一個分散式二元森林 K-互斥方法

A BINARY FOREST QUORUM STRATEGY FOR K-MUTUAL EXCLUSION IN DISTRIBUTED SYSTEMS

張玉盈[†]
Ye-In Chang[†]

陳柏旭[‡]
Bor-Hsu Chen[‡]

[†]資訊工程學系
[†]Dept. of Computer Science and Engineering
國立中山大學
National Sun Yat-Sen University
Kaohsiung, Taiwan
Republic of China
E-mail: changyi@cse.nsysu.edu.tw

[‡]應用數學學系
[‡]Dept. of Applied Mathematics
國立中山大學
National Sun Yat-Sen University
Kaohsiung, Taiwan
Republic of China

摘要

在k-互斥的問題中, 對執行一稱爲臨界區間結構性的程式片段以取得共享資源的情況必須加以控制, 以確保在任何時間內最多只能有k個程序能進入臨界區間。在本篇論文中, 我們提出二元森林方法, 其邏輯架構乃以二元森林架構爲基礎。而其在最佳情況下, 所得的法定節點集爲$2\lceil \lg_2 \frac{n}{2k} \rceil$, 在最差的情況下, 所得的法定節點集爲 $(\lceil \frac{n}{2k} \rceil + 1)$, n是系統中的節點個數。另外, 在最佳的情況下, 利用此法可使其錯誤容忍度提升爲$(n - 2k\lceil \lg_2 \frac{n}{2k} \rceil)$個錯誤節點, 而在最差的情況下則爲$2k(\lceil \lg_2 \frac{n}{2k} \rceil - 1)$。而且從我們的比較結果顯示, 在大部分的情況中, 二元森林方法可以提供比k-majority, cohorts, 以及 DIV方法更好的可用度。

(關鍵詞: k-互斥, 可靠度, 分散式系統, 容錯特性, 法定一致性)

ABSTRACT

In the problem of k-mutual exclusion, concurrent access to shared resource or the critical section (CS) must be synchronized such that at any time at most k processes can access the CS. In this paper, we propose a *binary forest quorum* strategy. This strategy is based on a logical binary forest structure. The quorum size constructed from the strategy is $2\lceil \lg_2 \frac{n}{2k} \rceil$ in the best case and is $(\lceil \frac{n}{2k} \rceil + 1)$ in the worst case, where n is the number of nodes in the system. Moreover, the strategy can be fault-tolerant up to $(n - 2k\lceil \lg_2 \frac{n}{2k} \rceil)$ node failures in the best case and $2k(\lceil \lg_2 \frac{n}{2k} \rceil - 1)$ in the worst case. From our performance analysis, we show that the binary forest quorum strategy can provide a higher availability than k-majority, cohorts, and DIV strategies almost all the time.

(*Key Words:* K-mutual exclusion, availability, distributed systems, fault tolerance, quorum consensus.)

1. INTRODUCTION

A distributed system consists of a collection of geographically dispersed autonomous nodes connected by a communication network. The nodes have no shared memory, no global clock, and communicated with one another by passing messages. Message propagation delay is finite but unpredictable.

The mutual exclusion problem was originally considered in centralized systems for the synchronization of exclusive access to the shared resource. In the problem of k-mutual exclusion, concurrent access to shared resource or the critical section (CS) must be synchronized such that at any time at most k processes can access the CS, where $k \geq 1$. In distributed systems, the k-mutual exclusion problem arises in several interesting applications. For example, it could be used to monitor the number of processes in distributed systems that are allowed to perform a certain action, such as issuing broadcast messages. In such a case, the system may restrict the number of broadcasting processes so as to control the level of congestion.

Over the past decade, many strategies have been proposed to achieve k-mutual exclusion in distributed systems. These strategies can be divided into two classes: *token-based* strategies and *non-token-based* strategies (or *permission-based* strategies) [7, 17]. In token-based strategies [3, 4, 5, 14, 16, 18, 19], there are k tokens in the system. A node is allowed to enter its CS if it processes the token. In non-token-based strategies [2, 6, 10, 11, 12, 13, 20], a node should collect enough permissions (votes) to form a quorum for entering the critical section. K-mutual exclusion is guaranteed if we can assure that at most k quorums can be formed at any instance.

To make distributed k-mutual exclusion strategies fault-tolerant to node and communication failures, many strategies based on the replica control strategies, for example, *coterie*, have been proposed. In [11, 20], they extended the *majority quorum* strategy to *k-majority quorum* strategy; any permission from $\lceil \frac{n+1}{k+1} \rceil$ $(= W)$ nodes would form a quorum for k-mutual exclusion, when n is the number of nodes in the system. (Note that in the *k-majority quorum* strategy, the following conditions must hold: $k \times W \leq n$ and $(k+1) \times W > n$.) In [10], they proposed a *cohort quorum* for k-mutual exclusion based on a *cohort structure*, $Coh(k, l)$, which has l pairwise disjoint cohorts with the first cohort having k members and the others having more than $(2k - 2)$ members. In [2], they partition n nodes into k classes with each class using any traditional approach to enforce 1-mutual exclusion. When the traditional approach is the majority quorum strategy, the constructed quorum will be called *DIV of majority* quorums.

To reduce the overhead of achieving k-mutual exclusion while supporting fault tolerance, in this paper, we propose a strategy called *binary forest quorums* for k-mutual exclusion, which imposes a logical structure on the network. The proposed strategy is based on a logical binary forest structure. The quorum size constructed from this strategy is $2\lceil \lg_2 \frac{n}{2k} \rceil$ in the best case and is $(\lceil \frac{n}{2k} \rceil + 1)$ in the worst case, where n is the number of nodes in the system. Moreover, this strategy can be fault-tolerant up to $(n - 2k\lceil \lg_2 \frac{n}{2k} \rceil)$ node failures in the best case and $2k(\lceil \lg_2 \frac{n}{2k} \rceil - 1)$ in the worst case. From our performance analysis, we show that the binary forest strategy can provide a higher availability than k-majority, cohorts, and DIV strategies almost all the time.

The rest of the paper is organized as follows. Section 2 describes the background in this paper. In Section 3, we give a survey of several non-token-based strategies for k-mutual exclusion. In Sections 4, we present the binary forest quorum. In Section 5, we make a comparison of the binary forest quorum strategy with k-majority, cohorts, and DIV strategies. Finally, Section 6 gives a conclusion.

2. BACKGROUND

A distributed system is a collection of nodes that may communicate with each other by exchanging messages. K-mutual exclusion strategies concern themselves with controlling the nodes such that at most k nodes can simultaneously access their critical sections. Such strategies can be used to coordinate the sharing of a resource that can be allocated to no more than k nodes at a time [2, 10, 11, 20].

Definition 1. *A* **k-coterie** *C is a family of non-empty subsets of an underlying set U, which is a set containing all system nodes $1, 2, ..., n$. Each member Q in C is called a quorum, and the following properties should hold for the quorums [11, 20].*

1. **The non-intersection Property.** *For any $h(< k)$ pairwise disjoint quorums $Q_1, ..., Q_h$ in C, there exists one quorum Q_{h+1} in C such that $Q_1, ..., Q_{h+1}$ are pairwise disjoint.*

2. **The intersection Property.** *There are no m, $m > k$, pairwise disjoint quorums in C (i.e., there are at most k pairwise disjoint quorums in C).*

3. **The minimality Property.** *There are no two quorums Q_i and Q_j in C such that Q_i is a super set of Q_j where $i \neq j$.*

By the non-intersection property, if there exists one unoccupied entry of the critical section, then some node that waits for entering the critical section can proceed. The intersection property assures that no more than k nodes can form quorums simultaneously, so no more then k nodes can access the critical section at the same time. Again, the minimality property for the k-coterie is for the enhancement of efficiency.

Example 1: $\{\{1, 2\}, \{3, 4\}, \{1, 3\}, \{2, 4\}\}$ is a 2-coterie under $U = \{1, 2, 3, 4\}$.

3. A SURVEY

In this section, we give a survey of several strategies for k-mutual exclusion, including k-majority [11, 20], cohorts [10], and DIV [2] quorums.

3.1 K-Majority Quorums

Suppose there are n nodes in the system. Q is said to be k-majority quorum for k-mutual exclusion if Q contain at least $\lceil \frac{n+1}{k+1} \rceil$ $(= W)$ nodes, where $k \times W \leq n$ and $(k+1) \times W > n$ [11, 20]. For example, there are 1, 2, 3, 4 nodes in the system, the set R of 2-majority quorum is as follows: R = { {1, 2}, {1, 3}, {1, 4}, {2, 3}, {2, 4}, {3, 4} }. Totally, R contains 6 quorums.

The availability of a coterie is defined as the probability that a quorum can be successfully formed. For the rest of the paper, we assume that p is the probability that a node is up and there are n nodes in the systems. For the availability of the k-majority strategy, let $AV(k, h)$ be the function evaluating the probability that h pairwise disjoint quorums can be formed simultaneously. Function $AV(k,h)$ has the following condition:

$AV(k,h) = \sum_{i=h\times\lceil\frac{n+1}{k+1}\rceil}^{n} C(n,i) \times p^i \times (1-p)^{n-i}$.

3.2 Cohorts

In this section, we describe a cohort quorum for k-mutual exclusion, which is based on a cohort structure. The definition of a cohort structure is given as follows [10].

Definition 2. A Cohort Structure. *A cohort structure Coh(k,l) = (C_1, C_2,..., C_l) is a list of pairwise disjoint sets; each set C_i is called a cohort. The cohort structure should observe the following two properties:*

1. $|C_1| = k$.
2. $\forall i : 1 < i \leq l : |C_i| > max(2k-2, k)$, *where max(a, b) = a, if $a \geq b$; otherwise, max(a, b) = b.*

To sum up, a cohort structure $Coh(k,l)$ has l pairwise disjoint cohorts with the first cohort having k members and the other cohorts having more than $(2k-2)$ members. For example, ({1, 2}, {3, 4, 5}, {6, 7, 8, 9, 10}) is $Coh(2,3)$ since it has three pairwise disjoint cohorts with the first cohort and the other cohorts having 2 (= k) and more than 2 (= $2k-2$) members, respectively.

Definition 3. A Cohorts Quorum. *A set Q is said to be a quorum under $Coh(k,l)$ if some cohort C_i in $Coh(k,l)$ is Q's primary cohort, and each cohort C_j, $j > i$, is Q's supporting cohort, where*

1. *a cohort C is Q's primary cohort if $|Q \cap C| = |C| - (k-1)$ (i.e., Q contains all except k-1 members of C), and*
2. *a cohort C is Q's supporting cohort if $|Q \cap C| = 1$ (i.e., Q contains exactly one member of C).*

For example, the following sets are quorums under $Coh(2,2)$ = ({1, 2}, {3, 4, 5}): $Q_1 = \{3,4\}, Q_2 = \{3,5\}, Q_3 = \{4,5\}$, $Q_4 = \{1,3\}, Q_5 = \{1,4\}, Q_6 = \{1,5\}$, $Q_7 = \{2,3\}, Q_8 = \{2,4\}$, and $Q_9 = \{2,5\}$. Quorums Q_1, Q_2, and Q_3 take {3, 4, 5} as their primary cohort and no supporting cohort is needed, and quorums $Q_4, ..., Q_9$ take {1, 2} as their primary cohort and {3, 4, 5} as their supporting cohort. It is easy to check that these nine sets constitute a 2-coterie.

For the availability of the cohorts strategy, since up to k pairwise disjoint quorums can be simultaneously formed in a k-coterie, we should discuss up to k cases for the availability of a k-coterie: the probability of a quorum being formed successfully, the probability of two pairwise quorums being formed successfully, ... , and the probability of k pairwise disjoint quorums begin formed successfully. The (k,l)-availability, $1 \leq l \leq k$, is defined to be the probability that l pairwise disjoint quorums of a k-coterie can be formed successfully; it is used as a measure for the fault-tolerant ability of a solution using k-coterie.

Let $AV(h,l)$ be the function evaluating the probability that h pairwise disjoint quorums under $Coh(k,l)$ can be formed simultaneously. Function $AV(h,l)$ has the following three conditions:

1. $AV(0,l) = 1$.
2. $AV(h,1) = PR(S_1,h,S_1)$. (Note that a quorum takes only one member from the first cohort to make it the primary cohort because $S_1 - k + 1 = k - k + 1 = 1$. We also use S_i to denote $|C_i|$ for $1 \leq i \leq l$, where C_i is the ith item of $Coh(k,l)$ = $(C_1, ..., C_l)$ and we use $PR(s,a,b)$ to denote $\sum_{i=a}^{b} C(s,i) \times p^i \times (1-p)^{s-i}$.
3. $AV(h,l) = AV(h-1,l-1) \times PR(S_l, S_l - k + h, S_l) + AV(h,l-1) \times PR(S_l, h, S_l - k + h - 1)$.

3.3 DIV of Majority Quorums

In the DIV strategy [2], the nodes in the network are partitioned into k classes with each class using any traditional approach to enforce 1-mutual exclusion. When the traditional approach is the majority, the constructed quorum is called *DIV of majority quorum.* For example, there are 1, 2, 3, 4, 5, 6 nodes in the system, and we divided nodes into two classes, (1, 2, 3) and (4, 5, 6). The set R of DIV of majority quorum for 2-mutual exclusion is as follows: R = { {1, 2}, {1, 3}, {2, 3}, {4, 5}, {4, 6}, {5, 6} }.

For the availability of DIV of majority quorum, let $AV(k,h)$ be the function evaluating the probability that h pairwise disjoint quorums can be formed simultaneously. Function $AV(k,h)$ has the following two conditions:

1. $AVM = \sum_{i=\lfloor\frac{n}{2k}\rfloor+1}^{\lceil\frac{n}{k}\rceil} C(\lceil\frac{n}{k}\rceil, i) \times p^i \times (1-p)^{\lceil\frac{n}{k}\rceil - i}$.
2. $AV(k,h) = \sum_{i=h}^{k} C(k,i) \times AVM^i \times (1-AVM)^{k-i}$.

4. BINARY FOREST QUORUMS

In this section, we present a *binary forest quorum* for k-mutual exclusion, in which n nodes are divided into the $2k$ groups. Between groups, we apply the k-majority strategy, and inside each group, we apply the binary tree quorum for 1-mutual exclusion [1]. Therefore, the proposed strategy can be considered as a hybrid approach which contains the k-majority and the binary tree quorum for 1-mutual exclusion [11, 20]. Note that the availability of k-majority is good, when less than $\lceil \frac{k}{2} \rceil$ nodes enter CS from our previous simulation study [8]. That is why we divide n nodes into $2k$ groups.

4.1 Definitions

In this section, we first define the logical binary tree and give the definition of the binary tree quorum for 1-mutual exclusion [1]. Next, based on the binary tree quorum for 1-mutual exclusion, we present the binary forest quorum for k-mutual exclusion.

Definition 4. A Binary Tree. *A binary tree is a finite set of one or more nodes such that*

1. *there is a specially designated node called the root in level 0.*
2. *the remaining nodes are partitioned into S_1, S_2, where each of these sets is a binary tree. S_1, S_2 are called the subtrees of the nodes.*

Therefore, there are 2^i nodes in level i. Consequently, for a complete binary tree of level (h+1), there are totally ($2^{h+1} - 1$) nodes. Moreover, each node in the binary tree of level (h+1) is numbered from top to down and left to right as 0, 1, 2, ..., (2^{h+1}-2) as shown in Figure 1. For a node i, node ($\lfloor \frac{i+1}{2} \rfloor - 1$) is its parent.

Definition 5. A Binary Tree Quorum [1]. *The binary tree quorum strategy logically organizes the nodes in a system as a binary tree structure. A binary tree quorum (recursively) for 1-mutual exclusion consists of*

1. *the root and a binary tree quorum of the left subtree, or*
2. *the root and a binary tree quorum of the right subtree, or*
3. *a binary tree quorum of the left subtree and a binary tree quorum of the right subtree.*

Note that, here we let each node in the distributed system be mapped to a node in the logical binary forest, and the number of nods be denoted as n.

Example 2: For a binary tree of level 3 as shown in Figure 2, the set R of binary tree quorums for 1-mutual exclusion is as follows: R = { {0, 1, 3}, {0, 1, 4}, {0, 3, 4}, {0, 2, 5}, {0, 2, 6}, {0, 5, 6}, {1, 3, 2, 5}, {1, 3, 2, 6}, {1, 3, 5, 6}, {1, 4, 2, 5}, {1, 4, 2, 6}, {1, 4, 5, 6}, {3, 4, 2, 5}, {3, 4, 2, 6}, {3, 4, 5, 6} }.

Definition 6. Binary Forest Quorums. *There are n nodes, which are divided into 2k groups, denoted as groups, S_0, S_1, S_2, ..., and S_{2k-1}. For the nodes in each group S_i, $0 \le i \le (2k-1)$, let R_i be the binary tree quorum for 1-mutual exclusion. When $k \ge 1$, a binary forest quorum Q contains any two quorums from R_0, R_1, R_2, ..., and R_{2k-1}.*

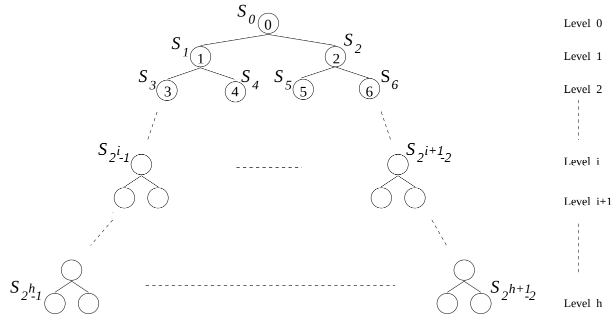

Figure 1: A binary tree

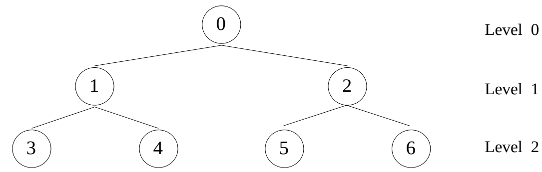

Figure 2: A binary tree for 1-mutual exclusion with $n = 7$

Example 3: For the binary forest as shown in Figure 3, the set R of binary forest quorums for 2-mutual exclusion is as follows: R = { {1, 5, 2, 7}, {1, 5, 2, 8}, {1, 5, 7, 8}, {1, 5, 3, 9}, {1, 5, 3, 10}, {1, 5, 9, 10}, {1, 5, 4, 11}, {1, 5, 4, 12}, {1, 5, 11, 12}, {1, 6, 2, 7}, {1, 6, 2, 8}, {1, 6, 7, 8}, {1, 6, 3, 9}, {1, 6, 3, 10}, {1, 6, 9, 10}, {1, 6, 4, 11}, {1, 6, 4, 12}, {1, 6, 11, 12}, {5, 6, 2, 7}, {5, 6, 2, 8}, {5, 6, 7, 8}, {5, 6, 3, 9}, {5, 6, 3, 10}, {5, 6, 9, 10}, {5, 6, 4, 11}, {5, 6, 4, 12}, {5, 6, 11, 12}, {2, 7, 3, 9}, {2, 7, 3, 10}, {2, 7, 9, 10}, {2, 7, 4, 11}, {2, 7, 4, 12}, {2, 7, 11, 12}, {2, 8, 3, 9}, {2, 8, 3, 10}, {2, 8, 9, 10}, {2, 8, 4, 11}, {2, 8, 4, 12}, {2, 8, 11, 12}, {7, 8, 3, 9}, {7, 8, 3, 10}, {7, 8, 9, 10}, {7, 8, 4, 11}, {7, 8, 4, 12}, {7, 8, 11, 12}, {3, 9, 4, 11}, {3, 9, 4, 12}, {3, 9, 11, 12}, {3, 10, 4, 11}, {3, 10, 4, 12}, {3, 10, 11, 12}, {9, 10, 4, 11}, {9, 10, 4, 12}, {9, 10, 11, 12} }. Totally, R contains 54 quorums.

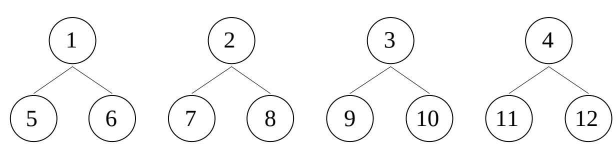

Figure 3: A binary forest for 2-mutual exclusion with $n = 12$

4.2 Correctness

In this section, we prove that the set of the binary forest quorums for k-mutual exclusion is a k-coterie. Here, we will refer to such a k-coterie as the binary forest coterie.

Lemma 1. *The set of the K-majority quorums is a k-coterie [11, 20].*

Lemma 2. *The set of the binary tree quorum is a 1-coterie [1].*

Lemma 3. *Let U_1 and U_2 be two nonempty sets of nodes such that $U_1 \cap U_2 = \emptyset$, and $x \in U_1$. Let $U = (U_1 - x) \cup U_2$. The coterie join operation $\otimes_x$ is defined as $\bar{Z} = \bar{X} \otimes_x \bar{Y} = \{ CT_x(X, Y) \mid X \in \bar{X}, Y \in \bar{Y} \}$, where $\bar{X}$ is a k-coterie under U_1, $\bar{Y}$ is a 1-coterie under U_2, and*

$$CT_x(X,Y) = \begin{cases} (X - \{x\}) \cup Y & \text{if } x \in X \\ X & \text{otherwise} \end{cases}$$

Then, $\bar{Z}$ is a k-coterie under U [9].

Theorem 1. *The set of the binary forest quorums for k-mutual exclusion is a k-coterie.*

Proof. Based on the definition of binary forest quorums, there are n nodes, which are divided into $2k$ groups, denoted as groups, S_0, S_1, ..., and S_{2k-1}. Therefore, $S_i \cap S_j = \emptyset$, $0 \leq i, j \leq 2k - 1$, and $i \neq j$. For the nodes in each group S_i, $0 \leq i \leq (2k - 1)$, let R_i be the binary tree quorum under S_i. When $k \geq 1$, a binary forest quorum Q contains any two quorums from R_0, R_1, ..., R_{2k-1}. Let $U_1 = \{g_0, g_1, ..., g_{2k-1}\}$ and $U_1 \cap S_i = \emptyset$, $0 \leq i \leq 2k-1$; let $\bar{X}$ be the set of the k-majority quorums under U_1, the quorum size of $\bar{X}$ is 2 ($=\lceil \frac{2k+1}{k+1} \rceil$). Moreover, let $x = g_0 \in U_1$, $U_2 = S_0$, and $\bar{Y}$ be the set of the binary tree quorums under S_0, i.e., set R_0.

Based on **Lemma 3**, we have that $\bar{Z}$ is a k-coterie under (U_1 - $\{g_0\}$) $\cup$ S_0, since $\bar{X}$ is a k-coterie based on **Lemma 1**, and $\bar{Y}$ is a 1-coterie based on **Lemma 2**. Note that, based on the definition of $CT_x(X, Y)$ of **Lemma 3**, we have a binary tree quorum under S_0, which is R_0; we replace g_0 in $\bar{X}$ with R_0 and forms a new quorum in $\bar{Z}$ under $(U_1 - \{g_0\}) \cup S_0$. Therefore, $\forall$ ($Q_x \in \bar{X}$ under U_1 and $g_0 \in Q_x$), we have a new quorum $(R_0 \cup (Q_x - \{g_0\})) \in \bar{Z}$ under $(U_1 - \{g_0\}) \cup S_0$, where R_0 is the binary tree quorum under S_0. That is, a quorum in $\bar{Z}$ under $(U_1 - \{g_0\}) \cup S_0$, contains any two subset from R_0, g_1, ..., and g_{2k-1}, where R_0 is a binary tree quorum under S_0. In the same way, we can replace U_2 with every S_i, where $1 \leq i \leq 2k - 1$. Therefore, a quorum in $\bar{Z}$ under $S_0 \cup S_1 \cup ... \cup S_{2k-1}$ ($=((U_1 - \{g_0, g_1, ..., g_{2k-1}\}) \cup S_0 \cup S_1 \cup ... \cup S_{2k-1})$), contains any two subsets from R_0, R_1, ..., and R_{2k-1}, where R_i be a binary tree quorum under S_i, $0 \leq i \leq 2k - 1$. Consequently, the set of binary forest quorums for k-mutual exclusion is a k-coterie. □

4.3 Availability of the Binary Forest Quorums

In this section, we first analyze the availability of the binary tree quorums for 1-mutual exclusion [1] and then the binary forest quorums for k-mutual exclusion. Here, we assume that all the nodes have the same up-probability p, which is the probability that a single node is up operational.

For the binary tree quorum strategy, the availability of a binary tree is the probability that at least one binary tree quorum can be formed from the binary tree [1]. Thus, the availability of a binary tree is the probability that

1. the root is operational and a tree quorum can be formed from the left subtree, or

2. the root is operational and a tree quorum can be formed from the right subtree, or

3. a tree quorum can be formed from the left subtree and a tree quorum can be formed from the right subtree.

Let $AVB(h)$ be the function evaluating the probability of a binary tree with $(h + 1)$ level. If a binary tree consists of only one node, it degenerates to a central controller and the availability of the availability of itself, i.e., $AVB(0) = p$. Thus by the above conditions (1), (2), and (3), we can get the condition $AVB(h) = 2p \times AVB(h-1) \times (1 - AVB(h-1)) + AVB(h-1)^2$.

Next, for the availability of the binary forest quorum strategy, let *(k,l)*-availability, $1 \leq l \leq k$, be the probability that l pairwise disjoint quorums of a k-coterie can be formed successfully; it is used as a measure for the fault-tolerant ability of a solution using k-coterie.

Let *AV(h, l)* be the function evaluating the probability that l pairwise disjoint quorums under binary forest can be formed simultaneously. The function $AV(h, l)$ has the following two boundary conditions:

1. $AVB(0) = p$ and $AVB(j) = 2p \times AVB(j-1) \times (1 - AVB(j-1)) + AVB(j-1)^2$.

2. $AV(h, l) = \sum_{m=2*l}^{2k} C(2k, m) \times AVB(h)^m \times (1 - AVB(h))^{2k-m}$, where $h = \lceil \log_2 \frac{n}{2k} \rceil$.

5. A COMPARISON

In this section, we make a comparison of the binary forest quorum, k-majority, cohorts, and DIV quorum strategies in terms of availability and quorum size,

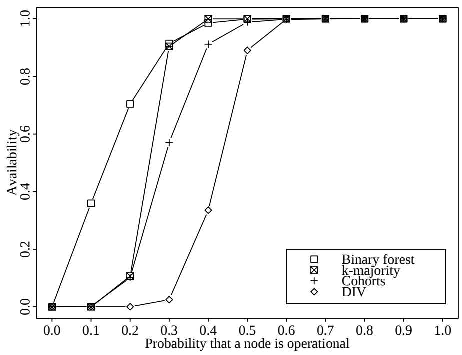

Figure 4: A comparison of the availability of the binary forest, k-majority, cohorts, and DIV strategies with n = 120 ($l = 1$)

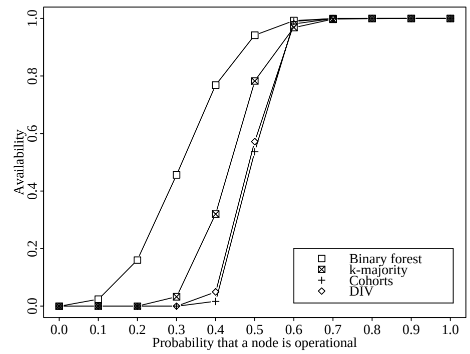

Figure 5: A comparison of the availability of the binary forest, k-majority, cohorts, and DIV strategies with n = 120 ($l = 2$)

where we assume that the system has a fully connected network topology and no communication failure will occur. However, a node failure can occur. (Note that, here, we assume that a failed node simply stops execution (i.e., a fail-stop system). That is, no Byzantine failure occurs.)

Figures 4, 5, 6, and 7 show a comparison of the availability of the binary forest, k-majority, cohorts, DIV quorums strategies with $n = 120$, and $l = 1$, 2, 3, and 4, respectively. For this comparison, there are 8 groups in binary forest quorum strategy and inside each group, the binary tree quorum is of level 4, and we let Coh(4, 15) = (C_1, C_2, ..., C_{15}) where $C_1 = 4$, $C_i = 8$, $2 \leq i \leq 11$, and $C_j = 9$, $12 \leq j \leq 15$. The observed results from Figures 4, 5, 6, and 7 are summarized in Table 1, where the binary forest quorum strategy is denoted as BF. From this table, we show that the availability of the binary forest quorum strategy is always better than that of the cohorts and DIV strategies, when $l = 1$; the availability of the binary forest quorum strategy is always the highest one among these four strategies, when $l = 2$; the availability of the binary forest quorum strategy is always better than that of the cohorts and k-majority strategies, when $l = 3$; the availability of the binary forest quorum strategy is always better than that of k-majority strategy, when l = 4.

Table 2 shows a comparison of these four k-mutual exclusion strategies imposing logical structures. The first two criteria are the quorum sizes in the best and worst cases, respectively. The number of messages required to construct a quorum is proportional to the size of the quorums. The quorum size of binary forest varies from $2\lceil \lg_2 \frac{n}{2k} \rceil$ to $\lceil \frac{n}{2k} \rceil$+1 as the number of node failures

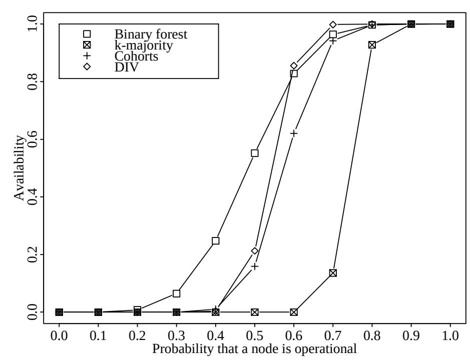

Figure 6: A comparison of the availability of the binary forest, k-majority, cohorts, and DIV strategies with n = 120 ($l = 3$)

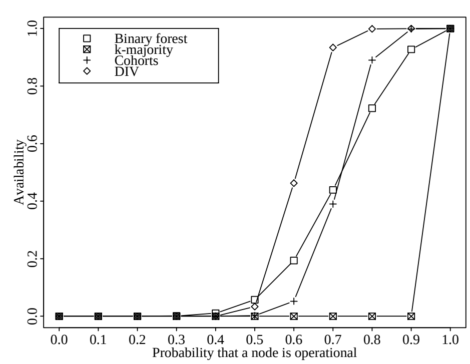

Figure 7: A comparison of the availability of the binary forest, k-majority, cohorts, and DIV strategies with n = 120 ($l = 4$)

Table 1: A comparison of the availability of the binary forest, k-majority, cohorts, and DIV strategies with n = 120

l	P	The Availability
1	$p < 0.35$	BF > k-majority > cohorts > DIV
	$p \geq 0.35$	k-majority > BF > cohorts > DIV
2	$p < 0.59$	BF > k-majority > DIV > cohorts
	$p \geq 0.59$	BF > k-majority > cohorts > DIV
3	$p < 0.59$	BF > DIV > cohorts > k-majority
	$p \geq 0.59$	DIV > BF > cohorts > k-majority
4	$p < 0.51$	BF > DIV > cohorts > k-majority
	$0.51 \leq p < 0.72$	DIV > BF > cohorts > k-majority
	$p \geq 0.72$	DIV > cohorts > BF > k-majority

Table 2: A comparison of four k-mutual exclusion strategies in terms of quorum size

	Binary Forest	Cohorts*
quorum size (best case)	$2\lceil \lg_2 \frac{n}{2k} \rceil$	2 or k**
quorum size (worst case)	$\lceil \frac{n}{2k} \rceil + 1$	l
fully distributed?	no	no
fault tolerance (best case)	$n - 2k\lceil \lg_2 \frac{n}{2k} \rceil$	$n - ks + \frac{k(k-1)}{2}$
fault tolerance (worst case)	$2k(\lceil \lg_2 \frac{n}{2k} \rceil - 1)$	$s - k + 1$

	K-majority	DIV
quorum size (best case)	$\lceil \frac{n+1}{k+1} \rceil$	$\lceil \frac{n+k}{2k} \rceil$
quorum size (worst case)	$\lceil \frac{n+1}{k+1} \rceil$	$\lceil \frac{n+k}{2k} \rceil$
fully distributed?	yes	yes
fault tolerance (best case)	$n - k\lceil \frac{n+1}{k+1} \rceil$	$n - k\lceil \frac{n+k}{2k} \rceil$
fault tolerance (worst case)	$n - k\lceil \frac{n+1}{k+1} \rceil$	$n - k\lceil \frac{n+k}{2k} \rceil$

* $Coh(k,l) = (C_1, C_2, ..., C_l), |C_1| = k, |C_i| = s, i > 1$.
** 2 when $k = 1$, or k when $k > 1$.

is increased. Because in the binary forest quorums, n nodes are divided into $2k$ groups. Between groups, we apply the k-majority strategy, and inside each group, we apply the binary tree quorum for 1-mutual exclusion. No matter in the best case or the worst case, the quorum size of k-majority strategy with $2k$ groups is always 2 groups. In the best case, the quorum size inside each group is $\lceil \lg_2 \frac{n}{2k} \rceil$; therefore, the quorum size of the binary forest is $2\lceil \lg_2 \frac{n}{2k} \rceil$. In the worst case, the quorum size inside each group is ($\lceil \frac{n}{2k} \rceil + 1$) / 2; therefore, the quorum size of the binary forest is $\lceil \frac{n}{2k} \rceil + 1$. Note that in the binary forest strategy, n nodes are divided into $2k$ binary trees. In the best case, the quorum size in each binary tree is $\lceil \lg_2 \frac{n}{2k} \rceil$; therefore, the quorum size of the binary forest strategy is $2\lceil \lg_2 \frac{n}{2k} \rceil$. In the worst case, which occurs when the node fails starting from the root to the leaf, and from the left to the right, the quorum size in each binary tree is ($\lceil \frac{n}{2k} \rceil + 1$) / 2; therefore, the quorum size of the binary forest strategy is $\lceil \frac{n}{2k} \rceil + 1$. Note that in the binary forest strategy, n nodes are divided into the k binary trees. In each binary tree, the number of the terminal nodes is equal to (the number of nonterminal nodes +1). Therefore, the number of terminal nodes is equal to (the total number of the nodes of the binary trees + 1)/2. Moreover, in the worst case, the quorum size of the binary forest strategy is 2*(the sum of the number of the terminal nodes in every binary tree). Consequently, in the worst case, the quorum size of the binary forest strategy is (the total number of the nodes + $2k$)/2k. The quorum size of cohort [10] varies from 2 (when $k = 1$) or k (when $k > 1$) to $l = \frac{n-k}{s} + 1$, for a cohort structure $Coh(k,l) =< k, s, ..., s >, l >> s$. The quorum size of k-majority strategy is always $\lceil \frac{n+1}{k+1} \rceil$ [11], and the quorum size of DIV strategy is always $\lceil \frac{n+k}{2k} \rceil$ [2].

The third criteria in Table 2 is whether the strategy is a fully distributed one. All of these four strategies are fully distributed ones. The last two criteria are the number of failed nodes which does not halt the system and such that at most k nodes can simultaneously access their critical section in the best case and worst case. While in the best case, all these strategies can be fault-tolerant up to all node failure except those nodes which have already constructed k quorums. In the best case, the cohorts strategy can be fault-tolerant up to $(n - ks + \frac{k(k-1)}{2})$ node failures when $Coh(k,l) = (C_1, C_2, ..., C_l)$, $|C_1| = k$, and $|C_i| = s$, $i > 1$ [10]. Note that, in the best case, $|Q_1| = s-(k-1)$, $|Q_2| = s-(k-2)$, ..., $|Q_{k-1}| = s-1$, and $|Q_k| = s$ in the cohorts strategy. While in the worst case, the binary forest strategy can be fault-tolerant up to $k(\lceil \lg_2 \frac{n}{k} \rceil - 1)$ node failures and the cohorts strategy can be fault-tolerant up to $(s - k + 1)$ node failures. While in the

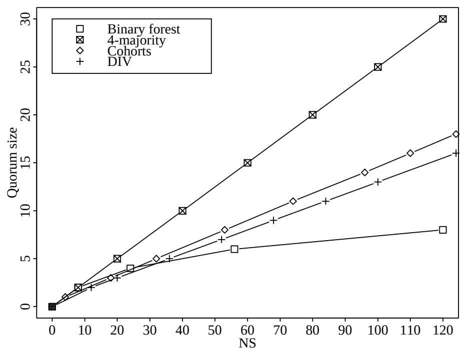

Figure 8: A comparison of the quorum size of binary forest, 4-majority, cohorts, and DIV when no node failure occurs

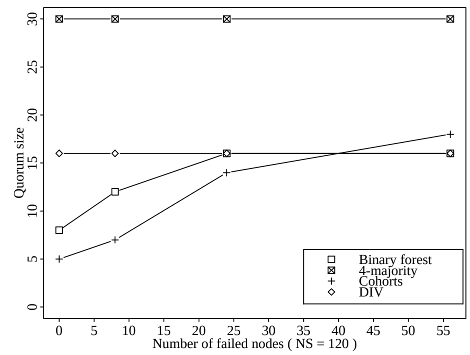

Figure 9: A comparison of the quorum size of binary forest, 4-majority, cohorts, and DIV when node failures occur in the worst case $NS = n = 120$ ($l = 4$)

worst case, the k-majority strategy can not be fault-tolerant to any node failure and the DIV strategy can be fault-tolerant up to $(n - k\lceil\frac{n+k}{2k}\rceil)$ node failures.

Figure 8 shows a comparison of the quorum size of these four strategies for 4-mutual exclusion when no node failure occurs. From this figure, we observe that the quorum size of these four strategies in a decreasing order is 4-majority > cohorts > DIV > binary forest quorum, when $n > 30$. That is, the quorum size of the binary forest strategy is always the smallest one among these four strategies, when $n > 30$. Figure 9 shows a comparison of the quorum size of these four strategies for 4-mutual exclusion when node failures occur in the worst case with $NS = 120$ ($l = 4$). From this figure, we observe that the quorum size of these four strategies in a decreasing order is 4-majority > DIV > binary forest > cohorts, when the number of failed nodes is less than 40; the quorum size of these four strategies in a decreasing order is 4-majority > cohorts > DIV = binary forest, when the number of failed nodes is greater than 40. That is, in the worst case, the quorum size of the binary forest quorum strategy is always smaller than that of 4-majority and DIV strategies.

6. CONCLUSION

In this paper, we have proposed a strategy called binary forest quorums for k-mutual exclusion, which imposes a logical binary forest structure on the network. In general, in the binary forest quorum strategy, n nodes are divided into the $2k$ groups. Between groups, we have applied the k-majority strategy, and inside each group, we have applied the binary tree quorums for 1-mutual exclusion. Therefore, the proposed strategy can be considered as a hybrid approach which contains the k-majority and the binary tree quorums for 1-mutual exclusion. The quorum size constructed from the strategy is $2\lceil\lg_2 \frac{n}{2k}\rceil$ in the best case and is $(\lceil\frac{n}{2k}\rceil + 1)$ in the worst case. From our performance analysis, we have shown that the binary forest quorum strategy can provide a higher availability than k-majority, cohorts, and DIV strategies almost all the time. How to extend the binary forest quorum strategy to tolerate even more node failures is the future research direction.

References

[1] D. Agrawal and A. EL Abbadi, "The Generalized Tree Quorum Protocol: An Efficient Approach for Managing Replicated Data," *ACM Trans. on Database Systems*, Vol. 17, No. 4, pp. 689-717, Dec. 1992.

[2] D. Agrawal, O. Egecioglu, and A. E. Abbadi, "Analysis of Quorum-Based Protocols for Distributed (k+1)-Exclusion", *IEEE Trans. on Computers*, Vol. 8, No. 5, pp. 533-537, May 1997.

[3] J. H. Anderson, Y. J. Kim, "A New Fast-Path Mechanism for Mutual Exclusion," *Distributed Computing*, Vol. 14, No. 1, pp. 17-29, 2001.

[4] R. Baldoni and B. Ciciani, "Distributed Algorithms for Multiple Entries to a Critical Section with Priority," *Information Processing Letters*, Vol. 50, pp. 165-172, 1994.

[5] A. Bouabdallah, "On Mutual Exclusion in Faulty Distributed Systems," *ACM Operating Systems Reviews*, Vol. 28, No. 1, pp. 80-87, Jan. 1994.

[6] G. Cao, M. Singhal, Y. Deng, N. Rishe and W. Sun, "A Delay-Optimal Quorum-Based Mutual Exclusion Scheme with Fault-Tolerance Capability," *Proc. of IEEE Int. Conf. on Data Engineering*, pp. 444-451, 1998.

[7] Y. I. Chang, "A Simulation Study on Distributed Mutual Exclusion," *Journal of Parallel and Distributed Computing,* Vol. 33, No. 2, pp. 107-121, March 1996.

[8] B. H. Chen, "Logical Structure-Based Strategies for K-Mutual Exclusion in Distributed Systems," *Master Thesis*, Dept. of Applied Math., National Sun Yat-Sen Univ., Kaohsiung, Taiwan, R.O.C., June 1997.

[9] J. R. Jiang and S. T. Huang, "Obtaining Nondominated K-Coteries for Fault-Tolerant Distributed K-Mutual Exclusion," *Proc. of ICPAD*, pp. 582-587, 1994.

[10] J. R. Jiang, S. T. Huang, and Y. C. Kuo, "Cohorts Structures for Fault-Tolerant K Entries to a Critical Section," *IEEE Trans. on Computers*, Vol. 46, No. 2, pp. 222-228, Feb. 1997.

[11] H. Kakugawa, S. Fujita, M. Yamashita,i and T. Ae, "Availability of K-Coterie," *IEEE Trans. on Computers*, Vol. 42, No. 5, pp. 553-558, May 1993.

[12] N. Kobayashi, T. Tsuchiya and T. Kikuno, "Minimizing the Mean Delay of Quorum-Based Mutual Exclusion Schemes," *The Journal of Systems and Software*, Vol. 58, No. 1, pp. 1-9, 2001.

[13] Y. C. Kuo and S. T. Huang, "Recognizing Nondominated Coteries and wr-Coteries by Availability," *IEEE Trans. on Parallel and Distributed Systems*, Vol. 9, No. 8, pp. 721-728, August 1998.

[14] M. Naimi, "Distributed Algorithm for K-Entries to Critical Section Based on the Directed Graphs", *ACM Operating Systems Reviews*, Vol. 27, No. 4, pp. 67-75, Oct. 1993.

[15] M. L. Neilsen and M. Mizuno, "Nondominated k-coteries for multiple mutual exclusion," *Information Processing Letters*, Vol. 50, pp. 247-255, 1994.

[16] K. Raymond, "A Distributed Algorithm for Multiple Entries to a Critical Section," *Information Processing Letters*, Vol. 30, pp. 189-193, 1989.

[17] M. Singhal, "A Taxonomy of Distributed Mutual Exclusion," *Journal of Parallel and Distributed Computing*, Vol. 18, pp. 94-101, 1993.

[18] P. K. Srimani and R. L. N. Reddy, "Another Distributed Algorithm for Multiple Entries to a Critical Section," *Information Processing Letters* , Vol. 41, pp. 51-57, 1992.

[19] S. Wang and S. D. Lang, "A Tree-Based Distributed Algorithm for the K-Entry Critical Section Problem," *Proc. of ICPAD,* pp. 592-597, 1994.

[20] S. M. Yuan and H. K. Chang, "Comments on Availability of k-Coterie," *IEEE Trans. on Computers,* Vol. 43, No. 12, p. 1457, Dec. 1994.