a checkpoint request message only once in a checkpoint procedure. That is, lower communication and synchronization overhead is required. If wireless communication links between mobile computers are connected and/or disconnected during a checkpoint procedure due to mobility, there may exist lost and/or orphan messages which are not detected in this protocol. In future work, we design an extended protocol to achieve consist global checkpoints even with such messages.

References

[1] Chandy, K.M. and Lamport, L., "Distributed Snapshots: Determining Global States of Distributed Systems," ACM Trans. on Computer Systems, Vol. 3, No. 1, pp. 63–75 (1985).

[2] Corson, M.S. and Ephremides, A., "A Distributed Routing Algorithm for Mobile Wireless Networks," ACM Journal of Wireless Networks, Vol. 1, No. 1, pp. 61–81 (1995).

[3] Elnozahy, E.N. and Zwaenepoel, W., "On the Use and Implementation of Message Logging," Proc. of the Fault-Tolerant Computing Symposium, pp. 298–307(1994).

[4] Ono, M. and Higaki, H., "Checkpoint Protocol for Mobile Ad-hoc Networks," IPSJ SIG MBL, Vol. 2003, No. 93, pp. 91–96 (2003).