Transversal Merge Operation: A Nondominated Coterie Construction Method for Distributed Mutual Exclusion

Takashi Harada, *Member*, *IEEE Computer Society*, and
Masafumi Yamashita, *Member*, *IEEE Computer Society*

Abstract—A coterie is a set of subsets (called quorums) of the processes in a distributed system such that any two quorums intersect with each other and is mainly used to solve the mutual exclusion problem in a quorum-based algorithm. The choice of a coterie sensitively affects the performance of the algorithm and it is known that nondominated (ND) coteries achieve good performance in terms of criteria such as availability and load. On the other hand, grid coteries have some other attractive features: 1) A quorum size is small, which implies a low message complexity, and 2) a quorum is constructible on the fly, which benefits a low space complexity. However, they are not ND coteries unfortunately. To construct ND coteries having the favorite features of grid coteries, we introduce the transversal merge operation that transforms a dominated coterie into an ND coterie and apply it to grid coteries. We call the constructed ND coteries ND grid coteries. These ND grid coteries have availability higher than the original ones, inheriting the above desirable features from them. To demonstrate this fact, we then investigate their quorum size, load, and availability, and propose a dynamic quorum construction algorithm for an ND grid coterie.

Index Terms—Coteries, distributed systems, grid structures, mutual exclusion algorithms, nondominatedness, quorum consensus, transversals.

1 Introduction

Let U be the set of processes forming a distributed system. A *coterie* is a set of subsets (called *quorums*) of U that satisfies *Intersection Property*—any two quorums intersect with each other, and *Minimality*—no quorum contains another quorum as a subset. Coteries are mainly used for solving the mutual exclusion problem: Let $\mathcal{Q}$ be a coterie. We prepare a single token named $\mathtt{permission}_v$ for each process $v \in U$ and place it at v initially.

1. (Request for Critical Section) When a process u wishes to enter the critical section, it selects a quorum $Q \in \mathcal{Q}$ and requests $\mathtt{permission}_v$ to each process $v \in Q$.
2. (Processing Request) Upon receiving the request from u, v sends $\mathtt{permission}_v$ to u, as soon as it holds it.
3. (Entering Critical Section) Upon receiving $\mathtt{permission}_v$ from every process $v \in Q$, it enters the critical section.
4. (Leaving Critical Section) Upon leaving the critical section, u returns $\mathtt{permission}_v$ to each process $v \in Q$.

Then, Intersection Property guarantees mutual exclusion. The above description, however, does not include mechanisms to avoid deadlocks and starvations. Maekawa, hence, proposed a complete mutual exclusion algorithm by exploring this idea in detail [13].[1]

A coterie $\mathcal{P}$ is said to *dominate* another coterie $\mathcal{Q}$ if, for any $Q \in \mathcal{Q}$, there is a $P \in \mathcal{P}$ such that $P \subseteq Q$. A coterie $\mathcal{P}$ is said to be *nondominated* (*ND*, for short) if there is no coterie $\mathcal{Q}$ ($\neq \mathcal{P}$) that dominates $\mathcal{P}$. Intuitively, ND coteries can implement mutual exclusion in a more efficient way than dominated ones. They are also characterized by their maximality; there is no coterie that includes an ND coterie as a proper subset. By these properties, Maekawa's algorithm adopting an ND coterie shows good performance under such criteria as availability [18], robustness against network 2-partition [2], load [16], and communication delay [5].

To make use of these attractive features, many construction methods have been proposed. They include the majority coteries [6], [7], [21], the tree coteries [1], the Lovász coteries [14], and the CWlog [19]. Several coterie transformation methods have also been proposed 1) to construct a large ND coterie from simple ones, 2) to enumerate ND coteries, and 3) to obtain a new ND coterie with better performance (e.g., [3], [7], [8], [9], [15]).

The selection of a coterie is an important implementation issue concerning Maekawa's algorithm. Another important implementation issue is the way of storing the coterie. When we choose an ND coterie, the problem becomes more serious since its size is usually larger than a dominated one because of its maximality. If processes can construct a

1. See also [20] for a bug in Maekawa's algorithm and its improvement.

- T. Harada is with the Graduate School of Management, Hiroshima University, 1-1-89 Higashi Senda, Naka-ku, Hiroshima, 730-0053 Japan. E-mail; haradat@mgt.hiroshima-u.ac.jp.
- M. Yamashita is with the Department of Computer Science and Communication Engineering, Kyushu University, 6-10-1 Hakozaki, Higashi-ku, Fukuoka, 812-8581 Japan. E-mail: mak@csce.kyushu-u.ac.jp.

Manuscript received 25 May 2003; revised 1 Dec. 2003; accepted 12 Aug. 2004; published online 20 Dec. 2004
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-0081-0503.

quorum on the fly whenever it is necessary, instead of maintaining the coterie, then it definitely contributes to the reduction of space complexity. Grid coteries are proposed to this end based on logical grid structures of processes [4], [10], [11], [12], [13], [16], [17]. Grid coteries discussed in [4], which we will call C-Grid coteries in this paper, virtually align the processes in a two-dimensional array. Then, a quorum consists of a one full row and an element from each row. A quorum is dynamically constructible, provided that processes know the logical grid structure.

Another advantage of the grid coteries is their quorum size. The number of messages necessary to enter the critical section is roughly proportional to the size of a quorum in Maekawa's algorithm. For most of the grid coteries, their quorum sizes are bounded by $O(\sqrt{N})$, where $N = |U|$ is the number of processes. Since $\sqrt{N}$ approximately meets a lower bound on the quorum size under the *fully distributed condition* [13], they are nearly optimal.

However, the grid coteries also have some drawbacks. To the best of our knowledge, all of the grid coteries are dominated except some very special cases. Moreover, the availability of any C-Grid coterie approaches 0 as N increases [10].

Peleg and Wool [19] proposed a coterie construction method in order to overcome these drawbacks. By generalizing the grid structure, they introduced a logical structure called a *Crumbling Wall* (*CW*, for short), in which processes are arranged into rows like grid coteries, but each row may contain a different number of processes, unlike the grid coteries. On the CW, a quorum is defined by a one full row and an element from each row below the full row. The set of all quorums is called a *CW coterie*. A CW coterie is called a *CWlog coterie* if it has only one process in the top row and the number of processes in each row increases logarithmically from top to bottom. They showed that every CWlog coterie is ND and has optimal load. They also showed that its availability approaches asymptotically to 1 as N grows.

The purpose of this paper is the same as [19]; relaxing the disadvantages of the grid coteries without losing their advantages. We take a more general approach than theirs.[2] Our method is based on the following facts: A *transversal* of a coterie $\mathcal{P}$ under U is a subset $T \subseteq U$ that intersects with all quorums in $\mathcal{P}$. Assume that $\mathcal{P}$ is dominated. Then, $\mathcal{P} \cup \{T\}$ is a coterie that dominates $\mathcal{P}$ if a transversal T of $\mathcal{P}$ is not a superset of any quorum in $\mathcal{P}$. Hence, an ND coterie seems to be constructible simply by adding all such transversals. However, this is not the case since two transversals need not intersect each other and the Intersection Property may be violated. Hence, given the set $\mathcal{T}$ of all transversals of $\mathcal{P}$, we need to select appropriate transversals from $\mathcal{T}$ to construct an ND coterie.

To this end, we first introduce the *transversal merge operation* (TM operation, for short), which uses another ND coterie $\mathcal{Q}$ for choosing appropriate transversals. The TM operation enables us to construct an ND coterie from a dominated coterie, provided that all of the transversals are given. We then derive a necessary and sufficient condition for a subset $T \subseteq U$ to be a transversal for each of the following four grid coteries and construct ND grid coteries by using the TM operation:[3]

C-Grid coterie [4]: The processes in one full row and a process from each row form a quorum.

C*-Grid coterie: In addition to the quorums of the C-Grid coterie, the processes in one full column and a process from each column also form a quorum.

M-Grid coterie [13]: The processes in one full row and one full column form a quorum.

T-Grid coterie [17]: The processes in one full row and a process from each row below the full row form a quorum.

We next investigate advantages of the constructed ND grid coteries. Although our investigations will be done only for ND C-Grid coteries, i.e., the ND coteries constructed from C-Grid coteries, the other ND grid coteries have similar properties.

First, we show that the quorum size of ND C-Grid coteries is bounded by $O(\sqrt{N} + K)$, where K is the maximum quorum size of the ND coterie $\mathcal{Q}$ used in the TM operation. We can thus keep the quorum size as small as C-Grid coteries by choosing a $\mathcal{Q}$ that satisfies $K \leq \sqrt{N}$. We then show that the availability of an ND C-Grid coterie converges to that of $\mathcal{Q}$ as N grows. Recall that the majority coterie is optimal in terms of the availability, provided that all processes have the same failure probability less than 0.5 (and links never fail) [18]. Thus, what we called a *C-Majority coterie* constructed from a C-Grid coterie and a majority coterie under the set of processes in a row enjoys both small quorum sizes and good availability. We also present a quorum construction algorithm for C-Majority coteries which constructs a quorum on the fly. We finally apply our method to the CW coterie. Our method can be regarded as a generalization of an ND coterie construction method in [19].

We would like to make a remark for fairness. The time complexity of the ND-ness test for a given coterie is likely to be co-NP-complete (see, e.g., [3], [9]), although it is still open. On the other hand, determining all minimal traversals of a coterie is equivalent to the ND-ness test [9]. Hence, our method is applicable only when the calculation of all minimal transversals is tractable, like the grid coteries.

The rest of this paper is organized as follows: After giving basic definitions, Section 2 shows some properties on transversals of dominated coteries. In Section 3, we introduce the TM operation. In Section 4, we characterize the transversals of C-Grid coteries and then construct ND C-Grid coteries. In Section 5, we analyze the performance of ND C-Grid coteries. Section 6 presents dynamic quorum construction algorithms for ND C-Grid coteries. We then discuss Crumbling Wall coteries in Section 7. Section 8 concludes the paper with discussions about the possibility of applying our method to other quorum systems.

2. We will make the relation between their and our approaches clear when we discuss CW coteries in Section 7.

3. We will present the results with proofs only for C-Grid coteries and just summarize the counterparts for the other grid coteries in the Appendix without proofs since they are obviously obtainable by similar arguments.

2 PRELIMINARIES

2.1 ND Coteries

We start this section with defining coteries. Let U be the set of all processes in a distributed system. A *coterie* under U is a set $\mathcal{Q}$ of nonempty subsets of U satisfying the following properties:

Intersection Property: $\forall P, Q \in \mathcal{Q}[P \cap Q \neq \emptyset]$.

Minimality: $\forall P, Q \in \mathcal{Q}[P \not\subset Q]$.

An element of a coterie is called a *quorum*.

Example 1. Let $U = \{1, 2, 3\}$. Then, $\mathcal{P} = \{\{1, 2\}, \{1, 3\}\}$ is a coterie under U. A singleton $\mathcal{Q} = \{\{1\}\}$ is also a coterie known as a *singleton coterie*. Another well-known coterie, *3-majority coterie*, is $\mathcal{R} = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$.

Garcia-Molina and Barbara [7] introduced the concept of *nondominatedness*: Let $\mathcal{P}$ and $\mathcal{Q}$ be coteries under U. $\mathcal{P}$ is said to *dominate* $\mathcal{Q}$ if $\mathcal{P} \neq \mathcal{Q}$ and, for each quorum $Q \in \mathcal{Q}$, there is a quorum $P \in \mathcal{P}$ such that $P \subseteq Q$. If there is no coterie that dominates $\mathcal{P}$, then $\mathcal{P}$ is said to be *nondominated* (*ND*, for short).

Example 2. Consider the three coteries in Example 1. Both of $\mathcal{Q}$ and $\mathcal{R}$ dominate $\mathcal{P}$. But, $\mathcal{Q}$ and $\mathcal{R}$ do not dominate each other. They are well-known ND coteries.

In the rest of this paper, we use the following notations: For a subset S of U, let $\overline{S} = U \setminus S$, which is the complement of S. Let $\mathcal{S}$ be a set of nonempty subsets of U. We denote, by $MinSet(\mathcal{S})$, the subset of $\mathcal{S}$ obtained from $\mathcal{S}$ by removing each element if its proper subset is in $\mathcal{S}$. Hence, Minimality holds for $MinSet(\mathcal{S})$.

Theorem 1 [7]. *Let $\mathcal{Q}$ be a coterie under U. $\mathcal{Q}$ is ND if and only if, for all $S \subseteq U$, exactly one of S or $\overline{S}$ contains some quorum of $\mathcal{Q}$ as a subset.*

2.2 Transversals of Coteries

Definition 1. *Let $\mathcal{Q}$ be a coterie under U. A* transversal *of $\mathcal{Q}$ is a set $T \subseteq U$ such that $T \cap Q \neq \emptyset$ for all $Q \in \mathcal{Q}$. A* minimal transversal *is a transversal T such that $T' \not\subset T$ for any transversal T' of $\mathcal{Q}$. We denote the set of all minimal transversals of $\mathcal{Q}$ by $Tr(\mathcal{Q})$.*

Ibaraki and Kameda [9] characterized ND coteries in terms of Boolean functions. We rephrase it in our notations.

Theorem 2 [9]. *Let $\mathcal{Q}$ be a coterie under U. Then, $\mathcal{Q}$ is ND if and only if $\mathcal{Q} = Tr(\mathcal{Q})$.*

Proposition 1. *Let $\mathcal{Q}$ be a dominated coterie under U. Then, there is a transversal $T \subseteq U$ of $\mathcal{Q}$ such that $Q \not\subseteq T$ for all $Q \in \mathcal{Q}$.*

Proof. Suppose that each transversal T of $\mathcal{Q}$ contains a quorum $Q \in \mathcal{Q}$ as a subset and derive a contradiction. Every quorum $Q \in \mathcal{Q}$ is a transversal of $\mathcal{Q}$ by definition. If there is a $T \in Tr(\mathcal{Q}) \setminus \mathcal{Q}$, a contradiction since there is a transversal $Q \in \mathcal{Q}$ such that $Q \subset T$. If there is a $Q \in \mathcal{Q} \setminus Tr(\mathcal{Q})$, again a contradiction since there is a transversal $T \subset Q$, which contains a quorum $Q' \in \mathcal{Q}$ as a subset. Hence, $\mathcal{Q} = Tr(\mathcal{Q})$, which contradicts Theorem 2.□

Proposition 2. *Let $\mathcal{Q}$ be a dominated coterie under U. If T is a transversal of $\mathcal{Q}$ such that $Q \not\subseteq T$ for all $Q \in \mathcal{Q}$, so is $\overline{T}$.*

Proof. If there is a quorum $Q \in \mathcal{Q}$ such that $Q \cap \overline{T} = \emptyset$, then $Q \subseteq T$, a contradiction. Hence, $\overline{T}$ is a transversal. If there is a quorum $Q \in \mathcal{Q}$ such that $Q \subseteq \overline{T}$, then $Q \cap T = \emptyset$, a contradiction. □

Proposition 3. *Let $\mathcal{Q}$ be a dominated coterie under U. If T is a transversal of $\mathcal{Q}$ such that $Q \not\subseteq T$ holds for all $Q \in \mathcal{Q}$, then $MinSet(\mathcal{Q} \cup \{T\})$ is a coterie that dominates $\mathcal{Q}$.*

Proof. Clear by definition. □

Example 3. Consider again a dominated coterie $\mathcal{P}$ in Example 1. The set of transversals of $\mathcal{P}$ is $\{\{1\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$. There are transversals $\{1\}$ and $\{2, 3\}$ that contain no quorum of $\mathcal{P}$ as a subset, as Proposition 1 guarantees. Observe that $MinSet(\mathcal{P} \cup \{\{1\}\}) = \{\{1\}\}$ and $MinSet(\mathcal{P} \cup \{\{2, 3\}\}) = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$. They both dominate $\mathcal{P}$.

Proposition 3 suggests a procedure for constructing an ND coterie from a given dominated coterie $\mathcal{P}$. Find a transversal T of $\mathcal{P}$ such that $P \not\subseteq T$ holds for all $P \in \mathcal{P}$, and construct $\mathcal{Q} = \mathcal{P} \cup \{T\}$. Then, $\mathcal{Q}$ dominates $\mathcal{P}$. If $\mathcal{Q}$ is still dominated, we repeat this procedure. Let $\mathcal{S}$ be the set of transversals that are added to construct an ND coterie from $\mathcal{P}$. Although a characterization of $\mathcal{S}$ seems to be difficult, we present a sufficient condition in the next section.

3 TRANSVERSAL MERGE OPERATION

Inspired by Ibaraki and Kameda [9, Formula (10)], this section introduces the *transversal merge operation*. Theorem 3 is a restatement of Formula (10) of [9] using our notation.

Theorem 3 [9]. *Let $\mathcal{R}$ be an ND coterie under U. Then, there are a coterie $\mathcal{P}$ and an ND coterie $\mathcal{Q}$ such that*

$$\mathcal{R} = MinSet(\mathcal{P} \cup \{Q \cup T | Q \in \mathcal{Q} \text{ and } T \in Tr(\mathcal{P})\}).$$

Definition 2. *Let $\mathcal{P}$ and $\mathcal{Q}$ be coteries under U and $\mathcal{T}$ be a set of transversals of $\mathcal{P}$. The* transversal merge operation *(*TM operation*, for short), $TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$, is defined as*

$$TM(\mathcal{P}, \mathcal{Q}, \mathcal{T}) = MinSet(\mathcal{P} \cup \{Q \cup T | Q \in \mathcal{Q} \text{ and } T \in \mathcal{T}\}).$$

Theorem 4. *$TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$ is a coterie under U. Moreover, if $\mathcal{Q}$ is ND and $MinSet(\mathcal{T}) = Tr(\mathcal{P})$, then $TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$ is an ND coterie under U.*

Proof. We first show that $TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$ is a coterie. Since Minimality holds by definition, we concentrate on showing Intersection Property. For any $R, R' \in TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$, we show $R \cap R' \neq \emptyset$. Without loss of generality, we can assume $R \notin \mathcal{P}$ since $\mathcal{P}$ is a coterie and by the symmetry. Let $R = Q \cup T$, where $Q \in \mathcal{Q}$ and $T \in \mathcal{T}$. If $R' \in \mathcal{P}$, then $R \cap R' \neq \emptyset$ since T is a transversal of $\mathcal{P}$. Otherwise, if $R' = Q' \cup T'$, where $Q' \in \mathcal{Q}$ and $T' \in \mathcal{T}$, then $R \cap R' \neq \emptyset$ since Q and Q' are quorums of $\mathcal{Q}$.

We now go on the second claim. To derive a contradiction, suppose that $TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$ is dominated, despite the facts that $\mathcal{Q}$ is ND and $MinSet(\mathcal{T}) = Tr(\mathcal{P})$. Since

$TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$ is a dominated coterie, by Theorem 1, there is an $S \subseteq U$ such that for any quorum $R \in TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$, both of $R \not\subseteq S$ and $R \not\subseteq \overline{S}$ hold. It is because there is no $S \subseteq U$ such that, for some quorum $R \in TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$, both $R \subseteq S$ and $R \subseteq \overline{S}$ hold.

We first show that S and $\overline{S}$ are transversals of $\mathcal{P}$. Let P be any quorum of $\mathcal{P}$. If P is a quorum of $TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$, then $P \not\subseteq S$ and $P \not\subseteq \overline{S}$ hold. Otherwise, if $P \notin TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$, then there is a quorum $R \in TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$ such that $R \subseteq P$ by definition. Since $R \not\subseteq S$ and $R \not\subseteq \overline{S}$, $P \not\subseteq S$ and $P \not\subseteq \overline{S}$ also hold.

Since $\mathcal{Q}$ is ND, there is a quorum $Q \in \mathcal{Q}$ such that either $Q \subseteq S$ or $Q \subseteq \overline{S}$ holds. Suppose first that $Q \subseteq S$. Since S is a transversal of $\mathcal{P}$ and $MinSet(\mathcal{T}) = Tr(\mathcal{P})$, $T \subseteq S$ for some $T \in \mathcal{T}$. Then, there is an $R \in TM(\mathcal{P}, \mathcal{Q}, \mathcal{T})$ such that $R \subseteq Q \cup T$, a contradiction since $R \subseteq S$. By the same argument, we can derive a contradiction when $Q \subseteq \overline{S}$. □

Observe that $TM(\mathcal{P}, \mathcal{Q}, Tr(\mathcal{P})) = \mathcal{P}$ if $\mathcal{P}$ is an ND coterie, since $Tr(\mathcal{P}) = \mathcal{P}$. If $\mathcal{P}$ is dominated, as the above theorem guarantees and as in Example 4, we can construct a new ND coterie $TM(\mathcal{P}, \mathcal{Q}, Tr(\mathcal{P}))$, given an ND coterie $\mathcal{Q}$.

Example 4. Consider a coterie $\mathcal{P} = \{\{1,2\},\{1,3,4\}\}$ under $\{1,2,3,4\}$. Then, $Tr(\mathcal{P}) = \{\{1\},\{2,3\}, \{2,4\}\}$. We first select a singleton coterie $\{\{3\}\}$ as $\mathcal{Q}$. Then,

$$TM(\mathcal{P}, \mathcal{Q}, Tr(\mathcal{P})) = \{\{1,2\},\{1,3\},\{2,3\}\},$$

which is an ND coterie. We next select another ND coterie $\{\{2,3\},\{2,4\},\{3,4\}\}$ as $\mathcal{Q}'$. Then,

$$TM(\mathcal{P}, \mathcal{Q}', Tr(\mathcal{P})) = \{\{1,2\},\{2,3\},\{2,4\},\{1,3,4\}\},$$

which is also ND.

There are, however, two difficult issues to overcome to apply the TM operation. First, the problem of determining $Tr(\mathcal{P})$ is likely to be co-NP-complete and, hence, we need to find a class of dominated coteries for which this problem becomes tractable. In the following sections, we discuss the class of grid coteries as it is a typical and practical class of coteries for which determination of $Tr(\mathcal{P})$ is tractable.

Second, the above example shows that the output ND coterie $TM(\mathcal{P}, \mathcal{Q}, Tr(\mathcal{P}))$ sharply depends on $\mathcal{Q}$. We will hence discuss, in Section 5, for a class of grid coteries what choice of $\mathcal{Q}$ creates an ND coterie that shows good performance when it is adopted in Maekawa's algorithm.

4 ND C-Grid Coteries

This section characterizes the minimal transversals of *C-Grid* coteries and then constructs ND C-Grid coteries. For *C*-Grid*, *M-Grid*, and *T-Grid* coteries, we can apply the same method to construct ND grid coteries. See the Appendix for the results.

We assume that, in this section, the processes in a distributed system are logically organized into an $m \times n$ grid, where m (≥ 2) and n (≥ 2) are the numbers of rows and columns, respectively. The rows and the columns, respectively, are labeled as $1, 2, \ldots, m$ from bottom to top and $1, 2, \ldots, n$ from left to right. The process at row i and column j is denoted by (i, j), where $1 \leq i \leq m$ and $1 \leq i \leq n$. Fig. 1 shows a 3×4 grid. Let $U_{m,n} = \{(i,j) | 1 \leq i \leq m$ and $1 \leq j \leq n\}$, which is the set of all processes on an $m \times n$ grid.

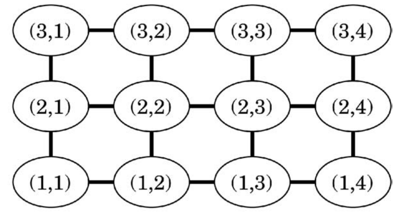

Fig. 1. A 3×4 grid.

Let S be a subset of $U_{m,n}$ and i be an integer such that $1 \leq i \leq m$. We frequently argue how completely the ith row belongs to S in the following. For compactness of description, we make use of the following predicates:

$$R_i\text{-}full(S) = \forall \ell (1 \leq \ell \leq n)[(i,\ell) \in S],$$
$$R_i\text{-}exist(S) = \exists \ell (1 \leq \ell \leq n)[(i,\ell) \in S],$$
$$R_i\text{-}partial(S) = \neg R_i\text{-}full(S), \text{ and}$$
$$R_i\text{-}null(S) = \neg R_i\text{-}exist(S).$$

Predicate $R_i\text{-}full(S)$ (respectively, $R_i\text{-}exist(S)$) is true if all (respectively, some) elements in row i belong to S.

If a set S includes all elements in some row i and another set T includes at least one element e_j from each row j, then S and T share element e_i. Proposition 4 claims this trivial fact.

Proposition 4. *Let S and T be subsets of $U_{m,n}$. Then, we have*

$$\exists i (1 \leq i \leq m) R_i\text{-}full(S) \wedge \forall i (1 \leq i \leq m) R_i\text{-}exist(T)$$
$$\Rightarrow S \cap T \neq \emptyset.$$

Definition 3 [4]. *Let S be a subset of $U_{m,n}$ and define the following predicate:*

$$C1(S) =$$
$$\exists i (1 \leq i \leq m) R_i\text{-}full(S) \wedge \forall i (1 \leq i \leq m) R_i\text{-}exist(S).$$

A C-Grid coterie $CG(m,n)$ *under $U_{m,n}$ is defined by*

$$CG(m,n) = MinSet(\{P \subseteq U_{m,n} | C1(P)\}).$$

A quorum $P \in CG(m,n)$ is said to be Type $C1$ *since $C1(P)$ holds.*

Now, we give a necessary and sufficient condition for a subset of $U_{m,n}$ to be a transversal of $CG(m,n)$. Define two predicates:

$$C2(S) = \exists i (1 \leq i \leq m) R_i\text{-}full(S)$$
$$C3(S) = \forall i (1 \leq i \leq m) R_i\text{-}exist(S).$$

Proposition 5. *S is a transversal of $CG(m,n)$ if and only if $C2(S) \vee C3(S)$ holds.*

Proof. If part: Suppose that an S satisfies $C2(S) \vee C3(S)$. Let $P \in CG(m,n)$ be any quorum. If S satisfies $C2(S)$, then $P \cap S \neq \emptyset$ by Proposition 4. If S satisfies $C3(S)$, then again $P \cap S \neq \emptyset$ by Proposition 4. Hence, S is a transversal of $CG(m,n)$.

Only if part: Assuming $\neg C2(S) \wedge \neg C3(S)$, we show that S is not a transversal. Since $\forall i (1 \leq i \leq m) R_i\text{-}partial(S)$ and $\exists i (1 \leq i \leq m) R_i\text{-}null(S)$, without loss of generality, we assume that $R_1\text{-}null(S)$ and, for $2 \leq i \leq m$, $(i, j_i) \notin S$.

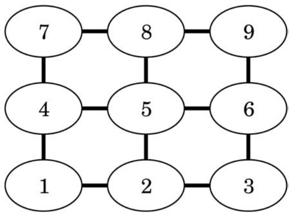

Fig. 2. A 3×3 grid.

Clearly, $P = \{(1, j) | 1 \le j \le n\} \cup \{(i, j_i) | 2 \le i \le m\}$ is in $CG(m, n)$ and $P \cap S = \emptyset$. □

By Proposition 5, a minimal transversal of $CG(m, n)$ is a subset S ($\subseteq U_{m,n}$) that consists of a process in each row or S that consists of all processes in some row. Then, we may define the following ND coterie: Let $\mathcal{Q}$ be an ND coterie under $U_{m,n}$. Then,

$$ND\text{-}CG(m, n, \mathcal{Q}) = TM(CG(m, n), \mathcal{Q}, CT(m, n))$$

is an ND coterie under $U_{m,n}$, where

$$CT(m, n) = MinSet(\{S \subseteq U_{m,n} | C2(S) \vee C3(S)\}).$$

We call $ND\text{-}CG(m, n, \mathcal{Q})$ an *ND C-Grid coterie*. A quorum P of $ND\text{-}CG(m, n, \mathcal{Q})$ is said to be *Type* Ci ($i = 2, 3$) if $Ci(P)$ and $Q \subseteq P$ for some $Q \in \mathcal{Q}$.

Example 5. Consider a C-Grid coterie $CG(3, 3)$ on a 3×3 grid shown in Fig. 2. We have

$$\begin{aligned} &CT(3, 3) \\ &= \{\{1,2,3\}, \{4,5,6\}, \{7,8,9\}, \{1,4,7\}, \{1,4,8\}, \{1,4,9\}, \\ &\quad \{1,5,7\}, \{1,5,8\}, \{1,5,9\}, \{1,6,7\}, \{1,6,8\}, \{1,6,9\}, \\ &\quad \{2,4,7\}, \{2,4,8\}, \{2,4,9\}, \{2,5,7\}, \{2,5,8\}, \{2,5,9\}, \\ &\quad \{2,6,7\}, \{2,6,8\}, \{2,6,9\}, \{3,4,7\}, \{3,4,8\}, \{3,4,9\}, \\ &\quad \{3,5,7\}, \{3,5,8\}, \{3,5,9\}, \{3,6,7\}, \{3,6,8\}, \{3,6,9\}\}. \end{aligned}$$

Clearly, $CG(3, 3) \ne CT(3, 3)$ and, hence, $CG(3, 3)$ is dominated, by Theorem 2. Let us select a singleton coterie $\{\{1\}\}$ as $\mathcal{Q}$. Then, we have an ND C-Grid coterie

$$\begin{aligned} &ND\text{-}CG(3, 3, \mathcal{Q}) \\ &= \{\{1,2,3\}, \{1,4,7\}, \{1,4,8\}, \{1,4,9\}, \{1,5,7\}, \{1,5,8\}, \\ &\quad \{1,5,9\}, \{1,6,7\}, \{1,6,8\}, \{1,6,9\}, \{1,4,5,6\}, \\ &\quad \{1,7,8,9\}, \{2,4,5,6,7\}, \{2,4,5,6,8\}, \{2,4,5,6,9\}, \\ &\quad \{2,4,7,8,9\}, \{2,5,7,8,9\}, \{2,6,7,8,9\}, \{3,4,5,6,7\}, \\ &\quad \{3,4,5,6,8\}, \{3,4,5,6,9\}, \{3,4,7,8,9\}, \{3,5,7,8,9\}, \\ &\quad \{3,6,7,8,9\}\}. \end{aligned}$$

In the above quorum set, $\{1,2,3\}$, $\{1,4,5,6\}$, and $\{1,7,8,9\}$ are Type $C2$, $\{1,4,7\}$, $\{1,4,8\}$, $\{1,4,9\}$, $\{1,5,7\}$, $\{1,5,8\}$, $\{1,5,9\}$, $\{1,6,7\}$, $\{1,6,8\}$, and $\{1,6,9\}$ are Type $C3$, and the other quorums are Type $C1$. Observe that $ND\text{-}CG(3, 3, \mathcal{Q})$ dominates $CG(3, 3, \mathcal{Q})$ and that $ND\text{-}CG(3, 3, \mathcal{Q})$ is indeed ND.

5 Performance of ND C-Grid Coteries

This section investigates the performance of ND C-Grid coteries in terms of the quorum size, the load, and the availability. We then introduce a class of ND C-Grid coteries, called *C-Majority coteries*, and show that they are optimal both in quorum size and in availability.

5.1 Quorum Size

Assume that the grid is square for simplicity. As earlier, N denotes the number of the processes in the grid, and $\mathcal{Q}$ is an ND coterie under the grid. Let K be the maximum quorum size of $\mathcal{Q}$, i.e., $K = \max_{Q \in \mathcal{Q}}\{|Q|\}$. The ND C-Grid Coterie constructed is $\mathcal{P} = ND\text{-}CG(\sqrt{N}, \sqrt{N}, \mathcal{Q})$.

The size of a Type $C1$ quorum of $\mathcal{P}$ is independent of $\mathcal{Q}$ and is $2\sqrt{N} - 1$. The size s of a Type $C2$ or $C3$ quorum P, however, depends on $\mathcal{Q}$ and satisfies $\sqrt{N} \le s \le \sqrt{N} + K$ since the size of a minimal traversal of $CG(\sqrt{N}, \sqrt{N})$ is $\sqrt{N}$. Hence, the quorum size is $O(\sqrt{N} + K)$.

We therefore can keep the maximum quorum size of ND C-Grid coteries as small as C-Grid coteries, i.e., $O(\sqrt{N})$, by choosing a coterie $\mathcal{Q}$ such that $K \le \sqrt{N}$.

5.2 Load

The load of a coterie is introduced for evaluating load sharing ability in [16]. A strategy is a list of probabilities that represents the frequencies of quorums being selected by the mutual exclusion algorithm. The strategy induces a load on each process u which is obtained by summing the probabilities of all quorums that include u. For example, consider a coterie $\mathcal{P}$ in Example 1 and assume that the algorithm selects two quorums in $\mathcal{P}$ with the same probability 0.5. Then, the load of a process 1 is 1 and those of both processes 2 and 3 are 0.5. The load $L(\mathcal{P})$ of a coterie $\mathcal{P}$ is defined as the minimum load of the busiest process among all strategies. Although $L(\mathcal{P})$ represents the potential of $\mathcal{P}$ with respect to load sharing, the practical load depends also on the algorithm that uses $\mathcal{P}$. Hence, the practical load can be bad even if $\mathcal{P}$ has good load.

It is known that, if a coterie $\mathcal{P}$ dominates a coterie $\mathcal{Q}$, then $L(\mathcal{P}) \le L(\mathcal{Q})$ [16]. The load of ND C-Grid coteries is thus lower than or equal to those of C-Grid coteries.

5.3 Availability

Given the probabilities that processes are operational, the availability $A(\mathcal{P})$ of a coterie $\mathcal{P}$ is the probability that all processes in some quorum are operational. By the definitions of dominatedness and availability, the availability of an ND C-Grid coterie is superior to that of the corresponding C-Grid coterie. In this section, we investigate an asymptotic behavior of the availability.

Assuming that all processes are operational with the same probability p, this section considers ND C-Grid coteries on an $m \times n$ grid.[4] Let $q = 1 - p$, which is the probability that a process is faulty. Due to [4], the availability of $CG(m, n)$ is shown to be:

$$A(CG(m, n)) = (1 - q^n)^m - (1 - p^n - q^n)^m.$$

Notice that $A(CG(m, n))$ converges to 0 as m and n approach ∞ and it is equivalent to the probability that a Type $C1$ quorum of an ND C-Grid coterie on the grid is available.

4. For a generality, we investigate this case, although our main concern in this section is the case of $m = n = \sqrt{N}$.

As in Section 4, let

$$C2(S) = \exists i(1 \le i \le m)R_i\text{-}full(S) \text{ and}$$
$$C3(S) = \forall i(1 \le i \le m)R_i\text{-}exist(S),$$

and let $\mathcal{Q}$ an ND coterie under $U_{m,n}$. Then, $T \subseteq U$ is a transversal of $CG(m,n)$ if and only if $C2(T)$ or $C3(T)$ holds by Proposition 5 and $Q \cup T$ is a superset of a quorum of $ND\text{-}CG(m,n,\mathcal{Q})$ for any transversal T and quorum $Q \in \mathcal{Q}$ by definition. Let α be the probability that all processes in one of such sets $Q \cup T$ are operational. Clearly, $\alpha \le A(ND\text{-}CG(m,n,\mathcal{Q}))$. We evaluate α. Since

$$Pr(\forall i(1 \le i \le m)R_i\text{-}exist(S)) = (1-q^n)^m$$

and

$$Pr(\exists i(1 \le i \le m)R_i\text{-}full(S)) = 1-(1-p^n)^m,$$

the probability β that all processes in some traversal T are operational is

$$\begin{aligned}\beta &= (1-q^n)^m + (1-(1-p^n)^m) - A(CG(m,n)) \\ &= 1-(1-p^n)^m + (1-p^n-q^n)^m.\end{aligned}$$

Since $\alpha \ge \beta \times A(\mathcal{Q})$ and β converges to 1, when m and n approach ∞, α converges to $A(\mathcal{Q})$. Consequently, we have the following theorem.

Theorem 5. *Let $\mathcal{Q}$ be an ND coterie under $U_{m,n}$. The availability of $ND\text{-}CG(m,n,\mathcal{Q})$ converges to that of $\mathcal{Q}$ when m and n approach ∞.*

Let $\mathcal{P}$ be a coterie under an ℓ-set U. Provided that $1/2 < p < 1$, a desirable asymptotic behavior of $A(\mathcal{P})$, when ℓ approaches ∞, is that $A(\mathcal{P})$ converges to 1. Such an $A(\mathcal{P})$ is said to be *Condorcet* [18]. By Theorem 5, we have the following proposition.

Proposition 6. *If $A(\mathcal{Q})$ is Condorcet, so is $A(ND\text{-}CG(m,n,\mathcal{Q}))$.*

5.4 C-Majority Coteries

Based on the characterizations shown in the above, this section presents an ND C-Grid coterie that achieves both small quorum size and good availability. We again consider the case where $m = n = \sqrt{N}$ for simplicity. The analysis of quorum size given above shows that $\mathcal{Q}$ must be chosen so that its maximum quorum size K is at most $\sqrt{N}$ to utilize the advantage in quorum size. Perhaps the easiest way is to choose one that is defined under the set of all processes in a single row. Let MAJ_i be the majority coterie under the set of processes in some row i.[5] For example, $MAJ_1 = \{\{1,2\}, \{1,3\}, \{2,3\}\}$ is the majority coterie under the set of all processes in the bottom row of the 3×3 grid of Fig. 2. Then, we define an ND C-Grid coterie $ND\text{-}CG(\sqrt{N}, \sqrt{N}, MAJ_i)$, which we call the *C-Majority coterie*. As majority coteries have Condorcet availabilities [18], so have C-Majority coteries by Proposition 6. Furthermore, the minimum quorum size is $\sqrt{N}$ and the maximum quorum size is $2\sqrt{N}-1$, which are the same order as C-Grid coteries.

5. A majority coterie under U is defined as follows: Let $N = |U|$. When N is odd, every $(N+1)/2$-set of processes forms a quorum. When N is even, pick up a process u and then the majority coterie is defined to be the one under the set of processes $U \setminus \{u\}$.

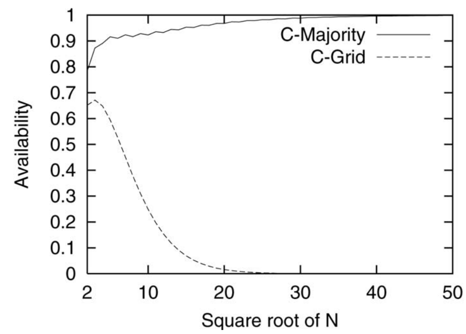

Fig. 3. Availabilities of C-Majority coterie and C-Grid coterie when $p = 0.7$.

To demonstrate the difference between C-Majority and C-Grid coteries, in Fig. 3 we show the availabilities of $ND\text{-}CG(\sqrt{N}, \sqrt{N}, MAJ_i)$ and $CG(\sqrt{N}, \sqrt{N})$ when $p = 0.7$ and $2 \le \sqrt{N} \le 50$.

6 Quorum Construction Algorithms for ND C-Grid Coteries

In the introduction, we mentioned, as an advantage of grid coteries, the fact that a quorum is constructible on the fly when it is necessary for a process. In this section, we present an online algorithm to construct a quorum for ND C-Grid coteries to demonstrate that ND C-Grid coteries preserve this merit of grid coteries.

Let $U_{m,n}$ and $\mathcal{Q}$, respectively, be the set of processes under consideration and an ND coterie under $U_{m,n}$. Then, an algorithm, GetQuorum, given in Fig. 4 decides an order to collect token $\texttt{permission}_v$ from all processes v in some quorum of $ND\text{-}CG(m,n,\mathcal{Q})$. The quorums of $\mathcal{Q}$ are stored in an array Q with h elements $Q[1], Q[2], \ldots, Q[h]$, where h is the number of quorums. Variable S records the processes from which permissions have been collected. GetQuorum returns "success" if the permissions have been collected from all members of a quorum. It returns "fail" if it has decided to abandon this quorum. In this case, the permissions that have been collected are returned, to avoid deadlock.

GetQuorum makes use of three functions $R\text{-}Cover(k,\ell)$, $R\text{-}Line(k,\ell)$, and $Q\text{-}Union(Q)$. $R\text{-}Cover(k,\ell)$ (respectively, $R\text{-}Line(k,\ell)$, $Q\text{-}Union(Q)$) tries to collect permissions from processes in such a way that $\forall i(k \le i \le \ell)R_i\text{-}exist(S)$ (respectively, $\exists i(k \le i \le \ell)R_i\text{-}full(S)$, $\exists w(1 \le w \le |Q|)[Q[w] \subseteq S]$) holds. These functions return "true" if the permissions have been successfully collected and, otherwise, return "false."[6] By definition, the permission is collected from every member of a Type $C1$ (respectively, $C2$, $C3$) quorum when both of $R\text{-}Cover(1,m)$ and $R\text{-}Line(1,m)$ (respectively, both of $R\text{-}Line(1,m)$ and $Q\text{-}Union(Q)$, both of $R\text{-}Cover(1,m)$ and $Q\text{-}Union(Q)$) return true. That is, GetQuorum forms a quorum if at least two out of $R\text{-}Cover(1,m)$, $R\text{-}Line(1,m)$, and $Q\text{-}Union(Q)$ return true.

Although GetQuorum dynamically constructs a quorum for a general $ND\text{-}CG(m,n,\mathcal{Q})$, it still needs space to store $\mathcal{Q}$ in Q. By adopting a coterie $\mathcal{Q}$ whose quorums are dynamically constructible, we can even cut the space for

6. Although we omit the algorithm descriptions for these three functions because of their simplicity, the reader may obviously construct them.

```
Algorithm GetQuorum(m, n, Q):
Let S ← ∅.                 /* S records processes whose permissions have been collected */
if R-Cover(1, m) then
    if R-Line(1, m) then
        return "success"                              /* Type C1 quorum is formed */
    else
        if Q-Union(Q) then
            return "success"                          /* Type C3 quorum is formed */
        else return permissions to all processes in S, and return "fail"
else
    if R-Line(1, m) then
        if Q-Union(Q) then
            return "success"                          /* Type C2 quorum is formed */
        else return permissions to all processes in S, and return "fail"
    else return permissions to all processes in S, and return "fail."
```

Fig. 4. Algorithm GetQuorum.

```
Algorithm GetQuorum2(m, n):
Let S ← ∅.                 /* S records processes whose permissions have been collected */
if R-Line(m, m) then
    return "success"                                  /* Type C2 quorum is formed */
else
    if (n is odd and |S| ≥ (n + 1)/2) or (n is even and |S \ {(m, n)}| ≥ n/2) then
        if R-Cover(1, m − 1) then
            return "success"                          /* Type C3 quorum is formed */
        else
            if R-Line(1, m − 1) then
                return "success"                      /* Type C2 quorum is formed */
            else return permissions to all processes in S, and return "fail"
    else
        if |S| ≠ 0 then
            if R-Cover(1, m − 1) and R-Line(1, m − 1) then
                return "success"                      /* Type C1 quorum is formed */
            else return permissions to all processes in S, and return "fail"
        else return permissions to all processes in S, and return "fail"
```

Fig. 5. Algorithm GetQuorum2.

$\mathcal{Q}$. We give in Fig. 5 a quorum construction algorithm GetQuorum2 for a C-Majority coterie.[7]

We would like to make two remarks on GetQuorum2. First, GetQuorum2 calls the function $R\text{-}Line(m, m)$ and succeeds forming a Type C2 quorum in the absence of faulty processes. That is, it requests the permissions only to the processes in row m. In this case, GetQuorum2 achieves the optimal quorum size n, but the load is worst. However, we can reduce the load by modifying GetQuorum2 so that it tries to construct a Type C1 quorum first at the expense of the quorum size.

Second, the TM operation produces a coterie $MinSet(\mathcal{R})$, where

$$\mathcal{R} = \mathcal{P} \cup \{Q \cup T | Q \in \mathcal{Q} \text{ and } T \in \mathcal{T}\}.$$

What GetQuorum2 constructs is a member of $\mathcal{R}$, which may not be a quorum of $MinSet(\mathcal{R})$. Since any member of $\mathcal{R}$ contains a quorum of $MinSet(\mathcal{R})$, $\mathcal{R}$ satisfies the Intersection Property and, hence, $\mathcal{R}$ is sufficient for Maekawa's algorithm to guarantee mutual exclusion. This modification does not affect the availability. A little worry is the quorum size. We, however, can show that it is $O(\sqrt{N})$.

7. We assume that the majority coterie adopted is under the set of processes in the mth row.

7 ND Crumbling Wall Coteries

As a generalization of rectangular grid structure, Peleg and Wool [19] introduced a new grid structure, called *Crumbling Wall* (*CW*, for short), and defined a *Crumbling Wall coterie* (*CW coterie*, for short) on it.

A CW is defined as follows: Let m be an integer denoting the number of rows, where $m \geq 2$. We number the rows as $1, 2, \ldots, m$ from bottom to top. Let $\vec{n} = (n_1, n_2, \ldots, n_m)$ be a sequence of m positive integers, where n_i denotes the number of processes in row i. For each row i, the processes in row i are denoted as $(i, 1), (i, 2), \ldots, (i, n_i)$ from left to right. Fig. 6 shows a CW with $m = 4$ and $\vec{n} = (3, 2, 4, 2)$. We denote by $U_{m,\vec{n}}$ the set of processes in the CW, i.e., $\{(i, j) | 1 \leq i \leq m \text{ and } 1 \leq j \leq n_i\}$. Let S and i, respectively, be a subset

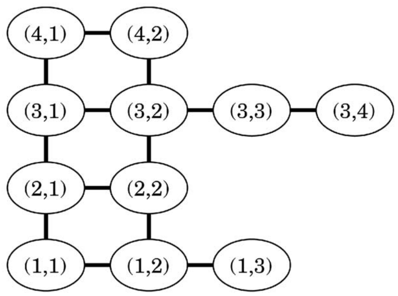

Fig. 6. A crumbling wall with $m = 4$ and $\vec{n} = (3, 2, 4, 2)$.

of $U_{m,\vec{n}}$ and and an integer such that $1 \le i \le m$. Recall the following two predicates.

$$R_i\text{-}full(S) = \forall \ell(1 \le \ell \le n_i)[(i, \ell) \in S]$$
$$R_i\text{-}exist(S) = \exists \ell(1 \le \ell \le n_i)[(i, \ell) \in S].$$

Definition 4 [19]. *Let S be a subset of $U_{m,\vec{n}}$. A* CW *coterie $CW(m, \vec{n})$ under $U_{m,\vec{n}}$ is defined by*

$$CW(m, \vec{n}) = MinSet(\{P \subseteq U_{m,\vec{n}} | W1(P)\}),$$

where

$$W1(S) = \exists i(1 \le i \le m)[R_i\text{-}full(S) \wedge \forall k(1 \le k < i)R_k\text{-}exist(S)].$$

$CW(m, \vec{n})$ is ND if and only if $n_m = 1$ and $n_i \ge 2$ for all $1 \le i < m$ [19]. For all CWs such that $n_i \ge 2$ for all $1 \le i \le m$, $CW(m, \vec{n})$ are hence dominated. We construct ND coteries from those dominated CWs. Define

$$W2(S) = \forall i(1 \le i \le m)R_i\text{-}exist(S).$$

By using an argument similar to that in the proof of Proposition 5, we can show that a minimal transversal of $CW(m, \vec{n})$ is either a quorum of $CW(m, \vec{n})$ or a set $T \subset U_{m,\vec{n}}$ consisting of a single process from each row, as formally stated in the following.

Proposition 7. *T is a transversal of $CW(m, \vec{n})$ if and only if $W1(T) \vee W2(T)$ holds.*

Let $\mathcal{Q}$ be an ND coterie under $U_{m,\vec{n}}$. We define an ND CW coterie under $U_{m,\vec{n}}$ as follows:

$$ND\text{-}CW(m, \vec{n}, \mathcal{Q}) = TM(CW(m, \vec{n}), \mathcal{Q}, CWT(m, \vec{n})),$$

where

$$CWT(m, \vec{n}) = MinSet(\{S \subseteq U_{m,\vec{n}} | W1(S) \vee W2(S)\}).$$

Peleg and Wool gave a method to construct ND coteries on CWs [19]. Our method can be regarded as its generalization; if we restrict ourselves to adopting as $\mathcal{Q}$ a singleton coterie $\{\{u\}\}$ for a process u in the top row, then our method is the same as theirs. Consider a CW in Fig. 6 and another CW with $m = 4$ and $\vec{n'} = (3, 2, 4, 1)$ for instance. Let $\mathcal{Q}$ be a singleton coterie $\{\{(4, 1)\}\}$. Then, $ND\text{-}CW(m, \vec{n}, \mathcal{Q}) = CW(m, \vec{n'})$.

8 Conclusions and Discussions

Grid coteries have many desirable properties, but they are unfortunately not ND, except some very special cases. This paper has proposed a method to construct, from grid coteries, ND coteries that preserve the advantages of grid coteries. To this end, we have introduced the TM operation that produces an ND coterie, given a dominated coterie, its minimal transversals, and another ND coterie. Then, for each of several well-known grid coteries, we have characterized its transversals and then constructed ND grid coteries by using the TM operation. Finally, we have demonstrated that our method improves the disadvantages of the original grid coteries with preserving their advantages by 1) evaluating the quorum size, the load, and the availability and 2) presenting a simple dynamic quorum construction algorithm. During the analyses, we have introduced C-Majority coteries and showed that they have almost all good properties. Although those analyses have been done for simple grid coteries, the same method is applicable for more general framework of crumbling wall coteries, as has been demonstrated in the last section.

Before concluding the paper, we discuss the possibility of applying our method to another quorum system. Let U be the set of processes. A *write-read coterie* (*wr-coterie*, for short) under U is an ordered pair $(\mathcal{Q}, \mathcal{P})$, where $\mathcal{Q}$ and $\mathcal{P}$ are sets of nonempty subsets of U satisfying the following properties [5]:

1. Both $\mathcal{Q}$ and $\mathcal{P}$ satisfy Minimality,
2. $\mathcal{Q}$ satisfies Intersection Property, and
3. $\forall Q \in \mathcal{Q} \forall P \in \mathcal{P}[Q \cap P \neq \emptyset]$.

The notion of a wr-coterie is introduced to model *write* and *read quorums* for maintaining consistent access to replicated data, where an element of $\mathcal{Q}$ models a write quorum and that of $\mathcal{P}$ a read quorum. By definition, every write quorum intersects with any other write and read quorum, while there is no intersection constraint between two read quorums. Hence, any read process can access the latest data, assuming that a process updates replicated data at every process in a write quorum when to update them.

Let $(\mathcal{Q}, \mathcal{P})$ and $(\mathcal{Q}', \mathcal{P}')$ be wr-coteries under U. Then, $(\mathcal{Q}, \mathcal{P})$ is said to *dominate* $(\mathcal{Q}', \mathcal{P}')$, if the following conditions hold:

1. $\mathcal{Q} \neq \mathcal{Q}'$ and $\mathcal{P} \neq \mathcal{P}'$,
2. $\forall Q' \in \mathcal{Q}' \exists Q \in \mathcal{Q}[Q \subseteq Q']$, and
3. $\forall P' \in \mathcal{P}' \exists P \in \mathcal{P}[P \subseteq P']$.

If there is no wr-coterie that dominates $(\mathcal{Q}, \mathcal{P})$, then $(\mathcal{Q}, \mathcal{P})$ is said to be *nondominated* (*ND*, for short). ND wr-coteries achieve good performance by their maximality. Since a wr-coterie $(\mathcal{Q}, \mathcal{P})$ is ND if and only if $\mathcal{P} = Tr(\mathcal{Q})$ [9], we immediately obtain the following ND wr-coteries:

a. $(CG(m, n), CT(m, n))$, known as a *modified grid wr-coterie* [11].
b. $(C^*G(m, n), C^*T(m, n))$.
c. $(MG(m, n), MT(m, n))$.
d. $(TG(m, n), TT(m, n))$.
e. $(CW(m, \vec{n}), CWT(m, \vec{n}))$.

Appendix A

Other ND Grid Coteries

Section 4 presents how to construct ND coteries from C-Grid coteries. This appendix prepares the cases of the other grid coteries, C*-Grid, M-Grid, and T-Grid coteries, the definitions of which are given in the Introduction. In the following, we present the results without proofs, but readers can check them easily by the same arguments in Section 4.

Let j be an integer such that $1 \leq j \leq n$. We use the following predicates, where C stands for a column.

$$C_j\text{-}full(S) = \forall k(1 \leq k \leq m)[(k, j) \in S)],$$
$$C_j\text{-}exist(S) = \exists k(1 \leq k \leq m)[(k, j) \in S)].$$

A.1 ND C*-Grid Coteries

An *C*-Grid coterie* $C^*G(m, n)$ under $U_{m,n}$ is defined by

$$C^*G(m, n) = MinSet(\{S|C1(S) \vee C^*2(S)\}),$$

where

$$C^*2(S) = \exists\, j(1 \leq j \leq n)C_j\text{-}full(S)) \wedge \forall j(1 \leq j \leq n)\ C_j\text{-}exist(S).$$

The set $C^*T(m, n)$ of all the minimal transversals is given by

$$C^*T(m, n) = MinSet(\{S|C^*3(S) \vee C^*4(S) \vee C^*5(S)\}),$$

where $C^*3(S) = \exists i(1 \leq i \leq m)R_i\text{-}full(S)$, $C^*4(S) = \exists j(1 \leq j \leq n)\ C_j\text{-}full(S)$, and

$$C^*5(S) = \forall i(1 \leq i \leq m)R_i\text{-}exist(S) \wedge \forall j(1 \leq j \leq n)C_j\text{-}exist(S).$$

We then define *ND C*-Grid coterie* $ND\text{-}C^*G(m, n, \mathcal{Q})$ as follows:

$$ND\text{-}C^*G(m, n, \mathcal{Q}) = TM(C^*G(m, n), \mathcal{Q}, C^*T(m, n)).$$

A.2 ND M-Grid Coteries

An *M-Grid coterie* $MG(m, n)$ under $U_{m,n}$ is defined by

$$MG(m, n) = MinSet(\{S|M1(S)\}),$$

where

$$M1(S) = \exists i(1 \leq i \leq m)R_i\text{-}full(S) \wedge \exists j(1 \leq j \leq n)\ C_j\text{-}full(S).$$

The set $MT(m, n)$ of all the minimal transversals is given by

$$MT(m, n) = MinSet(\{S|M2(S) \vee M3(S)\}),$$

where $M2(S) = \forall i(1 \leq i \leq m)R_i\text{-}exist(S)$ and $M3(S) = \forall j (1 \leq j \leq n)C_j\text{-}exist(S)$.

We then define an *ND M-Grid coterie* $ND\text{-}MG(m, n, \mathcal{Q})$ as follows:

$$ND\text{-}MG(m, n, \mathcal{Q}) = TM(MG(m, n), \mathcal{Q}, MT(m, n)).$$

A.3 ND T-Grid Coteries

Let S be a subset of $U_{m,n}$ and define the following predicates:

$$T1(S) = \exists i(1 \leq i \leq m)[R_i\text{-}full(S) \wedge \forall k(1 \leq k < i)R_k\text{-}exist(S)]$$
$$T2(S) = \forall i(1 \leq i \leq m)R_i\text{-}exist(S).$$

A *T-Grid coterie* $TG(m, n)$ under $U_{m,n}$ is defined by

$$TG(m, n) = MinSet(\{P \subseteq U_{m,n}|T1(P)\}).$$

The set $TT(m, n)$ of all the minimal transversals is given by

$$TT(m, n) = MinSet(\{S|T1(S) \vee T2(S)\}).$$

We then define an *ND T-Grid coterie* $ND\text{-}TG(m, n, \mathcal{Q})$ as follows:

$$ND\text{-}TG(m, n, \mathcal{Q}) = TM(TG(m, n), \mathcal{Q}, TT(m, n)).$$

ACKNOWLEDGMENTS

The authors wish to thank the anonymous referees and the associate editor for their helpful comments and suggestions. A preliminary version of this paper appeared in the *Proceedings of the 15th IASTED International Conference on Parallel and Distributed Computing and Systems*, November 2003. The authors gratefully acknowledge the partial support of the Scientific Grant in Aid by the Ministry of Education, Science, Sports, Culture, and Technology of Japan.

REFERENCES

[1] D. Agrawal and A. El Abbadi, “An Efficient and Fault-Tolerant Solution for Distributed Mutual Exclusion,” *ACM Trans. Computer Systems,* vol. 9, no. 1, pp. 1-20, Feb. 1991.
[2] D. Barbara and H. Garcia-Molina, “The Vulnerability of Vote Assignments,” *ACM Trans. Computer Systems,* vol. 4, no. 3, pp. 187-213, Aug. 1986.
[3] J.C. Bioch and T. Ibaraki, “Generating and Approximating Nondominated Coteries,” *IEEE Trans. Parallel and Distributed Systems,* vol. 6, no. 9, pp. 905-914, Sept. 1995.
[4] S.Y. Cheung, M.H. Ammar, and M. Ahamad, “The Grid Protocol: A High Performance Scheme for Maintaining Replicated Data,” *IEEE Trans. Knowledge and Data Eng.,* vol. 4, no. 6, pp. 582-592, Dec. 1992.
[5] A.W. Fu, “Delay-Optimal Quorum Consensus for Distributed Systems,” *IEEE Trans. Parallel and Distributed Systems,* vol. 8, no. 1, pp. 59-69, Jan. 1997.
[6] D.K. Gifford, “Weighted Voting for Replicated Data,” *Proc. Seventh Symp. Operating Systems Principles,* pp. 150-159, Dec. 1979.
[7] H. Garcia-Molina and D. Barbara, “How to Assign Votes in a Distributed Systems,” *J. ACM,* vol. 32, no. 4, pp. 841-860, Oct. 1985.
[8] T. Harada and M. Yamashita, “Improving the Availability of Mutual Exclusion Systems on Incomplete Networks,” *IEEE Trans. Computers,* vol. 48, no. 7, pp. 744-747, July 1999.
[9] T. Ibaraki and T. Kameda, “A Theory of Coteries: Mutual Exclusion in Distributed Systems,” *IEEE Trans. Parallel and Distributed Systems,* vol. 4, no. 7, pp. 779-794, July 1993.
[10] A. Kumar and S.Y. Cheung, “A High Availability $\sqrt{N}$ Hierarchical Grid Algorithm for Replicated Data,” *Information Processing Letters,* vol. 40, no. 6, pp. 311-316, Dec. 1991.
[11] A. Kumar, M. Rabinovich, and R.K. Sinha, “A Performance Study of General Grid Structures for Replicated Data,” *Proc. 13th Int'l Conf. Distributed Computing Systems,* pp. 178-185, May 1993.
[12] A. Kumar, “An Efficient Supergrid Protocol for High Availability and Load Balancing,” *IEEE Trans. Computers,* vol. 49, no. 10, pp. 1126-1133, Oct. 2000.
[13] M. Maekawa, “A $\sqrt{N}$ Algorithm for Mutual Exclusion in Decentralized Systems,” *ACM Trans. Computer Systems,* vol. 3, no. 2, pp. 145-159, May 1985.
[14] M.L. Neilsen, “Quorum Structures in Distributed Systems,” PhD thesis, Kansas State Univ., 1992.
[15] M.L. Neilsen and M. Mizuno, “Coterie Join Algorithm,” *IEEE Trans. Parallel and Distributed Systems,* vol. 3, no. 5, pp. 582-590, Sept. 1992.

[16] M. Naor and A. Wool, "The Load, Capacity, and Availability of Quorum Systems," *SIAM J. Computing,* vol. 27, no. 2, pp. 423-447, Mar. 1998.
[17] N. Preguiça and J. Martins, "Revisiting Hierarchical Quorum Systems," *Proc. 21st Int'l Conf. Distributed Computing Systems,* pp. 264-274, Apr. 2001.
[18] D. Peleg and A. Wool, "The Availability of Quorum Systems," *Information and Computation,* vol. 123, no. 2, pp. 210-223, Dec. 1995.
[19] D. Peleg and A. Wool, "Crumbling Walls: A Class of Practical and Efficient Quorum Systems," *Distributed Computing,* vol. 10, no. 2, pp. 87-97, 1997.
[20] B. Sanders, "The Information Structure of Distributed Mutual Exclusion Algorithm," *ACM Trans. Computer Systems,* vol. 5, no. 3, pp. 284-299, Aug. 1987.
[21] R. Thomas, "A Majority Consensus Approach to Concurrency Control," *ACM Trans. Database Systems,* vol. 7, no. 3, pp. 323-342, June 1979.

Takashi Harada received the BEng and MEng degrees from Hiroshima University, Japan, in 1983 and 1985, respectively, and the DrEng degree from Kyushu University, Japan, in 2000. From 1986 to 2000, he was a research associate in the Information Processing Center at Hiroshima University. He is currently an associate professor in the Graduate School of Management at Hiroshima University. His research interests include distributed computing, fault-tolerant computing, and communication networks. He is a member of the ACM, the IEEE Computer Society, the IEICE of Japan, and the IPS of Japan.

Masafumi Yamashita received the BE and ME degrees from Kyoto University, Kyoto, Japan, in 1974 and 1977, respectively. He received his Doctor of Engineering degree from Nagoya University, Nagoya, Japan, in 1981. From 1980 to 1985, he was with the Department of Information and Computer Sciences, Toyohashi University of Technology, Aichi, Japan. In 1985, he joined the Faculty of Engineering at Hiroshima University as an associate professor, and was a professor from 1992 to 1998. Since 1998, he has been a professor in the Graduate School of Information Science and Electrical Engineering, Kyushu University, Fukuoka, Japan. He has held visiting appointments several times with Simon Fraser University, B.C., Canada and the University of Wisconsin-Milwaukee. His research interest include distributed algorithms/systems, parallel algorithms, and cooperative systems. He is a member of the Institute of Electronics, Information and Communication Engineers of Japan, the Information Processing Society of Japan, SIAM Japan, the IEEE Computer Society, and the ACM.