IJCSC International Journal of Computer Science and Communication Vol. 3, No. 1, January-June 2012, pp. 211-213

DISTRIBUTED COMPUTING IN MOBILE ENVIRONMENT

Apekshit Sharma[1], Chandrakant Sharma[2]

[1,2]Assistant Professor, IBB Group of Colleges, Ranpur, Kota, Rajasthan, *E-mail: apxit21@gmail.com E-mail: cksmca@gmail.com*

ABSTRACT

This paper considers central issues of distributed computing in a mobile environment. Its aim is to light on the first brick of a common view for mobile systems.

We propose a distributed algorithm for solving the h-out of-k mutual exclusion problem for ad hoc mobile networks. The h-out of-k mutual exclusion problem is also known us the *h*-out of-*k* resource allocution problem.

Keywords: Distributed, Mobile, Network, Token

1. INTRODUCTION

The distributed computing area includes henceforth mobile systems (i.e., cellular and ad-hoc networks, peer-to-peer systems, mobile virtual reality and cooperative robotics). The classic models and solutions are not adapted to mobile world's hence new models; new problems and obviously new solutions should be designed. This paper advocates for the creation of a common view for mobile worlds based on the similarities between them.

We continue our investigation with the problems solved in mobile systems starting with specific problems. Then we present solutions provided so far for some fundamental problems in distributed computing (i.e., (*k*–) mutual exclusion and leader election).

There are several papers proposed to solve mutual exclusion related problems for ad hoc mobile networks by adapting a token-based mutual exclusion algorithm named RL (Reverse Link) algorithm. The k-RL algorithm and the G-RL algorithm are two examples. The k-RL algorithm is proposed to solve the k-mutual exclusion problem; and the G-RL algorithm, the group-mutual exclusion problem. The RL algorithm assumes that there is a unique token initially and utilizes the partial reversal to maintain a token oriented DAG (directed acyclic graph). A node should gain the token along the DAG to access the shared resource. The proposed algorithm in this paper is also adapted from the RL algorithm. The RL algorithm and its adaptations are sensitive to link forming and link breaking.

Distributed computing is a method of computer processing in which different parts of a program run simultaneously on two or more computers that are communicating with each other over a network. There are numerous technologies and standards used to construct distributed computations, including some which are specially designed and optimized for that purpose, such as Remote Procedure Calls (RPC).

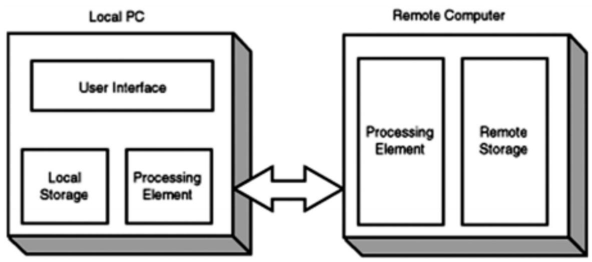

Figure 1: The Distributed Programming Model

1.1 Objective

The main goal of a distributed computing system is to connect users and resources in a transparent, open, and scalable way. Ideally this arrangement is drastically more fault tolerant and more powerful than many combinations of stand-alone computer systems.

2. DISTRIBUTED COMPUTING IN A MOBILE ENVIRONMENT

Its aim is to light on the first brick of a common view for mobile systems. We pool together mobile systems and analyze them from different angles including architecture and computability aspects. We show that mobile systems (i.e., cellular systems, ad hoc networks, peer-to-peer systems, virtual reality systems or cooperative robotics) are basically confronted with the same problems; hence it is useless to maintain the actual "misleading" barriers. Moreover, in the new model, the fundamental problems of distributed computing (i.e., (*k*–)mutual exclusion, leader election or group communication) should find appropriate specifications.

2.1 Introduction

At present, the distributed computing environment consists of a collection of dissimilar (heterogeneous) computers connected through the fixed-wired networks such as the Ethernet.

2.2 Mobile Worlds

In this, we show that the Mobile System appellation is not restricted to wireless mobile systems, but includes several research areas.

2.2.1 Mobile Systems

In this, we present characteristics of some popular mobile systems.

2.2.1.1 Cellular Networks

In Cellular Networks, a base station manages an area of world called cell. Mobile hosts are connected with the base station of the cell to which they belong. Moreover, the mobile hosts cannot establish a direct connection between them; they rely on a base station to communicate.

2.2.1.2 Ad-Hoc Networks

Ad-Hoc Networks differ from Cellular Networks in not relying on any specific infrastructure. Each mobile host can send messages to the nodes within its transmission range. Moreover, it can route messages on behalf of others.

2.2.1.3 Peer-to-Peer Systems

A Peer-to-Peer system is a dynamic and scalable set of peers that distributes the cost of sharing data. Each peer can join or leave the system at any moment and can communicate with any other peer under the only hypothesis that the two peers are aware of each other.

2.2.1.4 Shared Virtual Reality Systems

In general, a Virtual World means a technology for moving through and interacting with a three-dimensional computer-generated environment such that the experience is perceived to be real.

3. PRELIMINARIES

In a token-based mutual exclusion algorithm, named RL (Reverse Link), for ad hoc networks is proposed. The RL algorithm takes the following assumptions, which we also take in this paper:

1. The nodes have unique node identifiers.
2. Node failures do not occur.
3. Communication links are bidirectional and FIFO.
4. A link-level protocol ensures that each node is aware of the set of nodes with which it can currently directly communicate by providing indications of link formations and failures.
5. Incipient link failures are detectable.
6. Message delays obey the triangle inequality (i.e., messages that travel 1 hop will be received before messages sent at the same time that travel more than 1 hop).
7. Partitions of the network do not occur.

The RL algorithm also assumes that there is a unique token initially and utilizes the partial reversal technique to maintain a token oriented DAG (directed acyclic graph). In the RL algorithm, when a node wishes to access the shared resource, it sends a request message along one of the communication link.

Now we present the scenario for the h-out of k-mutual exclusion problem. Consider an ad hoc network consisting of n nodes and k shared resources. Nodes are assumed to cycle through a non-critical section (NCS), an entry section (ES), and a critical section (CS). A node i can access the shared resource only within the critical section. Every time when a node i wishes to access h, $1 \leq h \leq k$, shared resources, node i moves from its NCS to the ES, waiting for entering the CS. The h-out of-k mutual exclusion problem is concerned with how to design an algorithm satisfying the following properties:

Mutual Exclusion:

No more than k resources can be accessed concurrently.

Bounded Delav:

If a node enters the ES to request h, $1 \leq h \leq k$, resources, then it eventually enters the CS.

4. THE PROPOSED SOLUTION

In this section, we propose a distributed algorithm to solve the h-out of k-mutual exclusion problem for ad hoc mobile networks. The algorithm is assumed to execute in a system consisting of n nodes and k shared resources. Nodes are labeled as 0, 1, . . ., n-l. We assume there is an unique token held by node 0 initially. The variables used in the algorithm for node i are listed below. Note that subscripts "i" are included when needed.

State: Indicates whether node i is in the *ES*, *CS*, or NCS state. Initially, state = NCS.

N: The set of all nodes (neighbors) in direct wireless contact with node i. Initially, N contains all neighbors of node i.

Height: A triplet (h_1, h_2, i) representing the height of node i. Links are considered to be directed from higher-height nodes toward lower-height nodes, based on lexicographic ordering. For example, if the height of node 1, height, is (2, 3, 1) and the height of node 2, height, is (2,2,2), then $height_1 > height_2$ and the link would be directed from node 1 to node 2. Initially, $height_0$ = (0, 0, 0), and $height_j$, $j \neq 0$, is initialized so that the directed links form a DAG where each node has a directed path to node 0.

hVector: An array of triplets representing node is view of height of node j, $j \in N$. Initially, hVector[j] = height of node j. From node i's viewpoint, the link between i and j is incoming to node i if hVector[j] > $height_i$, and outgoing from node i if hVector[j] < $height_i$.

next: Indicates the location of the token from node i's viewpoint. When node i holds the token, next = i, otherwise next is the node on an outgoing link. Initially, next = 0 if $i = 0$, and next is an outgoing neighbor otherwise.

tokenHolder: A boolean variable indicating whether or not node i holds the token. If node i holds the token, tokenHolder is set to true. It is set to false, otherwise.

Q: A queue which contains requests of neighbors. Initially, $Q = \emptyset$. Operations on Q include enqueue, dequeue, and delete. The enqueue operation inserts an item at the rear of Q, and the dequeue operation returns and removes the item at the front of Q, and the delete operation removes a specified item from Q, regardless of its location.

5. CONCLUSION

- We propose a distributed algorithm for solving the h-out of-k mutual exclusion problem for ad hoc mobile networks.
- The proposed algorithm is sensitive to link forming and link breaking and thus is suitable for ad hoc mobile networks.
- This paper pointed out similarities among common mobile systems (i.e., cellular networks, ad hoc networks, peer-to-peer systems, reality virtual systems and cooperative robots).

REFERENCES

[1] H. Karl and A. Willig, "A Short Survey of Wireless Sensor Networks", *Technical Report TKN-03-018, Telecommunication Networks Group, Technical University Berlin*, October 2003.

[2] Y. Li, C. Chen and Y. Song, "Real-time QoS Support in Wireless Sensor Networks: A Survey", in the *Proceedings of 7th IFAC IntConf on Fieldbuses & Networks in Industrial & Embedded Systems Toulouse*, France, Nov 2007.

[3] P. Rezayat, M. Mahdavi, M. Ghasemzadeh, M. AghaSarram, "A Novel Real-Time Power Aware Routing Protocol in Wireless Sensor Networks", *IJCSNS International Journal of Computer Science and Network Security*, **10(4)**, April 2010.

[4] H.J. Korber, H. Wattar, and G. Scholl, "Modular Wireless Real-time Sensor/actuator Network for Factory Automation Applications", *IEEE Transact Ions on Industrial Informatics*, **3(2)**, pp. 111-118, 2007.

[5] C. Lu, B. Blum and T. Abdelzaher, "RAP: A Real-time Communication Architecture for Large-Scale Wireless Sensor Networks", *in the Proceedings of the Eighth IEEE Real-Time and Embedded Technology and Applications Symposium*, 2002.

[6] T. He, J. Stankovic and C. Lu , "SPEED: A Stateless Protocol for Real-time Communication in Sensor Networks", *in the Proceedings of International Conference on Distributed Computing Systems*, Providence, RI, May 2003.

[7] E. Felembane, C. Lee and E. Ekici, "MMSPEED: Multipath Multi-SPEED Protocol for QoS Guarantee of Reliability and Timeliness in Wireless Sensor Networks", *IEEE Transactions on Mobile Computing*, **5(6)**, pp. 738-754, 2006.