RGB: A Scalable and Reliable Group Membership Protocol in Mobile Internet*

Guojun Wang[1,2], Jiannong Cao[1], Keith C. C. Chan[1]

[1]Department of Computing
Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
{csgjwang,csjcao,cskcchan}@comp.polyu.edu.hk

[2]School of Information Science and Engineering
Central South University
Changsha, Hunan, P. R. China, 410083
gordon434@163.net

Abstract

We propose a membership protocol for group communications in mobile Internet. The protocol is called RGB, which is the acronym of "a Ring-based hierarchy of access proxies, access Gateways, and Border routers". RGB runs in a parallel and distributed way in the sense that each network entity in the ring-based hierarchy maintains local information about its possible leader, previous, next, parent and child neighbors, and that each network entity independently collects/generates membership change information, which is propagated by the one-round membership algorithm concurrently running in all the logical rings. We prove that the proposed protocol is scalable in the sense that the scalability of a ring-based hierarchy is as good as that of a tree-based hierarchy. We also prove that the proposed protocol is reliable, in the sense that, with high probability of 99.500%, a ring-based hierarchy with up to 1000 access proxies attached by a large number of mobile hosts will not partition when node faulty probability is bounded by 0.1%; if at most 3 partitions are allowed, then the Function-Well probability of the hierarchy is 99.999% accordingly.

1. Introduction

Internet computing and wireless communications are two of the current most important network technologies. In recent years, these two network technologies are converging for mobile Internet computing. With the integration of heterogeneous wired Internet and different kinds of wireless access networks, such as wireless LANs, cellular networks, and satellite networks, more services such as multimedia services with QoS guarantee and personalized services with mobility support, will be deployed in mobile Internet in the near future.

Group communications systems provide communications services among groups of processes. A *group* consists of a set of processes called *members* of the group. A process may voluntarily *join* or *leave* a group, or cease to be a member due to *failure*. The *membership* of a group is a list of currently operational processes in the group. The task of *membership management* is to maintain membership in case of *Member-Join/Leave/Handoff/Failure* events. Typical applications using membership include replicated file systems, distributed database systems, peer-to-peer systems, video conferencing systems and distance learning systems.

Many existing group communications systems are mainly designed for generic LAN or WAN environment, which don't explicitly consider *Mobile Hosts* (MHs) as group members. Therefore, there is no guarantee that they can also work well in the presence of MHs. Our work deals with MHs as group members in mobile Internet. However, the design of a group membership protocol for mobile Internet is a challenging task. In fact, the intrinsic issues in WANs like *high message latency*, *frequent connectivity changes*, and *instability due to link failures or congestion* [15], still exist in mobile Internet. Furthermore, there are more difficult issues which need to be addressed due to in-

---

*This research is supported in part by the Hong Kong Polytechnic University Central Research Grant *G-YY41*, and in part by the University Grant Council of Hong Kong under the CERG Grant PolyU *5170/03E*.

troducing MHs as follows.

Frequent disconnection. MHs are often disconnected from their attached wireless networks. Disconnection can be categorized into three types: *temporary disconnection*, which may resume normal operation within a very short period of time; *voluntary disconnection*, which is initiated by the user, and after an arbitrary period of time may reconnect at any other cell and resume normal operation; *faulty disconnection*, which may be caused by any failure occurrences and may not be allowed to resume normal operation.

Frequent handoff. In order to accommodate requirements of mobile users, such as reducing power consumption and increasing moving speed, the trend is to build smaller wireless cells. With smaller cells, handoffs may occur more frequently. Therefore, fast handoff is needed to decrease service disruptions to mobile users.

Frequent failure occurrence. As is well known that Internet is unreliable in the sense that hosts, routers and communications links may become faulty, and that no communications latency bound can be guaranteed. With different kinds of wireless networks integrated into the wired Internet, it becomes more unreliable: MHs may become faulty more easily than stationary hosts; wireless communications links between MHs and their attached devices may be more unreliable than wired communications links. Therefore, frequent failures may occur in mobile Internet.

The rest of the paper is organized as follows. In Section 2, we introduce some related works. In Section 3, we propose a 4-tier integrated network architecture for mobile Internet computing. In Section 4, we propose a scalable and reliable group membership protocol called *RGB*, the acronym of "a *R*ing-based hierarchy of access proxies, access *G*ateways, and *B*order routers". In Section 5, we analyze the scalability and reliability of our ring-based hierarchy. We conclude this paper in Section 6.

2. Related Works

The work on membership problem in asynchronous systems has been pioneered by the *ISIS* system [6], and the work in synchronous distributed systems has been introduced in [8]. The asynchronous membership problem is later proved to be impossible to solve without additional assumptions on detection of crashed processes [7], which comes from a very famous Fischer-Lynch-Paterson *impossibility result* in asynchronous systems [12].

There are many researches on membership management problem targeted toward generic LAN or WAN environment. In [1, 2], some ring-based schemes are proposed for LANs or multiple LANs within a local area. There are some schemes for WANs with respect to scalability issues. In the *membership roles* scheme [5], *core members*, *client members* and *sink members* are distinguished. In the *Spread* system [3], two levels of protocols are integrated: one for LANs called *Ring*, another for WANs connecting the LANs, called *Hop*.

There are some hierarchical schemes with multiple levels for more scalable solutions as follows. In the *Transis* system [11], a WAN is viewed as a hierarchy of *multicast clusters*, each of which represents a domain of machines capable of communicating via broadcast or multicast hardware. The clusters are arranged in a hierarchy, with representatives from each local domain participating in the next level up the hierarchy. Such a hierarchy is called a *cluster-based hierarchy with representatives*. In the *CONGRESS* system [4], a WAN is viewed as a hierarchy of *domains*, where each domain is serviced by a CONGRESS server: *Local Membership Server* (LMS) and *Global Membership Server* (GMS). LMS is placed in each host, and serves for each client running on its host. GMSs are placed in a tree-based hierarchy, and the higher-level logical GMSs are indeed the lowest-level physical ones. Such a hierarchy is called a *tree-based hierarchy with representatives*. In [14, 15], the *Moshe* membership servers schemes are proposed, which are generalization of the schemes in [4], but without explicitly stating how to organize the servers. In [16], an explicit *layered* scheme is proposed, which is similar to that in the *Transis* system [11].

There are some *one-round* algorithms. The algorithm in [9] terminates within one round in case of a single process crash or join. But in case of multiple faults, it may take a linear number of rounds. In each round, a token revolves around a virtual ring consisting of all the member processes. In [13], all the group members form one logical ring and a token is used to reach agreement. The algorithm in [15] terminates within one round of message communications over 98% of the running time. However, the algorithms are inefficient in case of large group.

There are also some works which explicitly consider MHs as group members. In [17], the authors extend their layered scheme in [16] by reducing the overhead of membership management and the number of messages exchanged to complete the membership change process in mobile Internet. Our work also deals with MHs as group members in mobile Internet. In mobile Internet communications environment, scalability and reliability are two major issues. In particular, different kinds of faults occur more frequently in mobile Internet than in their wired counterpart. As is well known that the tree-based structure has good scalability property and it is used in many communications protocols, for example in IP multicast [10] in wired Internet. The proposed protocol based on the ring-based hierarchy has comparable scalability property with that of the tree-based structure. More interestingly, we show that the proposed protocol based on the ring-based hierarchy is more reliable than that based on the tree-based hierarchy.

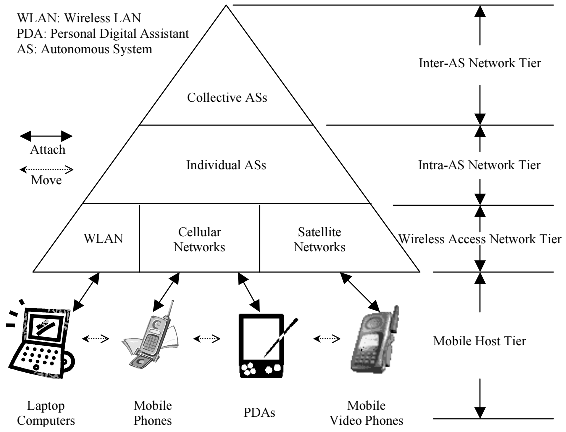

Figure 1. The 4-Tier Integrated Network Architecture for Mobile Internet Computing

3. The 4-Tier Mobile Internet Network Architecture

Different integration strategies for mobile Internet have been investigated by many researchers, such as *Unified Wireless Networks Architecture* [18], *System Architecture for Mobile Communications Systems* [20], *All-IP Wireless/Mobile Network Architecture* [22], and *FIT-MIP Global System Architecture* [19]. Based on these architectures, we propose a basic mobile Internet architecture called the *4-Tier Integrated Network Architecture for Mobile Internet Computing* in Figure 1, which are illustrated as follows.

Mobile Host Tier consists of different kinds of MHs which attach themselves to devices in upward wireless access networks tier. There are some traditional MHs like laptop computers, Personal Digital Assistants (PDAs) and mobile phones, which may be equipped with some new features like multi-mode operations. In addition, some forthcoming MHs like mobile video phones may also appear in the near future.

Wireless Access Network Tier comprises many different kinds of wireless access networks, such as local-area broadband wireless LANs, wide-area cellular networks, and world-wide satellite networks. MHs may attach to different kinds of wireless networks through devices like access points in wireless LANs, base stations in cellular networks, and satellites in satellite networks. We abstract all these devices as *Access Proxies* (APs).

Intra-AS Network Tier corresponds to individual *Autonomous Systems* (ASs) in Internet. As is well-known that Internet is organized as an interconnection of thousands of separate administrative domains called ASs. According to geographical and/or administrative factors, different wireless access networks may attach to different ASs through devices called *Access Gateways* (AGs).

Inter-AS Network Tier is the topmost tier in the hierarchy. As is well-known that *Border Gateway Protocol* (BGP) is the *de facto* standard for controlling routing of traffic across a collection of ASs among *Border Routers* (BRs) in these ASs. Usually AGs may communicate with BRs within the same ASs.

With the 4-tier architecture, we infer that in a mobile group communications system, group members may be highly dynamic, strongly heterogeneous, and size of the group may be potentially very large. In the following section, we propose a novel group membership protocol to manage such group with highly dynamic, strongly heterogeneous, and potentially very large characteristics in mobile Internet.

4. The Ring-based Hierarchical Group Membership Protocol

Based on the above 4-tier integrated network architecture for mobile Internet computing, we propose a *ring-based hierarchy* and associated protocol for membership management. The hierarchy is shown in Figure 2. In the proposed protocol, each mobile host can join or leave a group at will, or fail in the group due to the fact that some errors occur. The membership change information will firstly be captured in its attached access proxy, then propagated along the logical ring where the access proxy resides. The leader node in the logical ring will then propagate such information to its parent node. This process will continue until the leader node in the topmost logical ring is reached. Before presenting the *One-Round Token Passing Membership* algorithm and the *Membership-Query* algorithm, we firstly describe data structures maintained by *Mobile Hosts* (MHs), *Network Entities* (NEs) including *Access Proxies* (APs), *Access Gateways* (AGs) and *Border Routers* (BRs), and *Tokens* which circulate around the logical rings.

4.1. The Ring-based Hierarchy for Group Membership Management

Figure 2 shows different tiers of group membership hierarchy, namely, *Border Router Tier* (BRT), *Access Gateway Tier* (AGT), *Access Proxy Tier* (APT), and *Mobile Host Tier* (MHT), with the higher three tiers consisting of logically organized rings to form a ring-based hierarchy. Notice that only a portion of NEs configured to run the proposed protocol will be involved in the hierarchy.

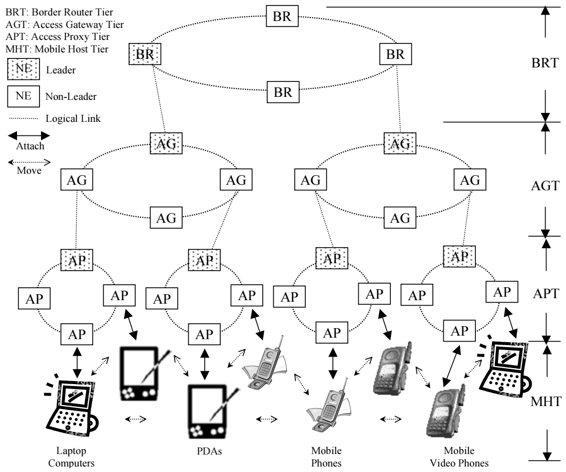

Figure 2. The Ring-based Hierarchy for Group Membership Management

4.2. The Data Structures of MHs, NEs and Tokens

Data structure of MHs. An MH as a group member records the following information.

- GID: GroupID. Group identity, available from some group addressing scheme, e.g. Class D address in IP multicast [10].
- AP: NodeID. Node identity of attached AP, e.g. its IP address.
- GUID: GloballyUniqueID. Globally unique identity of MH, available from some globally unique identity scheme, e.g. Mobile IP *Home Address* [21].
- LUID: LocallyUniqueID. Locally unique identity of MH, available from some locally unique identity scheme, e.g. Mobile IP *Care-of Address* [21].
- Status: Integer. Typical status like *operational*, *disconnected*, and *failed*.

Data structure of NEs. Different kinds of NEs may maintain slightly different information. Details of the data structure are as follows.

- GID: GroupID. See data structure of MHs.
- Current, Leader, Previous, Next, Parent, Child: NodeID. Node identity of the current, leader, previous, next, parent, and child node in the logical ring/hierarchy, e.g. its IP address.
- RingOK: Boolean. If the logical ring containing the current node is functioning well, i.e., a token can circulate normally along the logical ring, then TRUE, else FALSE.
- ParentOK: Boolean. If the parent node exists, and the logical ring containing the parent node functions well, then TRUE, else FALSE.
- ChildOK: Boolean. If the child node exists, and the logical ring containing the child node functions well, then TRUE, else FALSE.
- ListOfLocalMembers[ ]: MemberInfo. List of operational local members.
- ListOfRingMembers[ ]: MemberInfo. List of operational members within the union of coverage areas of all the nodes in the current logical ring for fast membership query.
- ListOfNeighborMembers[ ]: MemberInfo. List of operational members in neighboring nodes in the hierarchy for fast handoff.
- MQ: MessageQueue. Message queue which is self-optimized for aggregating some successive messages into one for further processing.

Data structure of Tokens. Each NE independently collects/generates membership change messages, which are propagated by using a *Token* as follows.

- GID: GroupID. See data structure of MHs.
- Holder: NodeID. Node identity of the holder of the Token, e.g. its IP address.
- OP: TypeOfAggregatedOperations. Type of aggregated Token operations, e.g. Member-Join/Leave/Handoff/Failure, NE-Join/Leave/Failure, Notification-to-Parent/Child, and Holder-Acknowledgement.

4.3. The One-Round Token Passing Membership Algorithm

For membership management, any membership change message such as Member-Join/Leave/Handoff/Failure, will be propagated in the hierarchy from bottom to top by using a *Token*. In each logical ring, after the Token successfully circulates the ring for one round, the control of the Token will be transferred to the next NE in the ring. Figure 3 shows the one-round algorithm on each logical ring.

One of the major functionalities of our one-round algorithm is to reach agreement within each logical ring. Another one is to maintain the status of the ring-based hierarchy. The *Function-Well* status within the current ring is set

```
Algorithm. One-Round Token Passing Membership
Input: The current node CurNode where a Token resides
        and the logical ring where CurNode resides.
Output: Propagate membership change information
        along the ring.

while TRUE do {
  Execute Token.OP on CurNode;
  Set CurNode.RingOK to TRUE;
  if CurNode.Current == CurNode.Leader
    and CurNode.ParentOK then
    CurNode.Parent.MQ.Insert(CurNode,
      Notification-to-Parent);
  if CurNode.ChildOK then
    CurNode.Child.MQ.Insert(CurNode,
      Notification-to-Child);
  if CurNode.Current == Token.Holder then
    Send a corresponding Holder-Acknowledgement
      message to its child(ren) which have sent
      the original messages to the Holder's MQ;
  if CurNode.Current == Token.Holder.Next then {
    Prepare a fresh Token at an appropriate node;
    Transfer control of the fresh Token to the node.}
}
```

Figure 3. The One-Round Token Passing Membership Algorithm

by *RingOK* of each node in the ring, and the *Function-Well* status of the parent node and the child node is set by *ParentOK* and *ChildOK* respectively. A logical ring *functions well* when the Token in the proposed algorithm can circulate around the logical ring normally. Furthermore, if all the logical rings in the ring-based hierarchy function well, then the hierarchy is a *Function-Well* hierarchy. Since at any time there is at most one membership change message propagated along a ring through our one-round algorithm, membership information maintained in the Function-Well hierarchy is consistent.

We illustrate how a membership change message is propagated from bottom to top along the ring-based hierarchy with the *Member-Join* message as an example. If an MH hopes to join a group, it firstly contacts with the AP it knows, which is either manually configured or dynamically acquired. If the MH successfully contacts with the AP, then it sends a Member-Join message to the AP, which is queued in the AP's MQ. If the AP happens to be a single one which is not in any AP logical ring, then it tries to join such a ring. If any *Access Proxy Ring* (APR) satisfies some locality/proximity criterion, then the AP joins the APR. In case that the contact procedure fails, an APR is built to include the single AP itself and make itself the ring leader. Notice that such an APR may merge with its neighboring AP ring or contact with upward tier AGs. Such a join process continues until the topmost logical ring in the hierarchy is reached.

The proposed protocol runs in a parallel and distributed way in the sense that each NE in the hierarchy maintains local information about its possible leader, previous, next, parent and child neighbors, and that each NE independently collects/generates membership change messages to be propagated by the one-round membership algorithm concurrently running in all the logical rings.

4.4. The Membership-Query Algorithm

With our ring-based hierarchy, there are many possible membership maintenance schemes. If only the nodes in the bottommost tier maintain local membership, we call it *Bottommost Membership Scheme* (BMS). If only the nodes in the topmost tier maintain global membership, we call it *Topmost Membership Scheme* (TMS). Since there may exist sub-tiers in each tier, some *Intermediate Membership Schemes* (IMSs) may be possible between BMS and TMS schemes.

For the Membership-Query algorithm with the TMS scheme, firstly the requesting application tries to find some NE with GID, then the NE sends global membership information to the application. For the BMS scheme, firstly the requesting application tries to find some NE with GID, then the NE forwards the request to each bottommost AP leaders, then the AP leaders send local membership back to the original NE or directly to the requesting application to generate global membership information.

The Membership-Query algorithm with the TMS scheme is more efficient than that with the BMS scheme with regard to the requesting application. However, to maintain membership information using the TMS scheme, it is both space- and time-consuming if the membership hierarchy becomes larger. Due to space restriction, we neglect algorithmic descriptions and analysis in this paper.

5. Scalability and Reliability Analysis of the Ring-based Hierarchy

Our proposed protocol uses a ring-based hierarchy, which has the properties of simplicity, scalability and reliability in designing membership and multicast protocols. In subsection 5.1, we show that the ring-based hierarchy has comparable scalability with the tree-based hierarchy, which has been widely used. In subsection 5.2, we argue that the ring-based hierarchy is more reliable than the tree-based hierarchy.

5.1. Comparative Analysis on Scalability Property

In [4], the authors propose a tree-based hierarchy of membership servers (LMSs and GMSs) with representatives called the *CONGRESS hierarchy*. Their one-round algorithm is documented in [14] and refined in [15]. In this section, we compare our ring-based hierarchy with the tree-based hierarchy.

The number *n* of LMSs/APs in the tree/ring-based hierarchy is considered equivalent scalability parameter in the sense that it is more reasonable to consider the number of LMSs/APs, not that of clients/MHs, as the group size.

The one-round algorithm in [14] and [15] for fast membership agreement in a fault-free case is used to measure scalability criteria for both the tree-based hierarchy and the ring-based hierarchy.

We calculate the total number of message hops *HopCount* to propagate the membership change message. Since our major concern is the *proposal* message in [14] and [15], HopCount is approximate to n times the number of the proposal message hops, or n times the number of edges in the hierarchy.

Consider the tree-based hierarchy without representatives. In a tree-based hierarchy with height $h \geq 3$ and with branches $r \geq 2$ for each non-leaf node, the number of leaf nodes (LMSs) is $n = r^{h-1}$. Then HopCount is:

$$HopCount_{Tree-based}(n,h,r)$$

$$\overset{def}{=} n * \sum_{i=0}^{h-2} r^{i+1} \qquad (1)$$

Then consider the tree-based hierarchy with representatives, where some hop counts should be removed from formula (1). For example, $h-2$ should be removed for the root GMS, since there is no real message transfer between the root GMS and its representatives. Then the being removed and the final HopCounts are:

$$HopCountsRemoved_{Tree-based}(n,h,r)$$

$$\overset{def}{=} n * \sum_{i=0}^{h-3}((h-i-2)*(r^i - \sum_{j=0}^{i-1} r^j)) \qquad (2)$$

$$HopCount_{Tree-based}(n,h,r) \overset{def}{=} n*(\sum_{i=0}^{h-2} r^{i+1} -$$

$$\sum_{i=0}^{h-3}((h-i-2)*(r^i - \sum_{j=0}^{i-1} r^j))) \qquad (3)$$

We then normalize HopCount by dividing it with *n*, which stands for the "average" number of messages for one membership change message. Notice that we simply denote it as HC^N_{Tree}.

Table I. Comparison on Scalability between the Tree-based Hierarchy and the Ring-based Hierarchy

n	h	r	HC^N_{Tree}	n	h	r	HC^N_{Ring}
25	3	5	29	25	2	5	35
125	4	5	149	125	3	5	185
625	5	5	750	625	4	5	935
100	3	10	109	100	2	10	120
1000	4	10	1099	1000	3	10	1220
10000	5	10	11000	10000	4	10	12220

$$HC^N_{Tree} \overset{def}{=} N\_HopCount_{Tree-based}(n,h,r)$$

$$\overset{def}{=} \sum_{i=0}^{h-2} r^{i+1} - \sum_{i=0}^{h-3}((h-i-2)*(r^i - \sum_{j=0}^{i-1} r^j)) \quad = r^{h-1} +$$

$$\sum_{i=0}^{h-3} \frac{r^i(r^2-(h-i-1)r+2(h-i-2))-(h-i-2)}{r-1} \qquad (4)$$

We then calculate HopCount in the ring-based hierarchy with height $h \geq 2$ and with each ring containing exactly $r \geq 2$ nodes. The number of APs in the bottommost logical rings is $n = r^h$, and the total number of logical rings is $tn = \sum_{i=0}^{h-1} r^i$. Then HopCount is:

$$HopCount_{Ring-based}(n,h,r) \overset{def}{=} n*((r+1)*tn-1) \qquad (5)$$

We then normalize HopCount by dividing it with *n*, which stands for the "average" number of messages for one membership change message. Notice that we simply denote it as HC^N_{Ring}.

$$HC^N_{Ring} \overset{def}{=} N\_HopCount_{Ring-based}(n,h,r)$$

$$\overset{def}{=} (r+1)*tn-1 = (r+1)*\sum_{i=0}^{h-1} r^i - 1 \qquad (6)$$

We then give numerical results according to formulae (4) and (6) in Table I. As we can see, the scalability property of the ring-based hierarchy is almost the same as that of the tree-based hierarchy. The two schemes are comparable with respect to scalability.

5.2. Comparative Analysis on Reliability Property

In order to compare reliability between the tree-based hierarchy with representatives [4] and our ring-based hierarchy, we define a *transformation* hierarchy called *a tree-based hierarchy without representatives*. In such a hierarchy: (A) nodes are physically different from each other; (B) nodes in the lowest level with the common parent are logically connected into a ring; and (C) nodes not in the lowest level but with the common parent are logically connected into a ring.

If we remove the root node and the associated edges from the transformation hierarchy and remove all the *parent-children* edges but the first one from such a relationship, then such a hierarchy becomes our ring-based hierarchy. Since we consider only node faults, we deduce that the tree-based hierarchy without representatives has similar reliability property as our ring-based hierarchy.

Since one representative node fault is indeed several logical node faults in the tree-based hierarchy with representatives, the tree-based hierarchy without representatives is more reliable than that with representatives. Thus, the ring-based hierarchy is more reliable than the tree-based hierarchy with representatives. Below we only analyze the reliability of the ring-based hierarchy.

We now present an analytical model for *Function-Well* (*fw*) probability of our ring-based hierarchy with the following parameters: *n*, *h*, *r*, which are previously mentioned; *f*, which denotes the node faulty probability with uniform and independent fault distribution in the hierarchy; and *k*, which is the maximal number of allowed partitions.

We assume only node fault in the hierarchy, while link fault will be simulated by node fault. We also assume that any single node fault in a logical ring can be detected quickly by *Token* retransmission schemes and be locally repaired by excluding the faulty node from the ring. If there are more than one fault in the ring, then the ring is considered to be partitioned, which will merge with other partitions later. In case that the hierarchy is partitioned into no more than *k* partitions, the hierarchy is considered *Function-Well*.

We firstly present the Function-Well probability *t* of each logical ring as:

$$t \overset{def}{=} Prob_{fw-ring}(r,f) \overset{def}{=} \sum_{i=0}^{1} \binom{r}{i} (1-f)^{r-i} f^{i}$$

$$= (1-f+rf)*(1-f)^{r-1} \qquad (7)$$

We then suppose a full ring-based hierarchy for *worst-case* analysis: it contains maximal number of tiers; each tier contains maximal number of logical rings; and each logical ring contains maximal number of nodes. Such a hierarchy contains *tn* = $\sum_{i=0}^{h-1} r^i$ logical rings and less than *k* logical rings may not function well. We present the Function-Well probability of the hierarchy as:

$$Prob_{fw-hierarchy}(n,h,r,f,k)$$

$$\overset{def}{=} \sum_{i=0}^{k-1} \binom{tn}{i} t^{tn-i}(1-t)^{i} \qquad (8)$$

Table II. Function-Well Probability of the Ring-based Hierarchy (Left: h=3, r=5; Right: h=3, r=10)

n	$f(\%)$	k	$fw(\%)$	n	$f(\%)$	k	$fw(\%)$
125	0.1	1	99.968	1000	0.1	1	99.500
125	0.1	2	99.999	1000	0.1	2	99.994
125	0.1	3	99.999	1000	0.1	3	99.996
125	0.5	1	99.211	1000	0.5	1	88.448
125	0.5	2	99.972	1000	0.5	2	99.215
125	0.5	3	99.975	1000	0.5	3	99.864
125	2.0	1	88.409	1000	2.0	1	16.094
125	2.0	2	98.981	1000	2.0	2	45.470
125	2.0	3	99.592	1000	2.0	3	72.038

Numerical results from (7) and (8) are given in Table II with conclusions as follows.

(1) Our ring-based hierarchy is reliable in the sense that, with high probability of 99.500%, a ring-based hierarchy with up to 1000 APs directly attached by a large number of MHs will not partition when the node faulty probability is bounded by 0.1%; if at most 3 partitions are allowed, then the *Function-Well* probability of the hierarchy is 99.999% accordingly.

(2) Under the definition of Function-Well hierarchy with at most 3 partitions allowed, with high probability of 99.864%, a group with up to 1000 APs directly attached by a large number of MHs guarantees that the hierarchy still functions well when the node faulty probability is bounded by 0.5%.

(3) With the node faulty probability increasing to 2.0%, the small-scale hierarchy still functions well with very high probability. For example, the *Function-Well* probability is 99.592% for a small-scale hierarchy with up to 125 APs directly attached by MHs. However, the large-scale hierarchy with up to 1000 APs directly attached by MHs functions well with probability of only 72.038%.

6. Conclusions

The scalable and reliable properties of the RGB protocol are very important to provide reliable large-scale group communications services for a large number of applications in unreliable communications networks. As a final remark, we argue that the proposed protocol is efficient similar to tree-based protocols since only a sequence of logical rings from bottom to top, not all the rings in the hierarchy, will be involved with respect to any specific membership change message. In particular, the delay for propagating membership messages with small-scale logical rings is smaller compared with that with large-scale logical rings, while the small-scale logical rings are more common than large-scale ones. As our future work, we will extend RGB with *Membership-Partition/Merge* algorithms to provide *partitionable* and *self-organizable* group membership services to applications.

References

[1] Y. Amir, L.E. Moser, P.M. Melliar-Smith, D.A. Agarwal, and P.Ciarfella, "The Totem single-ring ordering and membership protocol," *ACM Transactions on Computer Systems*, Vol. 13, Issue 4, pp. 311-342, November 1995.

[2] D.A. Agarwal, L.E. Moser, P.M. Melliar-Smith, and R.K. Budhia, "The Totem multiple-ring ordering and topology maintenance protocol," *ACM Transactions on Computer Systems*, Vol. 16, Issue 2, pp. 93-132, May 1998.

[3] Y. Amir and J. Stanton, "The Spread wide area group communication system," *Technical Report TR CNDS-98-4*, The Center for Networking and Distributed Systems, The Johns Hopkins University, Baltimore, MD, 1998.

[4] T. Anker, G.V. Chockler, D. Dolev, and I. Keidar, "Scalable group membership services for novel applications," *Networks in Distributed Computing (DIMACS Workshop)*, Vol. 45, American Mathematical Society, pp. 23-42, 1998.

[5] O. Babaoglu and A. Schiper, "On group communication in large-scale distributed systems," *Proc. the 6th workshop on ACM SIGOPS European workshop: Matching operating systems to application needs*, pp. 17-22, 1994.

[6] K.P. Birman and T.A. Joseph, "Reliable communication in the presence of failures," *ACM Transactions on Computer Systems*, Vol. 5, Issue 1, pp. 47-76, February 1987.

[7] T.D. Chandra, V. Hadzilacos, S. Toueg, and B. Charron-Bost, "On the impossibility of group membership," *Proceedings of the fifteenth annual ACM symposium on Principles of distributed computing*, pp. 322-330, 1996.

[8] F. Cristian, "Reaching agreement on processor-group membership in synchronous distributed systems," *Distributed Computing*, Vol. 4, pp. 175-187, 1991.

[9] F. Cristian and F. Schmuck, "Agreeing on processor group membership in timed asynchronous distributed systems," *UCSD Technical Report CSE95-428*, University of California, San Diego, 1995.

[10] S. E. Deering, "Host extensions for IP multicasting," *IETF RFC 1112*, SRI Network Information Center, August 1989.

[11] D. Dolev and D. Malki, "The Transis approach to high availability cluster communication," *Communications of the ACM*, Vol. 39, Issue 4, pp. 64-70, April 1996.

[12] M.J. Fischer, N.A. Lynch, and M.S. Paterson, "Impossibility of distributed consensus with one faulty process," *Journal of the ACM (JACM)*, Vol. 32, Issue 2, pp. 374-382, April 1985.

[13] M.A. Hiltunen and R.D. Schlichting, "A configurable membership service," *IEEE Transactions on Computers*, Vol. 47, Issue 5, pp. 573-586, May 1998.

[14] I. Keidar, J. Sussman, K. Marzullo, and D. Dolev, "A client-server oriented algorithm for virtually synchronous group membership in WANs," *Proc. the 20th International Conference on Distributed Computing Systems*, pp. 356-365, April 2000.

[15] I. Keidar, J. Sussman, K. Marzullo, and D. Dolev, "Moshe: A group membership service for WANs," *ACM Transactions on Computer Systems*, Vol. 20, Issue 3, pp. 191-238, August 2002.

[16] H. Kim, D. Lee, and H.Y. Youn. "A scalable membership service for group communications in WANs," *Proc. the 2000 Pacific Rim International Symposium on Dependable Computing*, pp. 59-66, 2000.

[17] B. Kim, D. Lee, and D. Nam, "Scalable group membership service for mobile Internet," *Proc. the Seventh International Workshop on Object-Oriented Real-Time Dependable Systems (WORDS 2002)*, pp. 295-298, 2002.

[18] W.W. Lu, "Compact multidimensional broadband wireless: The convergence of wireless mobile and access," *IEEE Communications Magazine*, Vol. 38, Issue 11, pp. 119-123, November 2000.

[19] L. Morand and S. Tessier, "Global mobility approach with Mobile IP in All IP networks," *IEEE International Conference on Communications (ICC 2002)*, Vol. 4, pp. 2075-2079, 2002.

[20] T. Otsu, N. Umeda, and Y. Yamao, "System architecture for mobile communications systems beyond IMT-2000," *Proc. the IEEE Global Telecommunications Conference (GLOBECOM 2001)*, Vol. 1, pp. 538-542, 2001.

[21] C. Perkins, "IP mobility support," *IETF RFC 2002*, October 1996.

[22] T.B. Zahariadis, K.G. Vaxevanakis, C.P. Tsantilas, N.A. Zervos, and N.A. Nikolaou, "Global roaming in next-generation networks," *IEEE Communications Magazine*, Vol. 40, Issue 2, pp. 145-151, February 2002.