How to Assign Votes in a Distributed System

HECTOR GARCIA-MOLINA AND DANIEL BARBARA

Princeton University, Princeton, New Jersey

Abstract. In a distributed system, one strategy for achieving mutual exclusion of groups of nodes without communication is to assign to each node a number of votes. Only a group with a majority of votes can execute the critical operations, and mutual exclusion is achieved because at any given time there is at most one such group. A second strategy, which *appears* to be similar to votes, is to define a priori a set of groups that intersect each other. Any group of nodes that finds itself in this set can perform the restricted operations. In this paper, both of these strategies are studied in detail and it is shown that they are *not* equivalent in general (although they are in some cases). In doing so, a number of other interesting properties are proved. These properties will be of use to a system designer who is selecting a vote assignment or a set of groups for a specific application.

Categories and Subject Descriptors: C.2.4 [**Computer-Communications Networks**]: Distributed Systems—*distributed databases*; H.2.4 [**Database Management**]: Systems—*distributed systems*

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Distributed databases, mutual exclusion, partitions

1. *Introduction*

In many distributed systems it is necessary to have a *mutual exclusion* mechanism that works even when nodes fail or the communication lines are broken. For example, consider a system that manages replicated data. Owing to a network partition, the system may be divided into isolated groups of nodes. We probably do not want users at isolated groups updating the database concurrently since this would cause the copies to diverge [12]. So, if a group is going to perform updates, it must be able to guarantee that no other group is performing this activity. This mutual exclusion has to be enforced without communication between groups.

One well-known solution is to assign a priori a number of *votes* (or points) to each node in the system, and a group whose members have a majority of the total votes is allowed to perform the restricted operation (e.g., [3, 5, 6, 15]). The mutual exclusion is achieved because at most a single group can have a majority of votes at a time. (It is possible that at a given time no group has a majority and can perform the operation. There seems to be no way to avoid this problem. Even giving one node all votes does not help since that node may fail.)

Votes are used to achieve mutual exclusion in a number of other algorithms. For example, in the so-called Byzantine Generals problem, nodes may fail and yield incorrect or even misleading results. The computation being performed must

Authors' present addresses: H. Garcia-Molina, Department of Electrical Engineering and Computer Science, Princeton University, Princeton, NJ 08544; D. Barbara, Universidad Simon Bolivar, Departamento de Matematicas y Ciencias de Computación, Sartenejas, Baruta, Venezuela.

be replicated, and if nodes with a majority of votes agree on a result, it is considered correct (e.g., [4], [8] and [11]). Votes are also used in some commit protocols (e.g., [13] and [14]). After a failure, nodes must decide whether to commit or abort a transaction, and the protocol must ensure that at most one group of nodes makes such a decision.

A second solution to the mutual exclusion problem was suggested by Lamport in 1978 [7], but because it *appears* to be so similar to vote assignment, it has received little attention. The idea is to define a priori a set of groups that may perform the restricted operation. Each pair of groups should have a node in common to guarantee mutual exclusion. For example, if we have nodes a, b, and c we may define the set $\{\{a, b\}, \{b, c\}, \{a, c\}\}$. Nodes a and b can perform the operation together, knowing that neither group $\{b, c\}$ or $\{a, c\}$ can be formed. Notice that this set of groups is equivalent to assigning one vote to each node (or n votes to each node).

The assignment of votes or the choice of set of groups can have a critical effect on the reliability of a distributed system. Consider, for example, a system with nodes a, b, c, and d and an assignment that gives one vote to each. This seems like a natural choice because it gives each node equal weight. Since three votes are needed for a majority, this is equivalent to the set of groups

$$S = \{\{a, b, c\}, \{a, b, d\}, \{a, c, d\}, \{b, c, d\}\}.$$

But now, consider an assignment that gives node a two votes and the rest a single vote. The majority is still three votes, so this is equivalent to

$$R = \{\{a, b\}, \{a, c\}, \{a, d\}, \{b, c, d\}\}.$$

Set R and its associated vote assignment is clearly superior to S because all groups of nodes that can operate under S can operate under R, but not vice versa. For instance, a and b can form a group under R but not under S. So, if the system splits into groups $\{a, b\}$ and $\{c, d\}$, there will be one active group under R but none under S. So clearly, no system designer should *ever* select set S (or its equivalent vote assignment), in spite of the fact it seems "natural."

We use the term "*R dominates S*" to mean that R is always superior to S. Obviously, we want to ignore dominated sets (or vote assignments). But even if we do, we must still select one of the nondominated sets, and this is no easy task. In our example, which of the nodes should get the two votes? Or should we give one node one vote and the rest three votes? Or four, three, two, and two votes?

There are many choices, but many are duplicates. For example, giving a four votes, b three votes, and c and d two each, yields exactly the same set of groups that was given by R. So again, in the selection process, we want to ignore duplicate vote assignments.

Once the number of choices is narrowed down, the system designer will have to consider each one in light of the failure characteristics of the system. The set or assignment that maximizes the probability that the system is in operation would be selected and used in practice.

The objective of this paper is to study vote assignments and sets of groups, in order to narrow down the number of choices that must be considered by the system designer. We start by formalizing notions like dominated sets, dominated assignments, and identical assignments. In doing so, we prove a number of interesting results, including the fact that vote assignments and sets of groups are *not* equivalent. That is, there are sets for which there exists no vote assignment.

In the second part of the paper, we show that any algorithm to enumerate all the nondominated (ND) choices is too expensive, and we develop a partial enumeration technique that covers all our needs and has interesting properties. In general, the number of choices is huge, but surprisingly, for systems with 5 or fewer nodes, the number is relatively small. For example, for 4 nodes, there are *only 3* basic sets of groups that "make sense." (Considering permutations of the nodes, this becomes a total of 12 cases. For 5 nodes, there are 7 basic sets, or 131 cases including permutations.) These results are very encouraging. They imply that, in "small systems," it is possible to study all choices and select the true optimum with respect to reliability.

For systems with 6 or more nodes, the number of choices explodes, and it will be practically impossible (even for 6 nodes) to select the true optimum. However, we expect that in reality a vast majority of the systems will be under 6 nodes, so this result is not too disheartening. Most systems will be small because the "system" only includes the nodes participating in the mutual exclusion algorithm, not the entire distributed system. Furthermore, many of the algorithms involve replication of resources, which because of its cost, tends to be limited. (For instance, even in the Space Shuttle, 5 replicated computers were judged to give sufficient reliability).

2. *Basic Concepts*

We start by defining sets of groups and domination. Notice that, to avoid confusion, we are referring to sets of nodes as *groups*. Sets of groups are thus sets of sets of nodes. In dealing with mutual exclusion, we do not want to have sets like $\{\{a\}, \{b, c\}\}$ or $\{\{a\}, \{a, b\}\}$. We use the term *coterie* (from the French) to refer to the sets of groups that are "well formed." (According to Webster's dictionary, a coterie is a "close circle of friends who share a common interest... A set refers to a group, usually larger and, hence, less exclusive than a coterie.")

Definition 2.1. *Coterie.* Let U be the set of nodes that compose the system. A set of groups S is a *coterie* under U iff

(i) $G \in S$ implies that $G \neq \emptyset$, and $G \subseteq U$.
(ii) (Intersection property) If $G, H \in S$, then G and H must have at least one common node.
(iii) (Minimality) There are no $G, H \in S$ such that $G \subset H$.

When U is understood, we will drop it from the discussion. □

Note that not all nodes must appear in a coterie. For instance, $\{\{a\}\}$ is a coterie under $\{a, b, c\}$.

Definition 2.2. *Domination for Coteries.* Let R, S be coteries (under U). R *dominates* S iff $R \neq S$ and, for each $H \in S$, there is a $G \in R$ such that $G \subseteq H$. (We say that G is the group that dominates H.) □

Definition 2.3. *Dominated and Nondominated Coteries.* A coterie S (under U) is *dominated* iff there is another coterie (under U) which dominates S. If there is no such coterie, then S is *nondominated* (ND). □

As discussed in the introduction, a dominated coterie should not be used because there is a coterie that provides more protection against partitions. For instance, the coterie

$$\{\{a, b, c\}, \{c, d, e\}\}$$

should be replaced by $\{\{c\}\}$, and the coterie

$$\{\{a, b\}, \{b, c\}\}$$

should be replaced by

$$\{\{a, b\}, \{a, c\}, \{b, c\}\}$$

or by

$$\{\{b\}\}.$$

The next theorem gives us a way to check if a coterie is ND without enumerating all other coteries. This will be useful later on.

THEOREM 2.1. *Let S be a coterie under U. S is dominated iff there exists a group $G \subseteq U$ such that*

(i) G is not a superset of any group in S.
(ii) G has the intersection property. That is, for all $H \in S$, $G \cap H \neq \emptyset$.

PROOF. First we show that conditions (i) and (ii) imply S is dominated. There are two cases to consider. If there are one or more $H_1, H_2, \ldots, H_n \in S$ such that $G \subset H_1, H_2, \ldots, H_n$, then construct set $R = (S - H_1 - H_2 - \cdots - H_n) \cup G$. It is easy to see that R is a valid coterie (Definition 2.1) and that R dominates S. If there are no supersets of G in S, then $R = S \cup G$ is the dominating coterie.

Now, assume that S is dominated by coterie R. We show that conditions (i) and (ii) hold by considering two cases. In the first case, $S \subset R$. Let G be one of the elements in $R - S$. Set G must satisfy conditions (i) and (ii) or else R would not be a coterie. For the second case, $S \not\subset R$ and there must be an $H \in S$ and a $G \in R$ such that $G \subset H$ (see Definition 2.2). If condition (i) is false for G, then $G \supseteq H'$ for some $H' \in S$ and S is not a coterie because $H \supset G \supseteq H'$. Similarly, if condition (ii) does not hold for G, R would not be a coterie. (If $H' \in S$ and $H' \cap G' = \emptyset$, then $G \cap G' = \emptyset$, where G' is the group in R that dominates H'.) So in either case, the conditions hold. □

Checking domination of coteries seems to be a hard problem. The best algorithm that we know at this point is the one suggested by Theorem 2.1. It generates all the possible subsets of the universe of n nodes, and for each one, checks if it can be added to the coterie. This algorithm is clearly exponential in n. The worst-case complexity of an algorithm to check for nondomination is however an open problem. Nevertheless, the following two theorems show that the size of the coterie to be tested can be exponential in n in the worst case.

THEOREM 2.2. *The maximum number of groups in a coterie under a universe of n elements is bounded by 2^{n-1}.*

PROOF. Since all the groups in the coterie must intersect with each other, no group and its complement may be present. Thus, a coterie can have at most half of the possible subsets of the set of n elements. □

THEOREM 2.3. *There are coteries that have an exponential number of groups on n.*

PROOF. Consider the coteries with groups of size $\lceil (n+1)/2 \rceil$. There are $\binom{n}{(n+1)/2}$ such groups. Using the definition of combinations, it can be proved that $\binom{n}{(n+1)/2} > 2^n/n$. □

As an example of the coteries discussed in Theorem 2.3, we have

$$R = \{\{a, b, c\}, \{a, b, d\}, \{a, b, e\}, \{a, c, d\}, \{a, c, e\}, \{a, d, e\}, \{b, c, d\}, \{b, c, e\}, \{b, d, e\}, \{c, d, e\}\},$$

which consists of all the groups of size 3 in a universe of five nodes.

We end this section offering some results that establish the relationship between coteries and a combinatorial object called hypergraph. This connection proves to be useful in showing some properties of coteries and in establishing another characterization of ND-Coteries. (For a complete treatment of hypergraphs, see [2].)

Definition 2.4 [2]. *Hypergraph.* Let $X = \{X_1, \ldots, X_n\}$ be a finite set and $E = (E_i \mid i = 1, \ldots, m)$ a family of subsets of X. If $E_i \neq \emptyset$ $(i = 1, \ldots, m)$ and $\bigcup_i E_i \subseteq X$,[1] the couple $H = (X, E)$ is called a hypergraph. The value $|X| = n$ is the *order* of the hypergraph, the elements $X_1, \ldots, X_n$ are called the *vertices*, and the sets $E_1, \ldots, E_m$ are called the *hyperedges.* □

A coterie can be viewed as a hypergraph where the coterie groups are the hyperedges. However, not all hypergraphs represent coteries, since clearly the properties of intersection and minimality must be present.

Definition 2.5. *Transversal.* A transversal of a hypergraph $H = (X; E_1, \ldots, E_m)$ is defined to be a set $T \subset X$ such that $T \cap E_i \neq \emptyset$ for $(i = 1, \ldots, m)$. A *minimal transversal* is a transversal such that no proper subset of it is a transversal. The same concept is applicable to coteries. □

Definition 2.6. *Coloring of a Hypergraph.* A coloring of a hypergraph is a coloring of the nodes so every hyperedge has at least two colors. □

Definition 2.7. *Chromatic Number of a Hypergraph.* The chromatic number is defined to be the smallest number of colors needed for a hypergraph coloring. A hypergraph for which there exist a k-coloring is said to be k-colorable. □

The following theorem is equivalent to the Theorem 2.1

THEOREM 2.4. *A coterie is dominated iff the corresponding hypergraph is 2-colorable.*

PROOF. If a coterie S is dominated, according to Theorem 2.1 there exists a group G not in S that is not a superset of any group in S and that has the intersection property and, therefore, that can be added to S. It is enough to view G as one color class and the complement of it as the other. □

Recognizing 2-colorable hypergraphs is known to be NP-complete [10], which reenforces our belief that the same is true for coterie domination. However, since coteries are special hypergraphs, this result cannot be directly extended to the problem of coterie domination.

In [10], Lovasz observed that if a hypergraph has the intersection property and is not 2-colorable, it must have rather strict properties, and called those types of hypergraphs *strange.* Indeed, if we add the restriction of minimality, those hypergraphs correspond to our ND coteries.

A better characterization of ND coteries in terms of hypergraphs can be given using the following definitions.

[1] But this is not required for a coterie.

Definition 2.8. *Critical Hypergraph.* An edge E of a hypergraph H is critical if the chromatic number of $H - \{E\}$ is less than the chromatic number of H, that is, if deleting the hyperedge reduces the chromatic number. A hypergraph is critical if it is connected and each edge of it is critical. (Critical hypergraphs were introduced by Lovasz [9].) □

We can now prove that ND coteries are 3-chromatic critical hypergraphs.

Definition 2.9. *Used Nodes.* The nodes that actually appear in a coterie S are called the used nodes and are represented by $\mu(S)$. □

LEMMA 2.1. *For a given ND coterie R, different from the singleton coterie* $(\{a\})$, *there exists at least one partition of* $\mu(R)$ *into three pairwise disjoint subsets* $\mu_1/\mu_2/\mu_3$ *such that no group of R is completely contained in one of the subsets.*

PROOF. Let G be any group of R with two or more elements. Define

$$\mu_1 = G_1, \qquad \mu_2 = G_2, \qquad \mu_3 = \bar{G} = \mu(R) - G,$$

with

$$G_1 \cup G_2 = G, \qquad G_1 \cap G_2 = \varnothing,$$

with G_1 and G_2 each nonnull.

Group G is clearly not contained in any of these subsets. Similarly, any other $J \in R$ must have elements in μ_3 (else it would be a subset of G) and in $\mu_1 \cup \mu_2$ (else J would not intersect G'). Thus the theorem follows. □

Lemma 2.1 gives way to the following theorem:

THEOREM 2.5. *ND coteries are critical* 3-*chromatic hypergraphs.*

PROOF. Dropping a group from a ND coterie will produce a dominated coterie; hence, according to Definition 2.8, ND coteries are critical hypergraphs. By Lemma 2.1, they are 3-colorable. □

Using Theorem 2.5, we can see that the following lemma is in accordance with a theorem proved by Benzaken in [1] that shows that a hypergraph H is 3-chromatic and critical iff the hyperedges of H are all minimal transversals of H.

LEMMA 2.2. *Let S be a ND coterie and G a transversal of it. Then G is either in S or a superset of a group in S.*

PROOF. Assume that there exists a group $J \notin S$ such that J is not a superset of any group in S, and J is a transversal of S. Then J would fulfill the intersection and minimality properties of Theorem 2.1, so S would not be an ND coterie. □

3. *Vote Assignments*

In this section we study vote assignments and show some of their properties.

Definition 3.1. *Vote Assignment.* Let U be the set of nodes that compose the system. A *vote* assignment is a function $v: U \to \mathbf{N}$. (**N** is the nonnegative integers.) $v(a)$ is the number of votes assigned to node a. □

Definition 3.2. *Total and Majority.* For a vote assignment v over U, TOT(v) and MAJ(v) are defined by

$$\mathrm{TOT}(v) = \sum_{a \in U} v(a)$$

and

$$\text{MAJ}(v) = \begin{cases} \dfrac{\text{TOT}(v)}{2} + 1 & \text{if } \text{TOT}(v) \text{ is even}, \\ \dfrac{\text{TOT}(v) + 1}{2} & \text{if } \text{TOT}(v) \text{ is odd}. \end{cases}$$ □

Each vote assignment implicitly defines a set of groups of nodes that may be active, that is, those holding a majority of the votes. This is formalized by the next definition.

Definition 3.3. *The Coterie Corresponding to a Vote Assignment.* Let v be a vote assignment over U. Let

$$Z = \left\{ G \mid G \subseteq U \text{ and } \sum_{a \in G} v(a) \geq \text{MAJ}(v) \right\}$$

and call the elements of Z the *majority groups.* The minimal elements of Z (i.e., those sets such that no subset of them is in Z) are the *tight majority groups* and constitute the *coterie corresponding to v.* (The set of tight majority groups is clearly a coterie, since it satisfies the properties of Definition 2.1.) □

Coteries may have more than a single vote assignment corresponding to them. For example, assignments $v(a) = 1$, $v(b) = 1$, $v(c) = 1$, and $w(a) = 2$, $w(b) = 2$, $w(c) = 3$ correspond to the same coterie, and we call them *similar assignments.*

Definition 3.4. *Similar Vote Assignments.* Two vote assignments are *similar* iff their corresponding coteries are equal. □

As we shall see shortly, some coteries do not have any corresponding vote assignments. As a matter of fact, even nondominated coteries may have no vote assignment. But first, let us define domination for vote assignments.

Definition 3.5. *Domination for Vote Assignments.* Let v, w be two vote assignments (under U), and let R, S be their corresponding coteries. v *dominates* w iff R dominates S. Similarly, we say that v is *dominated* if R is dominated; and v is nondominated (ND) if R is ND. □

THEOREM 3.1. *There are ND coteries such that no vote assignment corresponds to them.*

PROOF (by example). Let $U = \{a, b, c, d, e, f\}$ and consider the coterie

$$S = \{\{a, b\}, \{a, c, d\}, \{a, c, e\}, \{a, d, f\},$$
$$\{a, e, f\}, \{b, c, f\}, \{b, d, e\}\}.$$

We omit the proof that S is ND since it is simple but tedious. (We have to show that there is no set satisfying the properties of Theorem 2.1. I.e. we enumerate all sets that intersect the groups of S and verify that all such groups are already included in S.) Suppose that there is a vote assignment v corresponding to S. From Definition 3.3, v would have to satisfy the inequalities

$$v(a) + v(b) \geq \text{MAJ}(v),$$
$$v(a) + v(c) + v(d) \geq \text{MAJ}(v),$$
$$v(a) + v(c) + v(e) \geq \text{MAJ}(v),$$
$$\vdots$$

From these inequalities, it is easy to reach a contradiction. For instance, since $\{b, d, e\}$ is in S but $\{a, d, e\}$ is not, $v(b) + v(d) + v(e) \geq \mathrm{MAJ}(v)$ and $v(a) + v(d) + v(e) < \mathrm{MAJ}(v)$. Therefore, $v(b) > v(a)$. This fact, plus the second inequality above implies that $v(b) + v(c) + v(d) \geq \mathrm{MAJ}((v)$ and $\{b, c, d\}$ or a subset should be in S. This is not true. □

Theorem 3.1 tells us that coteries are a more powerful concept than vote assignments. That is, in some systems a coterie like the one in the theorem could actually yield the best reliability characteristics, and there would be no way to enforce the same groups with votes. However, vote assignments have some advantages over coteries (e.g., easier to implement), but we defer this discussion until after we discover some more properties of vote assignments and coteries.

In Section 2 we have shown that checking domination for coteries is hard; the next two theorems tell us that checking domination for vote assignments is trivial.

THEOREM 3.2. *Let v be a vote assignment for nodes U. If $TOT(v)$ is odd, then v is nondominated.*

PROOF (by contradiction). Assume that v is dominated. Let S be the (dominated) coterie corresponding to v. By Theorem 2.1, there is a group G that satisfies the intersection property but is not a superset of a group in S. Notice that G cannot be a majority group (Definition 4.3) because then it or a subset would be in S. Now, consider the group $U - G$. Since $\mathrm{TOT}(v)$ is odd, $U - G$ must be a majority group (i.e., there is no way to split the nodes into two groups without having one of them contain a majority of votes). Thus, $U - G$ or a subset must be in S. This is a contradiction, since G, which was supposed to intersect all sets in S, clearly does not intersect $U - G$ or its subsets. □

If $\mathrm{TOT}(v)$ is even, v may or may not be ND. For instance, the assignment $v(a) = 1$, $v(b) = 1$, $v(c) = 1$, and $v(d) = 1$ is an even assignment that is not ND, whereas the assignment $v(a) = 4$, $v(b) = 2$, $v(c) = 2$, and $v(d) = 2$ is ND.

However, as we shall show shortly, we can easily transform v into an ND odd assignment which is either similar to v or dominates v. Thus, we should always use the transformed assignment instead of v.

THEOREM 3.3. *Let v be a vote assignment for nodes U, where $TOT(v)$ is even. Let w be an assignment with $TOT(w) = TOT(v) + 1$ that distributes the $TOT(v)$ votes as v does and gives any node the extra vote. Then w is ND and either is similar to v or dominates v.*

PROOF. w is ND by the previous theorem. Since $\mathrm{MAJ}(w) = \mathrm{MAJ}(v)$, any group in the coterie for v will be in the coterie for w. This means that either the coterie for v equals the coterie for w (assignments similar) or the v coterie is a subset (w dominates v). □

Theorems 3.2 and 3.3 imply that it is simple to avoid dominated vote assignments. In general, this is not the case for coteries.

4. *Counting Coteries and Vote Assignments*

In this section we attempt to enumerate ND coteries. Shortly, we shall discover that any algorithm that attempts to enumerate all coteries will be inefficient due simply to the number of ND coteries that exist. However, since our aim is to optimize the choice of coterie in a given system, we need some way of enumerating the choices, or at least a convenient subset of them.

We enumerate coteries instead of vote assignments since they are more general (Theorem 3.1) and there are no "duplicates" (i.e., similar vote assignments). We return to vote assignments in the following section.

Before enumerating ND coteries, we must address the issue of isomorphic coteries. For instance, consider a system with nodes a, b, c, and d, and the two coteries

$$S = \{\{a, b, c\}, \{a, d\}, \{b, d\}, \{c, d\}\},$$
$$R = \{\{b, c, d\}, \{b, a\}, \{c, a\}, \{d, a\}\}.$$

Clearly, R is isomorphic to S; that is, if in S, we exchange a and d we obtain R. In enumerating coteries, we prefer not to list isomorphic ones. Instead, we only list a *representative* of each category, from which the isomorphic ones can easily be generated.

Definition 4.1. *Isomorphic Coteries.* Two coteries R and S under $U = \{a_1, a_2, \ldots, a_n\}$ are *isomorphic* iff there is a permutation Π (of the integers $1 \cdots n$) such that when we replace each a_i in S by $a_{\Pi(i)}$ we obtain R. □

Definition 4.2. *Enumeration.* A set of ND coteries (under U), E, is an *enumeration* (under U) iff

(i) every ND coterie (under U) is either in E or is isomorphic to one in E; and
(ii) no two coteries in E are isomorphic.

To avoid confusion, we enclose enumerations in square brackets instead of in set brackets (i.e., $E = [\ldots]$). □

One strategy for producing enumerations is to generate all possible coteries and then eliminate dominated and isomorphic ones. For references, let us call this strategy the *exhaustive algorithm.* If we use it for a system with a single node, we easily obtain the one and only enumeration:

Observation 4.1. The set $[\{\}, \{\{a\}\}]$ is an enumeration (and the only one) for $U = \{a\}$. □

For systems with more nodes, we quickly discover that the enumeration algorithm is too expensive. To formalize this assertion, we have to find a way of counting ND coteries. In order to do that, the following lemma establishes a way of constructing an ND coterie under U with $|U| = n$ elements from a coterie under U' with $|U'| = n - 1$ in a way that there is a one-to-one correspondence between the two.

LEMMA 4.1. *Consider a nonempty coterie R under $U' = \{a_1, a_2, \ldots, a_{n-1}\}$. Let Γ be the set of minimal transversals of R. From the set S that contains the groups of R plus all groups of the form $T \cup \{n\}$ where $T \in \Gamma$ and T not in R. Then S is an ND coterie under $U = \{a_1, a_2, \ldots, a_n\}$.*

PROOF. S is clearly a coterie since every new group added intersects with the rest of the groups, and no group is contained in another. For the ND part, assume S is dominated. Then, by Theorem 2.1, there exists a G not in S that has the intersection property and is not a superset of any group in S. There are two cases to consider:

(a) $n \in G$. In this case, since G has the intersection property, there must be a minimal transversal of R, T, such that $T \subset G$ and therefore $T \cup \{n\} \subseteq G$, so G is a superset of some group in S.

(b) $n \notin G$. Since G is not a superset of any group in S, then $\bar{G}$ has the intersection property. Therefore, there must be a minimal transversal in R, $T \subseteq \bar{G}$ and $(T \cup \{n\}) \cap G = \emptyset$. Thus, G does not have the intersection property as was claimed. □

To illustrate the transformation of Lemma 4.1, consider the coterie

$$R = \{\{a, b\}, \{a, c\}\},$$

and let $U = \{a, b, c, d\}$. The set $\Gamma = \{\{a\}, \{b, c\}\}$ contains all the minimal transversals for R, so we can form

$$S = \{\{a, b\}, \{a, c\}, \{a, d\}, \{b, c, d\}\},$$

which is an ND coterie.

It is easy to see that distinct R will give distinct ND coteries S and that R can be recovered from S by dropping all groups that contain the nth element. Now we can use this fact to count the number of ND coteries with n elements counting the number of distinct coteries of $n - 1$ elements.

THEOREM 4.1. (M. Yannakakis, personal communication). *The value $2^{2^{cn}}$ for some constant c is a lower bound of the number of ND coteries under a universe of n elements.*

PROOF. Assume that n is odd (the case for n even is handled similarly). Consider only coteries under $U' = \{a_1, \ldots, a_{n-1}\}$ with groups of size $(n + 1)/2$. That is, consider sets of the form $\{\{b_1, \ldots, b_m\} \mid b_i \in U' \text{ and } m = (n + 1)/2\}$. These sets are coteries since no group can possibly be a superset of another and any two groups will intersect. There are $\left[{n-1 \atop (n+1)/2}\right]$ potential groups of that size and, therefore, $2^{[n-1,(n+1)/2]}$ coteries under U' of this form. Now, since $\left[{n-1 \atop (n+1)/2}\right] > 2^n/(n + 1)$, and, for each of these coteries, there is an ND coterie under U, the result follows. □

Theorem 4.1 shows unequivocally that any attempt to enumerate all the ND coteries is hopeless for large n. Besides having a doubly exponential number of them, some of them are exponentially long, as seen in Theorem 2.3. Thus, the sample space for the optimization problem is really huge.

Although we have just shown that the total number of coteries is large, there are two important questions that arise:

(1) Can we enumerate them for small systems? As we stated in the introduction, these are the most probable systems in practice.
(2) Even for large systems, can we at least enumerate a subset of the class of ND coteries? From a practical point of view, optimizing over this subclass may be good enough. However, we are interested in covering at least the original choices in our problem, that is, those coteries that have a corresponding vote assignment. Therefore, we concentrate on finding a subclass of ND coteries that results in a more manageable number of choices and that includes those ND coteries with vote assignment.

We now develop an algorithm that will let us answer both of these questions positively. This algorithm is based on a transformation that generates ND coteries from other ones and is, hence, called the transformation algorithm (TA). The idea of the transformation is to delete one group of a coterie and to add the complement and a set of groups that are supersets of the deleted group, making the coterie grow in size. We prove that this procedure, when applied to ND coteries, produces new

ND coteries. The definitions that follow will lead to this algorithm. (This transformation may also be useful in an optimization procedure. That is, if we have found a good coterie, the transformation will give us related coteries that may be superior.)

Definition 4.3. *Coterie Transformation.* The coterie transformation CT is a function that takes as inputs an ND coterie S and two sets of nodes, G and U, and yields a set of groups. Let

$$\bar{G} = U - G$$

and

$$\beta = \{G \cup \{b\} \mid b \in \bar{G} \text{ and } H \not\subseteq G \cup \{b\} \text{ for any } H \in S - G\}.$$

The transformation CT(S, G, U) is undefined if any of the following conditions are *false*:

(i) $G \in S$,
(ii) $\mu(S) \subseteq U$,
(iii) $|\bar{G}| \geq 2$,
(iv) $|\beta| \geq 1$,
(v) S is ND.

If the conditions hold, then

$$\text{CT}(S, G, U) = (S - \{G\}) \cup \{\bar{G}\} \cup \beta.$$

We call the groups in β the *new* groups. □

To illustrate the use of the transformation, take $S = \{\{a\}\}$, $G = \{a\}$, and $U = \{a, b, c\}$. Since the conditions hold, we have $\bar{G} = \{b, c\}$ and $\beta = \{\{a, b\}, \{a, c\}\}$; therefore, CT($S$, G, U) $= \{\{b, c\}, \{a, b\}, \{a, c\}\}$. The resulting set of groups is an ND coterie, and as will prove later, this is no coincidence. If we change U to $\{a, b\}$, condition (iii) is violated; so CT is undefined. If we were to go ahead and compute CT as $(S - \{G\}) \cup \bar{G} \cup \beta$, we would get $\{\{b\}, \{a, b\}\}$, which is not even a coterie. Similarly, if condition (iv) is false, we would not get an ND coterie. For example, make $S = \{\{a, b, c\}, \{a, d, e\}, \{b, d\}, \{c, d\}, \{b, c, e\}\}$, $G = \{a, b, c\}$, and $U = \{a, b, c, d, e\}$.

The algorithm for partial enumeration of ND coteries is shown in Figure 1. The algorithm calls the nodes $a_1, a_2, \ldots, a_m$. It takes as input integer i and outputs a set of coteries under $\{a_1, a_2, \ldots, a_i\}$. We use *p-enum* for both the name of the algorithm and the set it returns as a result.

As an example, suppose we evaluate *p-enum* (2). The algorithm starts by storing *p-enum*(1) = [{ }, $\{\{a_1\}\}$] in *temp* and *p-enum.* The algorithm then considers each coterie in *temp* and attempts to transform it into a coterie with more groups. In this case, neither coterie can be transformed, so *p-enum*(2) = *p-enum*(1).

If we compute *p-enum*(3), we are able to tranform $\{\{a_1\}\}$ into $\{\{a_2, a_3\}\}$, $\{a_1, a_2\}$, $\{a_1, a_3\}\}$ (as illustrated before). This means that *p-enum*(3) will contain this coterie plus the ones in *p-enum*(2). In a similar fashion, we can compute *p-enum*(3), *p-enum*(4), and *p-enum*(5), which are displayed in Figure 2. We do not display higher ordered sets since they are much larger.

The algorithm for producing partial enumeration has some interesting properties. It does produce total enumerations for "small" systems, which we believe are the most common ones. And as we shall see in the following section, it enumerates all the ND coteries that correspond to a vote assignment.

```
function p-enum (i);
  if i ≤1 then p-enum ← /{ }, {{a1}}/
  else
      begin
          temp ← p-enum (i-1);
          p-enum ← temp;
          while temp ≠ ∅ do
              begin
                  select S ∈ temp;
                  temp ← temp - S;
                  for each G ∈ S do
                      begin
                          H←CT(S,G,{a1,...ai});
                          if "H is defined and neither it nor an
                            isomorphism is in p-enum" then
                              begin
                                  p-enum ← p-enum ∪ H;
                                  temp ← temp ∪ H;
                              end;
                      end;
              end;
      end;
```

FIG. 1. The algorithm for partial enumeration of ND.

$p\text{-}enum(1) = /\{\ \}, \{\{a\}\}/$

$p\text{-}enum(2) = /\{\ \}, \{\{a\}\}/$

$p\text{-}enum(3) = /\{\ \}, \{\{a\}\}, \{\{a,b\},\{a,c\},\{b,c\}\}/$

$p\text{-}enum(4) = /\{\ \}, \{\{a\}\}, \{a,b\},\{a,c\},\{b,c\}\},$
$\{\{a,b,c\},\{a,d\},\{b,d\},\{c,d\}\}/$

$p\text{-}enum(5) = /\{\ \}$
$\{\{a\}\},$
$\{\{a,b\},\{a,c\},\{b,c\}\},$
$\{\{a,b,c\},\{a,d\},\{b,d\},\{c,d\}\},$
$\{\{a,b,c\},\{b,d\},\{c,d\},\{b,c,e\},\{a,d,e\}\},$
$\{\{a,b,c\},\{c,d\},\{b,c,e\},\{a,d,e\},$
$\{a,c,e\},\{b,d,a\},\{b,d,e\}\},$
$\{\{a,b,c\},\{b,c,e\},\{a,d,e\},$
$\{a,c,e\}\{b,d,a\},\{b,d,e\},$
$\{a,b,e\},\{c,d,a\},\{c,d,b\},\{c,d,e\}\},$
$\{\{a,b,c,d\},\{a,e\},\{b,e\},\{c,e\},\{d,e\}\}/$

FIG. 2. The algorithm for partial enumeration of ND, using $\{a, b, c, d, e\}$ instead of $\{a_1, a_2, a_3, a_4, a_5\}$.

In the rest of this section, we prove the properties of the CT transformation and of the partial enumeration algorithm we have alluded to. We start by showing some simple but useful results for ND coteries.

LEMMA 4.2. *Say S is an ND coterie, G an element of it, and U a set of nodes such that $\mu(S) \subseteq U$. Let $\bar{G} = U - G$. Then,*

(*i*) *For all $H \in S$ such that $H \neq G$, $\bar{G} \cap H \neq \varnothing$.*
(*ii*) *For all $H \in S$, $H \not\subseteq \bar{G}$.*
(*iii*) *For each $a \in G$, the group $\bar{G} \cup \{a\}$ or a subset of it must be in S.*

PROOF

(i) If there were such an H, $H \cap \bar{G} = \varnothing$, $H \subseteq G$, and S would not be a coterie (Definition 2.1).
(ii) If $H \subseteq \bar{G}$, then $H \cap G = \varnothing$ and S is not a coterie.
(iii) If neither $\bar{G} \cup \{a\}$ nor a subset is in S, we can use this group to show that S is dominated (Theorem 2.1). Note that by (i), $\bar{G} \cup \{a\}$ intersects all S groups except G. But since $a \in G$, $\bar{G} \cup \{a\}$ also intersects G. □

THEOREM 4.2. *If the set produced by the CT transformation, CT(S, G, U), is defined, then it is a coterie.*

PROOF. We must show that the coterie properties (Definition 2.1) hold. The proofs for most cases are trivial because S is a coterie. Here we only cover the interesting cases. To show minimality:

(1) No $H \in S$ can be a proper subset of $\bar{G}$. True by Lemma 4.2.
(2) $\bar{G}$ should not be a proper subset of a new group (i.e., of a group in β). Each new group has a single $\bar{G}$ element, and $|\bar{G}| \geq 2$.
(3) $\bar{G}$ cannot be a proper subset of an $H \in S$. Suppose not; that is, $\bar{G} \subset H$. Let us write H as $\bar{G} \cup I (I \neq \varnothing)$. Sinc $|\beta| \geq 1$, there must be a set $G \cup \{a\}$ $(a \in \bar{G})$ such that no group in S is included in it. This means that all S members must have a member not in $G \cup \{a\}$, that is, a member in $\bar{G} - \{a\}$. With this information, we can show that S is dominated (a contradiction): Let $J = H - \{a\} = (\bar{G} - \{a\}) \cup I$. This set cannot be equal to or a superset of some other S group (or else S would not be a coterie). Also, J intersects all S groups (then all have an element of $\bar{G} - \{a\}$). Thus, S is dominated.

To show the intersection property for CT(S, G, U), there is only one nontrivial case to consider:

(4) If $H \in S$, $H, \neq G$, $H \cap \bar{G} \neq \varnothing$. True by Lemma 4.2. □

THEOREM 4.3. *If CT(S, G, U) is defined, then it is an ND coterie.*

PROOF. By the previous theorem, it is a coterie. To show it is ND, we assume it is dominated and use Theorem 2.1 to show that S is dominated (a contradiction). Let I be the set guaranteed by Theorem 2.1 for a dominated CT((S, G, U). This same I will prove that S is dominated.

Since all groups of S (except G) are in CT, we already know that I intersects these groups and is not equal to or a superset of them. We just have to show that these same properties hold for G.

(1) $G \not\subseteq$ I. Clearly, $I \neq G$, for else I and $\bar{G}$ (a group of CT) would not intersect. If $G \subset I$, I must contain at least one node $c \in \bar{G}$ that does no appear in a new group (else I would be a superset of the new groups). If c is not in a new group, there must be an $H \in S$ such that $H \subseteq G \cup \{c\} \subset I$, which is impossible.
(2) $I \cap G \neq \varnothing$. Suppose not. Then, $I \subset \bar{G}$ (since $I \neq \bar{G}$). Let c be one of the nodes in $\bar{G} - I$. This means that c cannot be one of the elements of $\bar{G}$ selected to form a new group, or else I would not intersect that group. But if c was not used, it must be because $H \subseteq G \cup \{c\}$ for some $H \in S$; and I does not intersect this H (a contradiction). □

The following three lemmas will be useful for proving the next theorem.

LEMMA 4.3. *Let U and U′ be two sets of nodes such that $U \subseteq U'$. If S is an ND coterie under U, then it is also an ND coterie under U′.*

PROOF. Suppose that S is dominated under U'. By Theorem 2.1, there is $G \subseteq U'$ not equal to or a superset of an S group, such that the intersection property holds. Since $\mu(S) \subseteq U$, we can remove from G the nodes in $U' - U$ without violating the properties of G. The existence of this new set implies that S is dominated under U, a contradiction. □

The next two lemmas describe the conditions that must hold for a coterie to be obtainable from another one through the CT transformation.

LEMMA 4.4. *Let R be an ND coterie. There exists an ND coterie S, $|S| < |R|$, such that CT transforms S into R iff there exist groups $I, H \in R$ such that $\mu(R) - H \subset I$, $|H| \geq 2$, and $|I - (\mu(R) - H)| = 1$.*

PROOF. If $R = \mathrm{CT}(S, G, U)$, $\mu(R) = U$ and $\bar{G}$, $G \cup \{a\}$ are elements of R ($a \in \bar{G}$). If we refer to $\bar{G}$ as H, and $G \cup \{a\}$ as I, we see that $\mu(R) - H \subset I$, $|H| \geq 2$, and $|I - (\mu(R) - H)| = 1$.

To prove in the opposite direction, assume that the sets I, H exist in R. Let

$$\gamma = \{G \mid G \in R \text{ and } \mu(R) - H \subset G\}, \qquad \bar{H} = \mu(R) - H,$$

and construct set

$$S = (R - \gamma - \{H\}) \cup \{\bar{H}\}.$$

We can check that $\mathrm{CT}(S, \bar{H}, \mu(R))$ is defined and yields R. Simply notice that, by Lemma 4.2(iii), all the groups G in γ must be of the form $\bar{H} \cup \{a\}$, with $a \in H$. With this in mind, we can see that the groups in γ are exactly the groups in the set β produced by CT. Also, since $|\gamma| \geq 1$, $|S| < |R|$. The last thing to check is that S is an ND coterie. Since all groups in S except $\bar{H}$ were in ND coterie R, we only have to verify that $\bar{H}$ in S does not create problems.

$\bar{H}$ cannot be a subset of a $J \in S$ because any such J would be in γ and not in S. A $J \in S$ cannot be a subset of $\bar{H}$, or else $J \cap H = \varnothing$ and R is not a coterie. $\bar{H}$ must interest each $J \in S$, for else $J \subseteq H$. Thus, S is a coterie.

To show that S is ND, assume the contrary. Let J be the group guaranteed by Theorem 2.1. There are two cases to consider:

Case 1: $J \not\supseteq H$. In this case we can use J directly to prove that R is dominated (a contradiction). We omit this part of the proof since it is straightforward.

Case 2: $J \supseteq H$. Here, J is not useful for showing R dominated. However, we can construct a new set for this. Say a is the node in $I - \bar{H}$, and construct $K = J - \{a\}$ ($a \in H$). It is easy to see that K is not equal to or a superset of any R group, including H. But now we must check that K still has the interesection property in R.

Since $J \cap \bar{H}$ is nonempty and not equal to $\{a\}$, $K \cap \bar{H}$ has the same properties. Hence, K intersects all of the new groups (of the form $\bar{H} \cup \{c\}$). Since $H \subseteq J$ and $|H| \geq 2$, J and H have at least two common elements, and thus $K \cap H \neq \varnothing$. Finally, take an $L \in S$ and R. We know that $J \cap L \neq \varnothing$, but suppose that $K \cap L = \varnothing$. This means that L is made up of a plus nodes in $\bar{H}$. Since there is a set $I = \{a\} \cup \bar{H}$ in R, and L cannot be a subset of it, $L = I$. But then L would not be in S. □

LEMMA 4.5. *If R is a ND coterie with $2 \leq \mu(R) \leq 5$, then there exist groups I, $H \in R$ such that $\bar{H} \subset I$, $|H| \geq 2$, and $|I - \bar{H}| = 1$ (where $\bar{H} = \mu(R) - H$).*

PROOF. We do not have to consider coteries with a single-node group because they must be of the form $\{\{a\}\}$ and $\mu(R) < 2$. If R only has groups with two nodes, it must be of the form $\{\{a, b\}, \{a, c\}, \{b, c\}\}$, and the sets I, H clearly exist. If R has a group $\{a, b, c, d\}$, $\mu(R)$ must be 5 (else it would be dominated) and $\{a, e\}$ must be in R (Lemma 4.2). Again, I and H exist. Also, R can never have a group $\{a, b, c, d, e\}$.

Thus, the only "interesting" case is when R has at least one 3-group, say $\{a, b, c\}$, and possibly some 2-groups. If $\mu(R) = 4$, $\{a, d\}$ must be in R too (Lemma 4.2), and we have found I and H. So, assume that $\mu(R) = 5$. We make $H = \{a, b, c\}$ and we will find an I satisfying the properties. By Lemma 4.2, we must have in R groups J, K, L such that $a \in J$, $b \in K$, $c \in L$, and $J \subseteq \{d, e, a\}$, $K \subseteq \{d, e, b\}$, $L \subseteq \{d, e, c\}$. If one of these groups contains d and e, then it is the I we are looking for. But, suppose none of them have both d and e. Without loss of generality, say, $J = \{d, a\}$. Then $K = \{d, b\}$ and $L = \{d, c\}$ (to intersect J). Now, the set containing e (and there must be one since $\mu(R) = 5$) must have a, b, c as elements, so it is a superset of H, a contradiction. Therefore, one of the groups J, K, L has three nodes and I exists. □

THEOREM 4.4. *If* $1 \le i \le 5$, *p-enum*(i) *is a total enumeration under* $\{a_1, a_2, \ldots, a_i\}$. *(Recall that p-enum*$(i)$ *is computed by the algorithm of Figure* 1.)

PROOF. Since the partial enumeration algorithm explicitly removes isomorphic coteries, we only have to show that every ND coterie R or one isomorphic to it is in *p-enum*(i) (Definition 4.2). We do this by induction on j, the number of groups in R. The algorithm always generates coteries $\{\}$, $\{\{a_1\}\}$, and all ND coteries with $j = 0$ or 1 are isomorphic to these. Now, assume that the algorithm generates all ND coteries (or isomorphic ones) with $j < m$ $(m \ge 2)$.

Take ND coterie R with $j = m$. By Lemmas 4.4 and 4.5, there exists an ND coterie S, $|S| < m$; such that S can be transformed into R. Since either S or a coterie isomorphic to it is in *p-enum*(i), the algorithm will generate R, or an isomorphic coterie. □

THEOREM 4.5. *For* $i \ge 6$, *p-enum*(i) *is not a total enumeration under* $\{a_1, a_2, \ldots, a_i\}$.

PROOF. Take the ND coterie S of Theorem 3.1. By Lemma 4.3, S (or an isomorphic coterie) must be in any total enumeration under $\{a_1, \ldots, a_i\}$. However, there are no two sets H, I in S satisfying the properties of Lemma 4.4, so S will never be in *p-enum*(i). □

5. *Vote Assignments Revisited*

If we try to find equivalent vote assignments for the ND coteries for systems with five or fewer nodes (Figure 2), we discover that every single one has an assignment. (To find an assignment, pose linear equations as was done in Theorem 3.1.) This raises some intriguing questions. Do all coteries in *p-enum*(i), $i \ge 6$, also have a corresponding vote assignment? Or are all coteries with ND vote assignments in *p-enum*(i)? Answers to these questions will shed light on the relationship between coteries and assignments, and could give us a way to generate coteries with vote assignments. In this section we answer these questions.

In the following two lemmas, we show that every ND coterie with a vote assignment can be obtained from a smaller coterie, also with an assignment, through our coterie transformation CT. An example that illustrates how to construct the smaller coterie follows the lemma.

LEMMA 5.1. *Let R be an ND coterie with vote assignment v, and let MAJ(v) be its majority. Suppose that there is a node $c \in \mu(R)$ that only participates in groups of R that have strictly more votes than MAJ(v). Then we can find a similar vote assignment w where this is not the case. That is, the coterie corresponding to w will be R and node c will be a member of a group $G \in R$ with exactly MAJ(w) votes.*

PROOF. Let us assume that TOT(v) is odd. (If it is not, we convert v to a similar odd assignment using Theorem 3.3.) Let $G \in R$ be the group that contains c and has the least number of votes, say MAJ(v) + $k(k \geq 1)$. Transform v into a new assignment w by subtracting $2k - 1$ votes from c. The new assignment will have TOT(w) = TOT(v) − ($2k - 1$) and MAJ(w) = MAJ(v) − ($k - 1$). Note that every group in R will continue to have a majority under w. This is clearly true for groups that do not contain c and thus lost no votes. Groups containing c had at least k votes to spare, lost ($2k - 1$), but still have a majority because MAJ(w) is ($k - 1$) votes less than MAJ(v). Since all R groups still have a majority, w either dominates v or is similar. Since R is ND, w must be equivalent to v. Finally, observe that under w, group G, containing c, has exactly MAJ(w) votes. □

LEMMA 5.2. *Let R be an ND coterie (different from the singleton coterie) with a corresponding vote assignment. There exists an ND coterie S with corresponding vote assignment such that $|S| < |R|$ and CT transforms S into R.*

PROOF. We proceed to construct the desired coterie. Let c be a node with the least number of votes in the assignment corresponding to R. If necessary, use Lemma 5.1 to convert the assignment into one where c participates in a group H with an exact majority. If the resulting assignment has an even number of votes, add one vote to any node in H (Theorem 3.3), so that H continues to have an exact majority. Let v be this resulting assignment and let $\bar{H} = \mu(R) - H$. Say $|H| = n$ and $|\bar{H}| = m$. (Note that c continues to be a node with the least number of votes.) We now transform v into the desired assignment w in two steps:

(1) multiply the votes of each node under v by $f = 2nm - 1$.
(2) subtract m votes from each node in H, and add n votes to each node in $\bar{H}$.

Let S be the coterie defined by this new vote assignment. Note that

$$\mathrm{MAJ}(v) = \frac{\mathrm{TOT}(v) + 1}{2},$$

$$\mathrm{TOT}(w) = f\,\mathrm{TOT}(v),$$

$$\mathrm{MAJ}(w) = \frac{\mathrm{TOT}(w) + 1}{2} = f\,\mathrm{MAJ}(v) - (nm - 1).$$

Since f is odd, TOT(w) will be also odd and hence S will be ND.

We will now show that

$$R = S - \{\bar{H}\} \cup \{H\} \cup \beta,$$

where

$$\beta = \{\bar{H} \cup \{a\} \mid a \in H\}$$

and therefore that S is the ND coterie we are searching for: $|S| < |R|$ and CT(S, $\bar{H}$, $\mu(R)$) = R.

Note that group H has MAJ(v) votes in R, f MAJ(v) votes after step (1) and fMAJ(v) − nm votes after step (2). Thus, it is not in S. Similarly, group $\bar{H}$ is not in R but is in S since it has f(MAJ(v) − 1) + nm = MAJ(w) votes. ($|\beta| \geq 1$ because $\bar{H} \cup \{c\}$ must be in S.) Furthermore, groups in β will not be in S but will be in R. However, note that groups of the form $H \cup \{a\}$ $a \in \bar{H}$, will *not* appear in S. Such groups do have a majority under w, but have an unnecessary node, mainly node c. (Under w, H is just one vote under the majority. Since c had the least number of votes in v, and in the transformation c lost and a gained votes, then $w(a) - w(c) \geq 1$.) To show that there are no additional differences between S and R, note that any group in R or S not containing H or $\bar{H}$ will not change its majority status in the transformation. That is, a loss or gain of nm votes is necessary to lose or gain a majority, and only groups containing H or $\bar{H}$ are in this situation. □

To illustrate the transformations of the lemmas, consider the vote assignment

$$v_1(a) = 6, \qquad v_1(b) = 3, \qquad v_1(c) = 3, \qquad v_1(d) = 3,$$
$$(\text{TOT}(v_1) = 15, \ \text{MAJ}(v_1) = 8),$$

corresponding to ND coterie

$$R = \{\{a, b\}, \{a, c\}, \{a, d\}, \{b, c, d\}\}.$$

Let us use node c as our node with minimum votes. Since there are no exact majority groups, we transform v_1, by subtracting 2 (1) − 1 votes from c. Since the new total is even, we add one vote to, say, d to obtain

$$v_2(a) = 6, \qquad v_2(b) = 3, \qquad v_2(c) = 2, \qquad v_2(d) = 4,$$
$$(\text{TOT}(v_2) = 15, \ \text{MAJ}(v_2) = 8),$$

still corresponding to R.

Applying the transformation of Lemma 5.2, we choose, for instance, $H = \{a, c\}$ and multiply v_2 by 2(2) − 1. Subtracting 2 votes from a and c, and adding 2 to b and d, we get

$$v_3(a) = 16, \qquad v_3(b) = 11, \qquad v_3(c) = 4, \qquad v_3(d) = 14,$$
$$(\text{TOT}(w_3) = 45, \ \text{MAJ}(w_3) = 23).$$

This corresponds to

$$S = \{\{a, b\}, \{a, d\}, \{b, d\}\}.$$

It is easy to check that CT(S, $\{b, d\}$, $\{a, b, c, d\}$) gives us back R.

THEOREM 5.1. *If R is the coterie corresponding to ND vote assignment v (under $\{a_1, \ldots, a_i\}$), then R or an isomorphic coterie will be in p-enum(i).*

PROOF. The proof is similar to the one for Theorem 4.4. Lemma 4.4 and 5.2 tell us that any such R (or an isomorphic one) can be obtained from a lower order ND coterie that also has a vote assignment and it is thus generated by the partial enumeration algorithm. □

THEOREM 5.2. *Not all coteries generated by the partial enumeration algorithm have a corresponding vote assignment.*

PROOF (by example). Start with coterie

$$R_1 = \{\{a, b, c\}, \{a, d, e\}, \{b, d\}, \{c, d\}, \{b, c, e\}\},$$

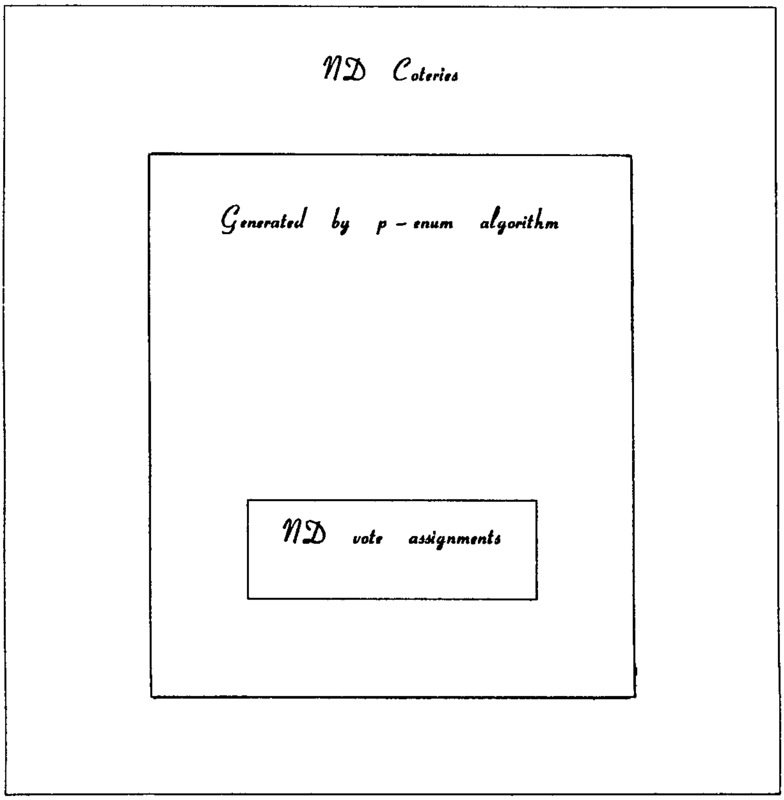

FIG. 3. Venn diagram.

which is in *p-enum*(5). Using $G = \{b, d\}$, $U = \{a, b, c, d, e, f\}$, transform R_1 into

$$R_2 = \{\{a, b, c\}, \{a, d, e\}, \{c, d\}, \{b, c, e\}, \{a, c, e, f\}, \{b, d, a\}, \{b, d, e\}, \{b, d, f\}\}$$

(R_2 still has a vote assignment). Now, using $G = \{a, b, c\}$, U as before, we obtain

$$R_3 = \{\{a, d, e\}\{c, d\}, \{b, c, e\}, \{a, c, e, f\}, \{b, d, a\}, \{b, d, e\}, \{b, d, f\}, \{d, e, f\}, \{a, b, c, f\}\}.$$

R_3 still has a vote assignment. One more transformation will yield the coterie we are after. Using $G = \{b, d, e\}$, $U = \{a, b, c, d, e, f, g\}$, we get

$$R_4 = \{\{a, d, e\}\{c, d\}, \{b, c, e\}, \{a, c, e, f\}, \{b, d, a\}, \{b, d, f\}, \{d, e, f\}, \{a, b, c, f\}, \{a, c, f, g\}, \{b, d, e, g\}\}.$$

If we set up the inequalities corresponding to this coterie we get a contradiction. For instance, since $\{b, c, e\} \in R_4$, $v(b) + v(c) + v(e) \geq \mathrm{MAJ}(v)$. Since $\{b, d, e\} \notin R_4$, then $v(b) + v(d) + v(e) < \mathrm{MAJ}(v)$; so therefore $v(d) < v(c)$. Now, since $\{b, d, f\} \in R_4$, $v(b) + v(d) + v(f) \geq \mathrm{MAJ}(v)$, and we see that $v(c) + v(b) + v(f) \geq \mathrm{MAJ}(v)$. This implies that a is not needed in group $\{a, b, c, f\}$, a contradiction. □

The Venn diagram of Figure 3 summarizes the findings of this section.

One last question arises: How large is the subclass of coteries that have a corresponding vote assignment? To bound the number of different vote assign-

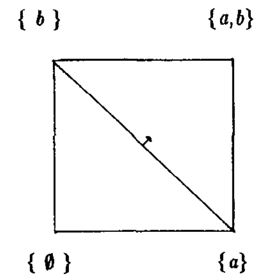

FIG. 4. The two-dimensional hypercube.

ments, we can view the possible subsets of the universe of nodes as the vertices of the n-dimensional unit hypercube. An n dimensional vector represents a hyperplane that cuts the cube in two halves, and each vote assignment corresponds to one such hyperplane. In Figure 4, we illustrate the two-dimensional hypercube and a vector that corresponds to the assignment that gives each of the two nodes a vote. In this case, the hypercube nodes on the right side of the vector (i.e., $\{a, b\}$) represent the groups of nodes that have a majority. The nodes on the left are the groups (e.g., $\{a\}$, $\{b\}$, $\varnothing$) without a majority. Of course, not all of the hyperplanes represent vote assignments (e.g., the hyperplane that leaves all four nodes on one side does not correspond to a vote assignment). If we move a hyperplane as much as possible without crossing any vertex, it will end up sitting on n vertices. We can use this argument to establish the following:

THEOREM 5.3 (M. Yannakakis, personal communication). *There are at most 2^{n^2} different vote assignments.*

PROOF. There are $\binom{2^n}{n} < 2^{n^2}$ choices for the sets of vertices of the hypercube in which to rest hyperplanes, hence the result follows. □

As we can see, vote assignments yield only a very small portion of the ND coteries, so there is a large number of ND coteries without a corresponding vote assignment.

6. *Conclusions*

In this paper we have studied vote assignments and sets of groups (coteries) used for mutual exclusion in a failure-prone distributed system. As we have discovered, the choice of mechanism as well as the selection of votes or coteries very much depends on the special number *five*:

If the system has five or fewer nodes (and we believe this will be by far the most common case), vote assignments and coteries are equivalent (at least if we ignore dominated assignments and coteries). We may wish to think in terms of coteries since the groups that can be formed are explicitly stated, but at implementation time we will probably opt for votes. They take less space to represent and are easier to implement. (Adding votes and checking for a majority is also faster than checking if a group of nodes is in a coterie, but, for five or fewer nodes, the improvement is negligible.)

Also, with five or fewer nodes, the number of choices for ND vote assignments or coteries is surprisingly small (as seen in Figure 2). Thus, the system designers can actually inspect *all* choices and select the one that yields the *best* reliability for the given hardware.

For systems with more than five nodes, the story is very different. In this case, coteries are more powerful, in the sense that there are coteries that cannot be represented by votes. However, from a practical point of view, it may be difficult to take advantage of this. For more than five nodes, the number of coteries is huge. As a matter of fact, just the coteries generated by the partial enumeration algorithm is very large. Thus, we will need heuristics to trim down the number of choices, and considering vote assignments only seems one practical way to do this. (Of course, there are other possible heuristics.) Furthermore, it is very easy to tell whether a vote assignment is ND (Theorem 3.2 and 3.3), but for six or more nodes, just checking if a coterie is ND (Theorem 2.1) can be very time consuming.

ACKNOWLEDGMENTS. A number of useful suggestions were provided by A. Wigderson and M. Yannakakis.

REFERENCES

1. BENZAKEN, C. Critical hypergraphs for the weak chromatic number. *J. Comb. Theory, Series B 29*, (1980), 328–338.
2. BERGE, C. *Graphs and Hypergraphs.* American Elsevier, New York, 1973.
3. DAVIDSON, S. Evaluation of an optimistic protocol for partitioned distributed database systems. Tech. Rep. 299, Dept. of Electrical Engineering and Comput. Sci., Princeton Univ., May 1982.
4. DOLEV, D., AND STRONG, H. R. Polynomial algorithms for multiple processor agreement. In *Proceedings of the 14th ACM Symposium on Theory of Computing* (San Francisco, Calif., May 5–7). ACM, New York, 1982, pp. 401–407.
5. GARCIA-MOLINA, H. Reliability issues for fully replicated distributed databases. *IEEE Trans. Comput. 15*, 9 (Sept. 1982) 34–42.
6. GIFFORD, D. K. Weighted voting for replicated data. In *Proceedings of the 7th Symposium on Operating System Principles* (Pacific Grove, Calif., Dec.). ACM, New York, 1979, pp. 150–162.
7. LAMPORT, L. The implementation of reliable distributed multiprocess systems. *Comput. Networks 2* (1978), 95–114.
8. LAMPORT, L., SHOSTAK, R., AND PEASE, M. The Byzantine Generals problem. *ACM Trans. Prog. Lang. Syst. 4*, 3 (July 1982), 382–401.
9. LOVASZ, L. On chromatic number of finite set systems. *Acta Math. Acad. Sci. Hungary 19* (1968), 59–67.
10. LOVASZ, L. Coverings and colorings of hypergraphs. In *Proceedings of the 4th Southeastern Conference on Combinatorics, Graph Theory and Computing* (Florida Atlantic University). Utilitas Mathematica, Winnipeg, Canada, 1973, pp. 3–12.
11. LYNCH, N. FISCHER, M., AND FOWLER, R. A simple and efficient byzantine generals algorithm. In *Proceedings of the 2nd Symposium on Reliability in Distributed Software and Database Systems* (Univ. of Pittsburgh, July). IEEE, New York, 1982, pp. 46–52.
12. ROTHNIE, J. B., AND GOODMAN, N. A survey of research and development in distributed database management. In *Proceedings of the 3rd International Conference on Very Large Database* (Tokyo, Japan, Oct. 6–8). ACM, New York, 1977, pp. 48–62.
13. SKEEN, D. A quorum-based commit protocol. In *Proceedings of the 6th Berkeley Workshop on Distributed Data Management and Computer Networks* (Asilomar, Calif., Feb.). Lawrence Berkeley Laboratories, Berkeley, Calif., 1982, pp. 69–80.
14. SKEEN, D. Crash recovery in a distributed database system. Ph.D. dissertation, Memorandum UCB/ERL M82/45, Univ. of California at Berkeley, Berkelely, Calif. May 1982.
15. THOMAS, R. H. A majority consensus approach to concurrency control. *ACM Trans. Database Syst. 4*, 2 (June 1979), 180–209.

RECEIVED FEBRUARY 1983; REVISED MAY 1985; ACCEPTED JUNE 1985