Mobile Cluster Computing and Timeliness Issues

Haihong Zheng, Rajkumar Buyya[1], and Sourav Bhattacharya
Dept. of Computer Science and Engineering
Arizona State University
Tempe, AZ 85287-5406, USA
Email: *haihongz@asu.edu, rajkumar@ieee.org, sourav@asu.edu*

Keywords: Cellular Network, Cluster Computing, Handover, Hypercube Topology, Mobile Cluster Computing, Mobile IP, Multicast, Rerouting, Timeliness, Triangle Mesh.

Edited by: Marcin Paprzycki

Received: December 15, 1998 **Revised:** February 25, 1999 **Accepted:** March 1, 1999

With the rapid advancement and extensive deployment of cluster computing and mobile communication, the integration of these two technologies has become feasible and lead to the emergence of a new paradigm called mobile cluster computing (MCC). Among the issues that need to be addressed before MCC can become a reality, the timeliness issue is an important one, especially when mobile nodes within a computing cluster migrate from one cell to another cell in a cellular wireless network. In this paper, we first define and analyze the potential application environment of mobile cluster computing. We also present a generic architecture of a mobile cluster computer and several potential research issues of mobile cluster computing. In the rest of this paper, we focus on the timeliness issue of routing and multicast when handover occurs, along with several solution approaches based on different system architectures.

1 Introduction

During the past decade many different computer systems supporting high performance computing have emerged. Among several common systems, clusters[2] have become increasingly popular to prototype, debug and run parallel applications [1]. Since individual workstations have become increasingly powerful, and the communication bandwidth between them is increasing and latency is decreasing, clusters can provide similar (or sometimes better) performance reliability as well as fault tolerance as the traditional mainframes or supercomputers.

In addition, we observe that, as a result of the rapid development of mobile wireless networking system, users can access information across distributed sites and exploit the capacities of the global network at any time without regard to the location or mobility of the end units. This suggests that in addition to stationary nodes, mobile nodes will enter the cluster computing arena and result is the emergence of a new paradigm called "Mobile Cluster Computing" (MCC).

Another boost to our proposition for MCC is the recent emergence of mobile processors from industry dominant microprocessor vendors [2]. They include Intel Mobile Pentium II [3], Intel Celeron [4], and Mobile AMD-K6-2 [5] processors.

The mobile processors run at a lower voltage than desktop processors and operate within the thermal envelope of today's notebook designs. The primary advantage of the lower voltage, lower power mobile processors, is extended system battery life.

These mobile processors are compatible with existing software and offer leading-edge 3D and multimedia performance; in near future, they aim to provide support for high-performance notebook computing.

[1]Currently associated with the School of Computer Science and Software Engineering at Monash University, Melbourne, Australia.

[2]Cluster is a collection of interconnected computers working together as a single system.

Although, due to the transmission latency and poor reliability of wireless media, mobile nodes may not be the best choice to participate in a cluster providing a high-performance computation facility for large-scale and grand-challenge applications, there are many applications that will benefit from clustering regardless. Further, this scenario will change as technology becomes mature. In these applications several of its participating nodes may need to be mobile. Examples include oil rig sensors, sensors and monitors in earth quake detection/prediction, marketing representatives travelling all over the country/world with laptops and sales data feeding in, disaster management systems, battlefield command systems, SWAT teams, and stock market wizards. A few sample applications of MCC include the following:

- A scientific, nomadic environment. For example, scientists stationed at several different cities may gather seismic data and employ a parallel algorithm that predicts future seismic activities.

- Embedded military supercomputing applications in which fault tolerance is a concern. For example, tanks, trucks, planes, etc. may be connected via a wireless network. They can gather data, send it to other nodes in the network, and use distributed algorithms to make decisions based on that data.

- Intelligent incast or multicast in which strict time deadlines are used to steer immediate actions. For example, consider the study of a weather phenomenon such as tornadoes. Several mobile nodes may gather data, send it to other nodes and use the data to make predictions about the path of the tornado.

Applications of this nature can be executed on a single system, but the data sources are inherently at distant mobile locations and thus necessitate the need for sharing mobile resources. Even in a normal environment mobile computers may be used extensively. The idle CPU cycles of these mobile nodes can be used to process whole or part of the input or output of a large scale application in cooperation with other mobile or stationary nodes. Therefore, mobile cluster computing is expected to play an important role in the modern computing era and mobile network systems.

Furthermore, in case of mobile cluster computing there is no need to invest in a new backbone infrastructure. The existing wireless network infrastructure (such as a cellular network system, PCS, satellite communication system, wireless LAN) can act as a communication backbone for mobile cluster computing. Among several wireless network configurations, a cellular structure is the most commonly used one, where the wireless service area is physically partitioned into different cells. Regardless of whether communication is cellular/stellite-based, they are all based on the wireless media and provide communication environment to the mobile users. Therefore, the infrastructure of mobile cluster computing can be built based on the combination and collaboration of the general cluster computing and the wireless network.

Contribution: In this paper, we define and analyze the application environments for mobile cluster computing. We have raised the potential research issues in mobile cluster computing, among them the timeliness issue in mobile cluster computing, especially during handover events, has been identified and addressed in detail. The contributions of this paper include block-based route optimisation which focuses on timeliness issue in rerouting during handover, and several multicast tree reconstruction algorithms that address the timeliness issue in multicasting during handover.

2 A Mobile Cluster Computer and its Architecture

A cluster generally refers to two or more computers (nodes) connected together with each node having the "strong sense"[3] of cluster membership [1]. A mobile cluster, in addition, consists of mobile and stationary nodes (see Figure 1). The nodes of a mobile cluster communicate either using a wireless network (e.g., cellular network) and/or high speed wired network system. It does not matter where the nodes are and what kind of backbone is used to support the

[3]It referes to the appearance of a collection of independent nodes as a single unified resource and of course, all nodes must aim towards this.

nodes, these nodes can communicate and collaborate with each other just as in a general computing cluster. That is, a mobile cluster consists of both mobile and non-mobile nodes interconnected through either physical interconnect or wireless network and work in a coordinated manner transparently sharing workload among themselves. In addition, a mechanism for automatic detection of node status–whether it is static or mobile–and communication via either physical or wireless network is appropriately supported.

The following are the basic components of mobile cluster computers:

- Multiple High Performance Computers (PCs, Workstations, or SMPs)
- State-of-the-art Operating Systems (Layered or Micro-kernel based)
- High Performance Networks/Switches (such as Gigabit Ethernet and Myrinet)
- Network Interfaces Cards (NICs) (e.g., wireless NIC)
- Wireless Network Infrastructure (e.g., cellular network system)
  - Hardware (such as Base station, MTSO, PSTN)
  - Operating System Kernel (such as mobility management, channel resource management, registration)
- Fast Communication Protocols and Services (such as Active and Fast Messages)
- Mobile Communication Protocols like Mobile IP
- Cluster Middleware (Single System Image (SSI) and System Availability Infrastructure)
  - Hardware (such as Digital Memory Channel, hardware DSM, and SMP techniques)
  - Operating System Kernel or Gluing Layer (such as Solaris MC and GLUnix)
  - Applications and Subsystems
    - Applications (such as system management tools and electronic forms)
    - Runtime Systems (such as software DSM and parallel file system)
    - Resource Management Systems (such as LSF and CODINE)
- Parallel Programming Environments and Tools (such as compilers, PVM, and MPI)
- Applications
  - Sequential
  - Parallel or Distributed

The network interface hardware acts as a communication processor and is responsible for transmitting and receiving packets of data between cluster nodes via a network/switch which could be ATM, fast Ethernet, wireless network, etc. When a wireless network is involved to provide communication methods between nodes within a cluster, cluster computing research should consider a lot of issues that the characteristics of mobility in a wireless network bring up. For example, when the mobile node is an Internet node assigned with an IP address, mobile IP, one of the communication protocols, may need to be included to provide a seamless connection across the Internet when the mobile node is roaming.

Communication software offers the means of fast and reliable data communication among cluster nodes and to the outside world. Often, clusters with a special network/switch like Myrinet use communication protocols such as active messages for fast communication among its nodes. They potentially bypass the operating system and thus remove the critical communication overheads providing direct user-level access to the network interface.

The cluster nodes can work collectively, as an integrated computing resource, or they can operate as individual computers. The cluster middleware is responsible for offering an illusion of a unified system image (single system image) and availability out of a collection of independent but interconnected computers.

Programming environments can offer portable, efficient, and easy-to-use tools for development of applications. They include message passing libraries, debuggers, and profilers. It should be noted that clusters could be used for the execution of sequential or parallel applications.

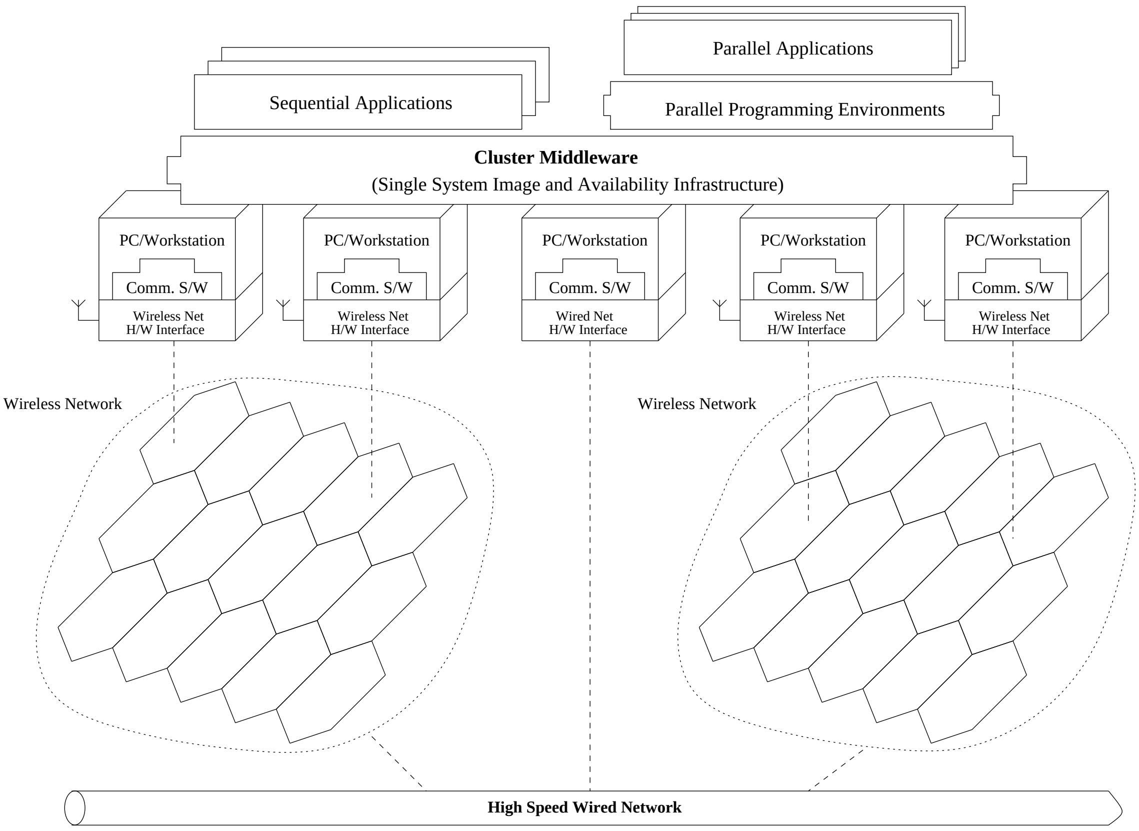

Figure 1: Mobile Cluster Computing Generic Architecture.

2.1 Infrastructure of Mobile Network

The infrastructure of mobile cluster computing mainly consists of conventional computing clusters and mobile networks. Mobile networks (wireless network along with wired network) provide connectivity and communication methods between nodes within a computing cluster, on which cluster computing relies at the higher layer. There are many implementation technologies of mobile networks as shown in Figure 2.

At the physical and MAC layers in OSI model, a non-exhaustive list of wireless networks is as follows.

- Satellite Communication
- Cellular Networks
- PCS
- Wireless LAN
- Microwave Communication
- High-speed Laser Links

For example, a cellular network system physically partitions the service area of wireless networks into different contiguous cells. Cells have the connotation of geographical area served by a base station. Base station consists of a transmitter and two receivers per channel, a controller, an antenna system, and data links to the cellular office. The base station acts as the user-to-MTSO interface. The Mobile Telephone Switching Office (MTSO) is the physical provider of connections from the cellular radio through the base station to the local exchange carrier. The connection can be on either a landline or a microwave radio system between the points.

Another example is wireless LAN which provides wireless connection for mobile nodes within a small area, such as a building. Currently, a lot of commercial products of wireless LAN have emerged, some of which can reach the transmission rate of 100Mbps. The following list is several examples of some wireless LAN implementation.

- http://www.radiolan.com/
- http://www.wilan.com/
- http://www.ccsinc.net/
- http://www.kcnet.com/dceclan/

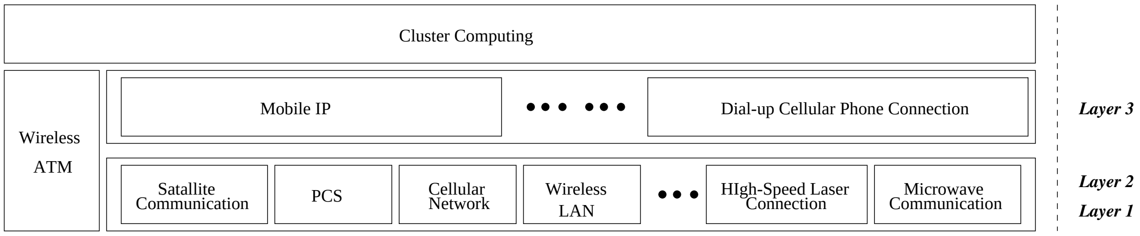

Figure 2: Infrastructure of Mobile Networks.

– http://www.dataequip.com/

At the network layer, a lot of mechanisms have been proposed or implemented to route the message throughout the entire network system even when the mobile node is roaming around. Such as wireless ATM, dial-up cellular phone connection, mobile IP, etc. For example, the IETF solution to route message for a mobile Internet host is mobile IP. The current research status and some implementation of mobile IP can be found at the following web-sites.

– National University of Singapore, Mobile IP for Linux - http://mip.ee.nus.sg/

– SUNY, Binghamton - http://anchor.cs.binghamton.edu/~mobileip

– Stanford - http://mosquitonet.Stanford.EDU/software/mip.html

– http://www.cs-ipv6.lancs.ac.uk/MobileIP

– http://www.ikv.de/products/roamin.html

Based on the services provided by the network layer, nodes (no matter mobile or stationary) could communicate and collaborate with each other and process the large-scale computing.

In summary, the wireless network infrastructure provides a communication scheme to every node in a computing cluster in the same way as wired network does. However, due to the introduction of mobility into cluster computing, a lot of new issues come up.

3 Research Issues

Similar to the mobile cellular system, the characteristics of mobility brings up a number of new research topics to mobile cluster computing.

There are a large number of research issues existing in mobile cluster computing, including the following:

– how to balance the long latency of wireless media and high speed fixed network and perform synchronisation;

– how to provide a complete transparency to users without the knowledge of the underlying system which includes both mobile nodes and stationary nodes;

– how to manage the resources and process the scheduling among different nodes to maximise their throughput.

All of these issues are very critical to the performance of mobile cluster computing and further research on these issues is needed. In this paper, we concentrate on the timeliness issue in mobile cluster computing.

One of the most important issues in mobile cluster computing is the timeliness of handovers that occur in the cellular network system. (The set of operations performed when a mobile node moves from one geographical cell to another adjacent cell is called *handover* [6]). When handover takes place, not only the channel resource occupied by the node should be managed to maintain the connectivity, but also the information sent to or from the mobile node should be rerouted to the new cell to which the mobile node is heading. In addition, if a mobile node is within a multicast application when a handover happens, the entire multicast tree [8] may need to be reconstructed. If the time spent on the resource reallocation, rerouting and its corresponding optimisation, and rebuilding the multicast tree is too long, the information sent to or from the migrating node may be lost and this may result in the serious failure of the entire cluster application.

Therefore, within a computing cluster, timeliness becomes an important issue when a mobile node switches from one cell to another.

4 Research Focus: Timeliness Issue in Mobile Cluster Computing

Timeliness is a critical issue in many computing and communication applications, especially in real-time systems. A real-time system is one in which the correctness of the system depends not only on the logical results, but also on the time at which those results are produced. Messages transmitted in such systems must be received by a deadline or they are lost. Such a real-time deadline, i.e., timeliness issue, is a key component of the QoS (Quality of Service) requirement. However, even if the system does not have the real-time requirement, timeliness issue is still an important component to provide a high QoS to users. For example, in a TCP/IP network, if an end-to-end acknowledgement doesn't reach the source node (e.g., due to network congestion) before the timer expires, the source node will resend the message which would increase the network load and thus deteriorate the network performance. Therefore, the timeliness issue not only exists in real-time systems but also in a normal system where timing is an important issue.

Similarly, the timeliness of an operation is an important issue in MCC under several circumstances. A few of them are identified and discussed below.

4.1 Resource Management

When a mobile node moves from one cell to another cell, the channel used in the old cell may not be reusable in the new cell due to co-channel or adjacent channel interference or low signal strength leading to node isolation from the rest of the cluster. If a new channel that needs to be used has not been allocated to the node within a short time period, the call may be dropped by the user due to the long waiting-time in a cellular telephone system, or, the messages transmitted will be delayed, resulting in retransmission in a mobile data communication system. If such a delay occurs in a time-critical system, the message may not be able to meet the real-time deadline and could get lost, thus leading to the failure of an entire application [9]. The system resources need to be managed (using checkpointing and migration techniques) to handle sudden unavailability of resources.

4.2 Topology Management

Timeliness is also an important issue in network topology management. Various types of logical topologies have been proposed, such as a *ring topology* where each node has exactly two neighbours, a 2-d mesh [7] which is a planar structure that has nodes arranged in a grid of rows and columns, a *tree topology* which has a regular and hierarchical structure of node levels which creates a tree appearance by having each lower level contain more nodes than the previous level, etc. When a mobile node roams and leaves its original position, the pre-defined topology is destroyed and the fragmented topology has to be reconstructed by selecting a new node as the alternative to the migrated one. The time spent on such reconstruction should not be too long, otherwise neither the cluster computing nor the normal network communication would show satisfactory performance.

4.3 Routing

In a mobile network system with mobile IP protocol, any mobile node is allowed to move about, changing its point of attachment to the Internet, while continuing to be identified by its home IP address. Corresponding nodes sending IP datagrams to a mobile node send them to the mobile node's home address by forcing all datagrams for the mobile node to be routed through its home agent. Thus, datagrams to the mobile node are often routed along paths that are significantly longer than optimal. Route optimisation, an extension of mobile IP, provides a means for nodes to bypass the possibly lengthy route to and from that mobile node's home agent by tunnelling their datagrams directly to the foreign agent. However, when a mobile node rapidly or frequently moves from one cell to its neighbouring cell, route optimisation would be processed too frequently. Since the binding information transfer and the computation of an optimal route at the corresponding node and all intermediate nodes are all time consuming processes, timeliness becomes an important issue in such an environment.

4.4 Multicast

Multicasting is an important paradigm of end-to-end communication, where simultaneous transmission of messages are required from a source to a group of destinations. To route these messages, a multicast tree is required to replicate the data available at the root node of the tree and forward the data along various branches leading to destinations at leaf nodes of the tree. However, in a mobile network system, when handover occurs, the multicast tree would be changed by constructing a new route and deleting the original unused route. Since the time spent on the multicasting tree reconstruction may become significant if many mobile nodes are involved in the multicast tree, the timeliness issue should be considered carefully for each multicast tree reconstruction algorithm.

4.5 Message Interaction Pattern between Different Activities

In a mobile cluster computing system, the nodes in the cluster communicate with each other over networks. Different nodes are responsible for different tasks and have different message interaction patterns. When one or several nodes are waiting for the message sent by a mobile node that happens to have a longer processing time than expected due to handover or some other time consuming activities, the normal operation of these nodes has to be paused until the message from the mobile node becomes available. Such blocking may significantly deteriorate the performance of the entire cluster computing application. Thus the timeliness issue also appears to be an important issue in a message interactive cluster computing environment.

4.6 Synchronisation of Cooperative Activities

In most cluster computing applications, the results from individual tasks distributed among several nodes will eventually be collected and processed by one node, which then generates the final result. If one or more nodes delay the assigned task and do not provide the result in time due to unexpected events, the final result cannot be generated within a stipulated time. It might lead to failure in a real-time application or reprocessing due to non-synchronisation of the results from other delayed tasks.

5 State-of-the-Art

A lot of research has been conducted in the area of mobile message rerouting based on different system architectures. To successfully reroute a message for the mobile node which is roaming on the Internet, mobile IP [12] is most popularly used. With mobile IP, when a mobile node migrates from one access point to another, the home network of the mobile node will transfer the message between the mobile node and the correspondent node via the foreign network. Thus, for every message meant for the mobile node, it is required to be routed through the home network. This route is efficient when the agent in the home network resides along or near the best route between the correspondent node and the mobile node. Since it is possible for the home agent to be far away from the best or the optimal route between the mobile node and the correspondent node, a route optimisation procedure that would aim to compute a direct and efficient route between the mobile node and the correspondent node is suggested in the route optimisation draft [13] to improve mobile IP. In [14] the operation of mobile computers using IPv6 which enables mobile computers to cache the home network and a care-of address, is specified. The correspondent node then can compute a direct optimum route to the foreign network (care-of address).

A large amount of research has been conducted in real-time multicast. Dijkstra's shortest path algorithm [15] and the Steiner tree generation problem [16] employed for the delay optimisation and the cost optimisation constraints in the real-time multicast can produce traffic and tree height minimised real-time multicast trees. Some heuristics for the Steiner tree problem have been developed that take polynomial time and produce new optimal results [17]. In [18] the KMB algorithm works under the assumption that a network is abstracted to a complete graph consisting of edges that represent the shortest paths between the source node and the destination nodes. The dynamic update of the tree if destination nodes join or leave the tree occasionally is examined in [16]. The dynamic algorithm proposed for the multipoint problem satisfies the bandwidth constraints based on the minimum spanning tree algorithms proposed for the Steiner tree problem. In [19] the optimisation on both traffic cost and delay is discussed; however, the authors assume that the cost and delay functions are identical. A source-based multicast algorithm that can set the variable delay bounds on destinations and can handle variants of network cost optimisation goals is proposed in [20]. Finally, in [21] real-time multicast application which is required to meet specified time and geographical constraints is considered. This study improves steadiness and tightness metrics, defined as functions of maximum and minimum individual point-to-point delay.

6 Timeliness Issue in Routing and Handover

In cellular wireless networks, a mobile node that roams from one cell to a neighbouring cell could be a cellular handset or an IP node assigned with an IP address. No matter what type of mobile node it is, when the handover occurs, the information sent to or from the mobile node should be rerouted in time. However, in current research, for different type of mobile nodes, the mechanism of rerouting and route optimisation is different and thus the corresponding timeliness issue is addressed in a different manner.

6.1 Timeliness Issue in Mobile IP

The basic IETF mobile IP protocol provides for transparent packet routing to mobile hosts on the Internet. This protocol suggests that all the packets must pass through the home agent, which will then tunnel them to the current foreign network. A proposed extension of mobile IP is a route optimisation scheme of sending a binding update message to the correspondent node so that it could perform optimisation in the route to the mobile node. In this approach, the correspondent node is required to maintain a binding cache, which is basically a tuple consisting of the foreign network the mobile node is currently in, the home agent of the mobile node, and the time for which the mobile node will be in the current foreign network. An extension to the route optimisation scheme has also been proposed which suggests an approach where the home network sends information about the mobile node's current location via a piggy-back message. The corresponding node then communicates directly with the mobile node. This approach has the additional advantage that it does not require the binding cache or the binding list to be present.

The current research indicates that there are two different perspectives to a node's mobility. In the first scenario, the mobile node is a station with a fixed IP. The mobile node then moves from the home network to a foreign network. The problem in mobile IP is maintaining connectivity within this mobile node after it moves to the new location (i.e., foreign network) provided it has the same IP as earlier. The mobile IP protocol addresses this problem with the "tunnelling" solution. The Route optimisation based on this is improved by constructing an optimum route between the correspondent node and the mobile node. In the second scenario, the mobile node is relatively more 'mobile' than it is in first scenario. In a cellular environment, the node is continuously moving between cells, i.e, it is rapidly initiating handovers. In such a situation, an idea has been proposed that alternate routes should be established between the correspondent node and the mobile node so that connectivity could be maintained, even under continuous and rapid handovers.

Consider the integration of the above two scenarios, where the mobile node is an Internet host, and is connected to the Internet via a base station and an onward wired link. The mobile node has a fixed IP address and also rapidly initiates handovers. To effectively maintain the connectivity under such a situation, one would look for a combined solution. Therefore, under this problem scenario, the home network would tunnel the datagram to the foreign network, until such time an alternate route is not established (via the route optimisation philosophy). But when the mobile node is in continuous motion, route optimisation should be performed each time a handover occurs.

Since the route optimisation is a time consuming task, it may delay the messages rerouting to the right destination. In addition, if such a route optimisation is performed frequently the network system will get burdened performing this task instead of the normal communication.

To solve this problem, we propose that it is not necessary to perform the route optimisation after each handover, but the optimisation should be based on the motion pattern of the mobile node [22]. Note that the handover here has two connotations: a change in the wireless link and a change in the point of contact. If the binding indicates that the mobile node has an extended period of stay in the foreign network, then the route optimisation needs to be carried out so that the following new messages could be rerouted to the mobile node quickly. However, if the mobile node is to be in the present foreign network for only a short period of time (either because it is continuously in motion or for some other reason), then route optimisation is not necessary because potential handover may happen in a short period of time. So the messages are still forwarded from the home network to the foreign network instead of finding a new route from the corresponding node to the foreign network and then routing all the following messages.

A simulation model has been conducted in [22] that aims to simulate the above circumstance, where mobile node movement has been modelled in four different patterns - linear single dimension movement, completely haphazard movement, straight movement and random movement. It compared the performance of the above two approaches, one of which carried out a route optimisation over every handover, while the other advocates forwarding, where the packet meant for a mobile node in a foreign network via the home network. The simulation results indicate that the overall cost of our approach is less than that of the former approach.

6.2 Timeliness Issue in Cellular ATM System

Like the handover process in mobile IP, efficient handover schemes have been proposed that aim to reduce latency for a cellular ATM network. The virtual connection tree (VCT) concept where virtual circuits are pre-established from the root of the tree to each base station was proposed in [10]. Therefore, a handover involves only the switching between virtual circuits. In [10] it was claimed that admission control is invoked only in new virtual connection establishment and the handover, which is cross to the adjacent virtual connection. Since the geographical area spanned by a virtual connection tree may cover large area and contain many base stations, the frequency of the admission control involvement is still low. Hence, the related handover problems caused by small cells are avoided. On the other hand, [11] doubted the low admission control involvement of the connection tree during a mobile handover, and pointed out that the connection tree generates large overhead during a handover due to call admission processing in every node along the new route. To address this drawback, we proposed a mobile virtual path network architecture (MVPA) where the pre-defined virtual path topology eliminates the need for elaborate call routing functions and switching table. Also, call admission control decisions only need to be executed in the mobile ATM switch (MAS) and the area communication server (ACS).

However, both VCT and MVPA ignore the fact that a large portion of the wireless communication is usually in the same area covered by the same VCT or ACS, which is generally defined as a local traffic. With the VCT or MVPA approaches, all local traffic still has to go through the VCT root or ACS even if the source and destination mobiles are covered by different MASs under the same ACS. The route via the root may not be the shortest path, and is likely to delay the transmission of messages, and also creates a large and unnecessary overhead. As a remedy, [23] proposes a cross-tree concept, namely virtual circuit cross-tree, to separate the local and across tree traffic. Similar to the routing in VCT, the traffic across different virtual connection tree would be sent to the root and transmitted to another virtual connection tree. However, for the local traffic within the region covered by the same virtual connection tree, it will be more efficient to route the traffic from the source node to its nearest parent controller which is able to route the traffic to the destination.

When a handover happens, the mobile node may move within or across a VCT region. In order to quickly reroute each handover virtual connection to the new cell, a remote and local mobile rerouting path architecture is needed. In this architecture, admission control is not necessary for every handover, but only for the handover across an adjacent region. When local traffic is the major traffic, the latency and overhead resulting from handover could be improved a lot than the original VTC and MVPA approaches.

7 Timeliness Issue in Multicast and Handover

Current trends in networking application, such as multimedia, indicates that there will be an increasing demand in future network for mobile multimedia communication. Multimedia applications require support from the underlying broadband network at the end-to-end communication level. Multicasting is an important paradigm of end-to-end communication. It is a type of group communication which requires simultaneous transmission of messages from a source to a group of destinations. The route of multicast can be viewed as a tree which is a data distribution path consisting of the router nodes and links to carry the data flow from a source to destinations. The routing system replicates the data at the root node of a tree and forwards the data along various branches leading to destinations at the leaf nodes of the tree.

In the cluster computing arena, multicast is also an often used method to distribute message from one node to multiple nodes within a computing cluster. The same principle applies to the mobile cluster computing applications. If a mobile node is a member of a computing cluster as well as a multicast tree which is used to distribute messages among multiple nodes within the cluster, when it handovers from one cell to another cell, the multicast tree may need to be changed by constructing a new route to handover mobile and deleting the original unused route. Since such multicast tree reconstruction is a time consuming process, timeliness becomes an important issue which the network designer should consider. If the reconstruction takes too long, the message sent to the handover node may be delayed or even be lost, then the message received by different nodes will not be synchronised which will lead to the retransmission or pending of some of the processes.

Many cellular networks follow a centralised network management scheme that uses a base station organisation where all communication between nodes is handled by the base stations. Another option for the cellular networks is to use a distributed network management scheme, which is called an ad-hoc cellular network, where several mobile nodes come together in a small area and establish peer-to-peer communication among themselves without the use of the base station. In this section, we discuss timeliness issue in multicast and handover under two types of mobile wireless networks: base-station-oriented and adhoc network. The type of nodes constituting the multicast tree in these two networks are different. In base-station-oriented cellular network, the multicast tree is composed by the base stations and the mobile nodes, where the base stations are the intermediate nodes of the tree and the mobiles are the leaf nodes of the tree. However in ad-hoc networks, since there is no base-station, all nodes in the multicast tree are mobile nodes. These nodes connect with each other according to a logical topology and one or many of them may be selected as leader(s) to perform multicast just as a base-station does in the base-station-oriented network.

7.1 Timeliness Issue of Multicast and Handover in Base-Station Oriented Network

Timeliness is a key component of the QoS requirements, and this is commonly measured as end-to-end delay. The purpose of our research is to maintain the end-to-end delay of the multicast session under a tolerable value during handover in a cellular network [24]. We first discuss the timeliness issue of multicast when the handover occurs in a cellular network. Then we commence with a triangle mesh topology. We also propose a new multicast tree reconstruction algorithm under the proposed triangle mesh topology.

7.1.1 Timeliness Issue in Handover and Multicast

When a mobile node handovers from one cell to another cell, the packets should be rerouted from the old base station to the new base station. If the mobile node is also within a multicast tree, when handover occurs, it may no longer receive any multicast messages if the new base station is not within the same tree and will not forward further multicast message to it. Under such circumstance, not only normal rerouting but also multicast tree reconstruction should be accomplished before the handover switching is finished so that the normal communication and multicast session would not be broken. Hence, there are two timeliness aspects we should consider during handover and multicast. One is the time to reroute the messages which has been discussed in the last section, the other is the time for a multicast message to reach the destination which is the focus of this research.

7.1.2 Multicast Tree Reconstruction

Multicast tree reconstruction is applied to adjust the multicast tree and to prepare for the potential handover. It aims to guarantee that the multicast tree can maintain the timeliness requirement even after been reconstructed. Due to the time cost of reconstructing the tree, we propose to keep a suitable multicast tree which satisfies the timeliness requirement during the multicast session by just adding a hop into the original multicast tree instead of reconstructing a new one after each handover. Such a new multicast tree may not be the optimum tree, but the reconstruction time has been reduced significantly by avoiding the global communication and tree construction (while satisfying the timeliness requirement of multicast). This "just add a hop" multicast tree reconstruction approach is advantageous even when frequent handovers happen to a mobile node, because if during each handover process, the multicast tree should be reconstructed globally, the system resources would be exhausted by a new tree construction instead of the normal communication. But when such a non-optimal tree cannot satisfy the timeliness requirement of multicast, a new multicast tree is required to be constructed. The process of deciding when to reconstruct the tree is called *multicast tree optimisation condition judgement*. This judgement assumes the direction of the next handover, estimates the end-to-end delay of the new path by adding a hop to the original path, and then compares such delay with the timeliness requirement. If the time delay along the new path is still within the tolerance of the normal multicast requirement, the new multicast tree construction should be initiated.

7.1.3 Triangle Mesh

Before the introduction of the new multicast tree construction and reconstruction algorithm, we commence with a new cellular network topology. In cellular network, hierarchical topology is the most commonly used model where PSTN, central controllers, base stations and mobile users construct a hierarchical architecture. All the traffic across the inter or intra-region cells will pass through the central controllers; this makes the cellular network system constructed as a multiple layer star topology with the central controller acting as the star centre. This topology is indeed a centralised topology where the central controller may become the bottleneck during a heavy loaded traffic hour. To solve this problem, we commence with a new topology where each base station is connected with other base stations in its neighbouring cells. In this way, the intra-region traffic are processed by base stations themselves through the physical links while the central controllers only handle the inter-region traffic. If we idealise every cell as a regular hexagon, the entire connection among the base stations is a triangle mesh. We also retain the hierarchical architecture at a higher level as it used to be.

7.1.4 Multicast Tree Construction and Reconstruction in Triangle Mesh

In the cellular network with the above topology which combines hierarchical architecture and triangle mesh architecture together, there are two types of handover - intra-region handover and inter-region handover. We focus on the intra-region handover in this research, while our algorithm over triangle mesh can be easily extended to the whole network.

As shown in Figure 3, a triangle mesh can be divided into the six areas along three lines across a source node. Since each line in a sparse direction will not be the part of the shortest path from the source node to the destination node, we can erase all the lines in the sparse direction inside every area. This is shown in Figure 4, where the triangle mesh changes to quarter 2D mesh in each

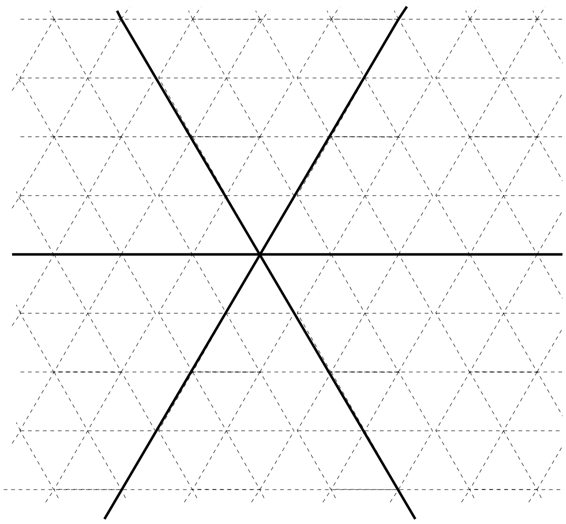

Figure 3: Triangle Mesh.

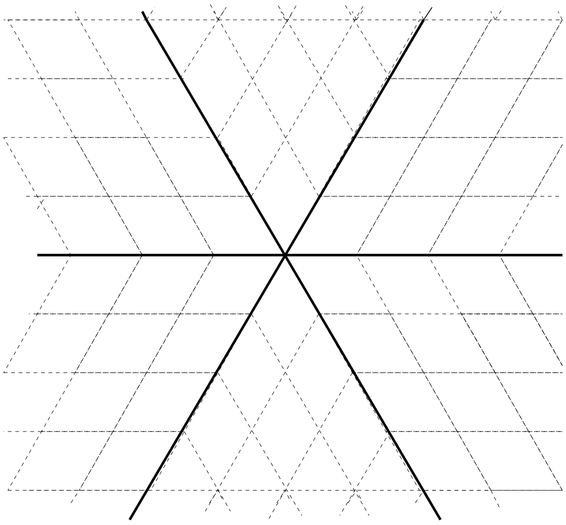

Figure 4: Optimised Triangle Mesh.

of the six areas. To construct the multicast tree in this quarter 2D mesh, we distribute all the nodes into six areas and get the common least ancestor node of every area. Then inside every area, we construct a submulticast tree which has the root at the common least ancestor node by using any of the existing 2D mesh multicast tree construction algorithms. With such a construction, we can get a multicast tree on a triangle mesh, which satisfies the condition that the path from the source node to every destination is the shortest path.

When a mobile node handovers across cells, and "just add a hop" approach is not able to satisfy the timeliness requirement, multicast tree should be reconstructed. First, we get the node that needs to be added to the tree (i.e., the new base station in the cell that the mobile is moving to) after last tree construction or reconstruction. Second, we get the nodes that are not necessary to be included in the tree any more because the mobiles inside these cells have already left for other cells. The reason that they are still in the tree is the "just add a hop" approach includes them as temporary. When the whole tree is being reconstructed, it is not necessary to include these nodes in the tree. If these nodes are leaf nodes, we remove them from the tree and also disconnect the links connecting them from other nodes. Then we check their parent nodes applying the same method and repeat the process until there is no node that is not necessary to be included into the new multicast tree. Finally, for every node that needs to be included into the new tree, we calculate the shortest path from all the nodes inside the tree to it, and use the shortest one that can satisfy the end-to-end delay requirement for each node. Then we add the node into the tree and set up the corresponding links. The multicast tree is then reconstructed.

7.2 Timeliness Issue of Multicast and Handover in Ad-hoc Network

In ad-hoc cellular networks, the timeliness issue becomes more important and needs serious consideration because the multicast tree is composed of mobile nodes which may handover from one cell to another cell at any time. When handover happens, the nodes connected to the migrating node become orphans and the multicast tree becomes disconnected. If the tree cannot be rebuilt soon, multicast message would not reach all the destinations within the time period it requests. We consider the multicast tree reconfiguration problem on an ad-hoc cellular network. Our purpose is not to generate the 'best' multicast tree, but to reconstruct a disconnected one [25].

For ease of multicast operation, rapid maintainability and other well known advantages of symmetric topologies, we embed the multicast tree on the top of a mesh of hypercubes topology for the intra- and inter-cell networks. In this logical topology, nodes within each cell are connected as a binary hypercube, while the nodes across the cells are connected as a hexagonal mesh. Heuristics multicast algorithms for hypercubes are proposed in [26], and anyone of them can be used to construct the initial multicast tree. We focus our discussion on the timeliness issue of the multicast tree reconstruction when the handover happens to a mobile node.

In ad-hoc cellular network, an orphan node is created when its parent node migrates. A node can also become an orphan, if it migrates and it is a destination node in the multicast tree. We identify three possible situations for creation of the orphan nodes.

- First, the migratory node is a lead node. Since this node migrates to a new cell, it becomes an orphan and it is necessary to find a new parent node belonging to the multicast tree.

- Second, the migratory node is an intermediate node, but not a multicast destination. In this case, the migratory node itself does not need to re-connect to the multicast tree. But, its children nodes become orphans and are required to find new parents.

- Third, the migratory node is an intermediate node, and is also a multicast destination. In this case, the migratory node needs to find a new parent after moving into the new cell, and at the same time its children become orphans and each one of them is required to find a new parent node.

Regardless of whether the node became an orphan due to its own migration, or its former parent node's migration, the net effect is identical, i.e., the fact that it is required to find a new parent. To accomplish that, a rapid replacement of the migrated node into the hypercube topology is first needed, and then a rapid reconfiguration of the multicast tree is required.

To keep the maximum distance from the root to any one of the leaf nodes minimised, it is preferred to create a balanced tree, instead of a skewed one. In doing so, if the multicast tree is uniformly spread and balanced to begin with, then during each tree reconstruction phase, if the orphan node(s) can connect to a new parent belonging to a tree at the same level as that of the previous parent, then the balanced property of the tree remains unchanged. Otherwise, the tree may become progressively imbalanced. As the tree gets more and more imbalanced, some of the leaf nodes will get unduly away from the root, causing additional message transmission delay. Hence our design objective is to have an orphan node get a parent whose tree level is most similar to that of the node's former parent. We propose two approaches below for rapid multicast tree reconstruction based on the proposed hypercube topology.

7.2.1 RRR Approach

The first solution approach, named "Request-Reply-Rejection" (RRR) is designed to operate at run-time, i.e, to be invoked at a time the multicast tree gets fragmented (due to node migration) and is required to be connected or reconstructed. With this approach, when a node migrates, an orphan node at the l-th level will attempt to find a parent so that orphan node request other nodes to be their parent. The node receiving the request either accepts or rejects the request depending on whether or not the timeliness constraints of message transmission can be satisfied. Also the node requested and accepting to be the new parent must be a hypercube adjacent to the requesting one (i.e., the orphan). This RRR algorithm can also be used to improve the multicast tree. During the system idle time, any node connected to a parent may use the same approach to find a more optimal parent, and consequently update the multicast tree. To maintain a "good shape" multicast tree, and avoid creating a skewed shaped one, ideally, each reconstruction step should maintain a height-balanced tree, where the difference between the maximum and minimum height of the leaf is at most one. However, including this check in each step of the RRR algorithm can make the algorithm computationally expensive. Besides, there is no guarantee that among the available set of topologically adjacent parents of the orphan node at least one would offer height-balance maintaining connection. Our proposed approach is to first list all the available parent nodes and their topological adjacency. Next, these available parent nodes are sorted in the tree level discrepancy from the previous parent of the orphan node. Clearly, the first element of the sorted list would offer a parent node that is closest in tree height level to the former parent of the orphan node.

7.2.2 LAP Approach

Another approach, named "Look-ahead Alternate Parent" (LAP) is based on precomputed alternate multicast tree reconstruction techniques. It is required to execute the RRR algorithm or an equivalent one in the background. In the LAP algorithm, each node has a local parent table containing the current parent and a number of alternate parents computed in the background. Due to one or more node migrations, when orphan node(s) are created, each orphan node readily selects one alternate parent from its alternate-parent table. Ideally, this approach would be able to reconstruct the multicast tree in zero waiting time; however, in practice a small table lookup delay would be involved. Also, if the alternate parent table is not ready and available, then associated delays for executing RRR algorithm in the background would also account as a foreground delay.

8 Conclusion

We have discussed the architecture of mobile cluster computer, timeliness issue of rerouting, and multicast when handover occurs in the mobile cluster computing area. Several solution approaches based on different system architectures have been discussed. To successfully reroute the message from the home network to the corresponding node when a mobile node handover occurs a mobile IP tunnels the message from the home network to the foreign network until route optimisation is accomplished. To accommodate the situation when mobile node handovers occur frequently and rapidly, we propose the idea to delay the route optimisation to a certain period and tunnel the message from the home network to a new foreign network so that the the system would not get burdened by frequent route optimisation. In an ATM-based cellular network, the messages are rerouted from the base station of the mobile node to the central controller, which makes the central controller to become the bottleneck and thus the timeliness requirement may not be meet. We propose that base stations within the same region communicate with each other and process the intra-region rerouting by themselves. For the timeliness issue in multicast during handover, we propose the "just add a hop" idea based on a base-station oriented triangle mesh topology so that the multicast tree construction and reconstruction become more efficient. In addition, in an ad-hoc cellular network, we commence with a hypercube topology where two approaches–RRR and LAP–are proposed to accelerate the multicast tree reconstruction.

With rapid developments/progress in the area of cluster computing, reliable wireless communication, mobile computing, and availability of applications that exploit this integrated infrastructure, mobile cluster computing is poised to become a reality in the coming 21st century.

Acknowledgments

We thank Toni Cortes (Universitat Politècnica de Catalunya, Spain), Dan Hyde (Bucknell University, USA), Lars Rzymianowicz (University of Mannheim, Germany), Norm Matloff (University of California, Davis, USA), Alessandro Bevilacquai (INFN Istituto Nazionale di Fisica Nucleare, Italy), Alfred C. Weaver (University of Virginia, USA), and Harjinder Sandhu (York University, Canada) for their suggestions and comments.

References

[1] M. Baker and R. Buyya, "Cluster Computing at a Glance", *High Performance Cluster Computing: Architectures and Systems*, Vol. 1, 1/e, Prentice Hall, NJ, USA, 1999.

[2] Intel, “Mobile Computing”, http://www.intel.com/mobile/

[3] Intel, “Mobile Pentium II Processors”, http://www.intel.com/mobile/pentiumii/

[4] Intel, “Mobile Celeron Processor”, http://www.intel.com/mobile/celeron/

[5] AMD. “Mobile Processors”, http://www.amd.com/products/cpg/mobile.html

[6] A. Noerpel and Y. Lin, “Handover Management for a PCS Network”, *IEEE Communications Magazine*, Vol. 3, No. 6, pp. 18-24, Dec. 1997.

[7] P. McKinley, H Xu, A-H. Esfahanian and L. Ni, “Unicast-based Multicast Communication in Wormhole-routed Networks”, *IEEE Trans. on Parallel and Distributed Systems*, Vol. 5, pp. 1252-1265, Dec. 1994.

[8] S. Bhattacharya, C. Albert, and W. Tsai, “Fault-tolerant Multicast on Hypercubes”, *Journal of Parallel and Distributed Computing*, Vol. 23, pp. 418-428, 1994.

[9] H. Zheng and S. Bhattacharya, “Working Set in Channel Management in Cellular Networks”, *Proceedings of ISADS'99 Conference*, Mar. 1999.

[10] Acampora and M. Naghishineh, “An Architecture and Methodology for Mobile Executed Handoff in Cellular ATM Networks”, *IEEE JSAC*, Oct. 1994, pp. 1365-1375.

[11] H. Vogel, “A Networking Concept for Wide Area Mobility and Fast Handoff in ATM Networks ”, *IEEE Globecom*, v2. 1997, pp.1124-1128.

[12] B. Lancki, A. Dixit, and V. Gupta, “Mobile IP: Supporting Transparent Host Migration on the Internet”, *Linux Journal*, Aug. 1996.

[13] C. Perkins and D. Johnson, “Route Optimisation in Mobile IP”, *Internet draft*, Nov. 1997.

[14] D. Johnson and C. Perkins, “Mobility Support in IPv6”, *IETF mobile IP Working Group*, Mar. 1998.

[15] E. Dijkstra, “A Note on Two Problems in Connection with Graphs”, *Numerische Mathematik*, Vol. 1, pp. 269-271, 1959.

[16] B. Waxman, “Delay-bounded Steiner Tree Algorithm for Performance-Driven Layout”, *Journal on Selected Areas in Communications*, Vol. 6, 1617-1622, Dec. 1988.

[17] H. Takahashi and A. Matsuyama, “An Approximate Solution for the Steiner Problem in Graphs”, *Mathematica Japonica*, Vol. 6, pp. 573-577, 1990.

[18] L. Kou, G. Markowsky, and L. Berman, “A Fast Algorithm for Steiner Trees”, *Acta Information*, Vol. 15, pp. 141-145, 1981.

[19] K. Bharatkumar and J. Jaffe, “Routing to Multiple Destinations in Computer Networks”, *IEEE Trans. on Communications*, Vol. COM-31, pp. 343-351, 1983.

[20] Z. Qing, M. Parsa, and J. Garcia-Luna-Aceves, “A Source-based Algorithm for Delay Constrained Minimum-cost Multicast”, *Infocom'95*, 1995.

[21] M. Pokam and G. Michel, “Guaranteeing Spatial Coherence in Real-time Multicasting”, *International Symposium on Information Theory*, pp. 41, May 1995.

[22] S. Palangala and S. Bhattacharya, “Finding When Route Optimisation Is Necessary”, *Technical Report*, CSE Dept. of Arizona State University, Dec. 1998.

[23] C. Yu and S. Bhattacharya, “Message Delivery Delay Refinement of Large Scale Dynamic Networks”, *Technical Report*, CSE Dept. of Arizona State University, Dec. 1998.

[24] Y. Chen, C. Yu, and S. Bhattacharya, “Real-time Multicast Reconstruction of Handover Process on Cellular Network”, *Technical Report*, CSE Dept. of Arizona State University, Oct. 1998.

[25] B. Gannod and S. Bhattacharya, “Real-time Multicast in Wireless Communication”, *Real-time Special Issue of Parallel and Distributed Computing Practices Journal*, Vol.2 (1), 1999.

[26] Lan Y., Esfahanian A-H., and Ni L.M., “Multicast in Hypercube Multiprocessors”, *Journal of Parallel and Distributed Computing*, Vol. 8, 1990.