アルゴリズム 34－2
（1993．8．19）

拡張された分散 k-相互排除

宮本 英典　角川 裕次　山下 雅史
広島大学工学部

分散相互排除問題とは、複数のプロセスとそれらの間を結ぶ通信リンクからなるシステム(以下分散システム)において、複数のプロセスによって共有される資源に対する排他的なアクセスを保証する問題である。従来の研究では、分散システム内の全てのプロセスにより同一の1個あるいはk個の資源を共有する問題、即ち、資源の共有が一様な問題しか研究されていない。この枠組では、資源を共有しているプロセス集合が資源によって異なるといった非一様な共有構造を持つ問題を解くことができない。本稿では、資源の共有が一様でない、より一般的な分散相互排除問題を提案する。そして、その問題を解決するために、コーラムコンセンサスをを用いた解放を考えるが、コーラム構造が満たすべき条件について議論する。更にある種の共有構造に対するコーラム集合の構成法を提案する。

An Extended Distributed k-Mutual Exclusion

Hidenori Miyamoto　Hirotsugu Kakugawa　Masafumi Yamashita
Faculty of Engineering, Hiroshima University

The distributed mutual exclusion problem is the problem that guarantees an exclusive access to a shared resource in a distributed system consisting of processes and communication links. Many researches on this problem have been done, but the model of sharing resources is uniform in the sense that all processes share the same resource(s). In this paper, we consider more general model of resource sharing such that sets of processes that share resource differ in resources. We discuss conditions which quorum sets must satisfy to solve the problem and propose how to construct quorum sets.

1 はじめに

分散相互排除問題とは、複数のプロセスとそれらの間を結ぶ通信リンクからなるシステム (以下分散システム) において、複数のプロセスによって共有される資源に対する排他的なアクセスを保証する問題である。例えば、分散データベースでは、分散システム上の複数のプロセス上にデータの複製を保持することによって、耐故障性を向上させることが行なわれる。そのような複製データの更新においては、データの一貫性を保つために、2つ以上のプロセスが同時にデータへの書き込みを行なわないことを保証しなければならない。このように、2つ以上のプロセスのどの共有資源への同時アクセスを許さない、言い替えれば、どの時刻においても、高々1つのプロセスにしか共有資源に対するアクセスを許さないことを保証する問題が、分散 1-相互排除問題である。この問題に対して、これまでに多くの研究がなされており、この問題を解くためのアルゴリズムが幾つか提案されている [1][4][7] [8][9]。

それらのアルゴリズムは、大別するとトークン型とコーラムコンセンサス型の2つのタイプに分けることが出来る。

トークン型のアルゴリズムは、システム内に特権トークンを1つ用意しておき、そのトークンを保持しているプロセスのみが共有資源へのアクセスができるとしたものである。トークンは必要に応じてプロセスからプロセスへ渡される。システム内にはどの時点に於いてもトークンは1つだけなので、共有資源をアクセスするプロセスはどの時点に於いても高々1つであることが保障される。この考えに基づいたアルゴリズムに、例えば [4],[9] がある。コーラムコンセンサス型のアルゴリズムでは、コータリーと呼ばれる構造を利用することで共有資源への排他的なアクセスを実現する。コータリーとは、コーラムと呼ばれるプロセス集合の集まりである。資源へアクセスを行なうプロセスは、いずれかのコーラムに属する全てのプロセスより許可を得るようにする。コータリーに属する任意の2つのコーラムは互いに交わりを持つので、どのプロセスも同時には高々1つの許可しか出さない事にすれば、コーラムの全てのプロセスから許可を得ることのできるプロセスは同時には2つ以上存在しえない。よって、排他的なアクセスが保障される。この考えに基づいたアルゴリズムに、例えば [1],[7],[8] がある。

共有される資源が1つでなく、より一般的な k個の同一な資源の共有問題についての研究が近年なされている。Raymond は Ricart と Agrawala の分散相互排除アルゴリズム [1] を拡張することで同時に k個までのプロセスが臨界領域に進むことのできるが $k+1$ 個はできないことを保障するアルゴリズムを提案した [6]。角川らはコータリーを拡張した k-コータリーを提案し、コーラムコンセンサス型のアルゴリズムを提案している [5]。本稿では、k個の共有資源への排他的なアクセスを保障する問題を k-相互排除問題と呼ぶ。

以上で述べた分散相互排除問題での資源を共有のモデルは、システム内の全てのプロセスが1つあるいは k個の資源を一様に共有する、というものである。本稿では更に一般的な共有の構造について考察する。即ち、資源により、それを共有しているプロセスの集合が異なる共有を許す場合を考える。このために、あるプロセスで利用できる資源の集合が、他のプロセスで利用できる資源の集合と異なる場合も生じ得る。このような共有構造のもとで相互排除問題を解くためのコータリーの構造について検討する。特に、共有構造が包含関係の場合のコータリーの構成法を提案する。最後に、その他の共有構造の場合について議論する。

2 定義

本稿で対象とする分散システムのモデルは以下の通りである。

分散システムは、独立に動作可能な複数個のプロセスと、各プロセス間を結ぶ通信リンクからなる。任意の2つのプロセス間の通信は他のプロセスを介さず、一対一の通信が可能である。すなわち、ネットワークトポロジーは完全グラフである。プロセス間の情報のやり取りは、通信リンクを通じてのメッセージ通信のみであり、共有変数は存在しない。

定義 1 (分散相互排除問題)

プロセスの全体集合を Uとする。各資源 $R_i(1 \leq i \leq m)$ に対し、R_iを共有しているプロセス集合を $G_i \subseteq U$とする。分散相互排除問題とは、各時点に

於いて、R_iを使用するプロセスのかずを高々1つに制限する問題である。複数の資源を利用できるプロセスが存在しうるが、その様なプロセスにとって、利用可能な資源は区別できず、同一なものに見えるとする。 □

この問題は、資源によって共有しているプロセスの集合が異なっている。従来の分散 k-相互排除問題は、以下の様に表現される:

- 資源は $R_1, R_2, \ldots, R_k$の k個、
- 各 i に対し、$G_i = U$。但し、Uはプロセスの全体集合である。

これより明らかなように、定義1で示した分散相互排除問題は、従来の分散 1-および分散 k-相互排除問題を一般化したものである。

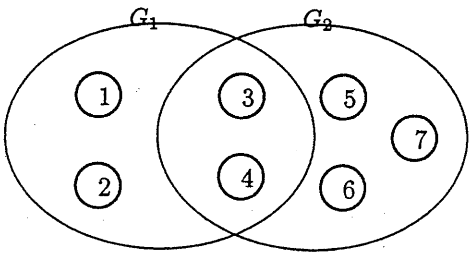

図 1: 共有構造の例

例えば、図1に示した例では、

- U={1,2,...,7},
- 資源 R_1, R_2が存在,
- G_1={1,2,3,4},G_2={3,4,5,6,7}

となっている。プロセス 3,4 は 2 つの資源 R_1とR_2が利用可能であるが、これらの区別は行なわず同一の資源と考える。即ち、資源を獲得するときは2つの内どちらでも利用可能なものを獲得するとする。プロセス1が資源を使用している間はプロセス2は使用できない。しかし、プロセス1と3は同時に使用可能である。また、プロセス3と4が同時に資源を使用している間は、他のプロセスは資源の使用はできない。

3 プロセスの論理構造についての考察

あるプロセス $p \in U$が資源を使用するには、プロセス集合 $Q_p \subseteq U$ に属するすべてのプロセスからの許可が必要であるとする。なお、各 $p \in U$に対し、可能な Q_pは複数個 $(Q_p^1, Q_p^2, \ldots)$ あてもよいとする。各プロセスは同時には高々1つのプロセスにしか許可を与えないものとしたときに、定義1で示された問題を解くための各 Q_p^iが満たす条件は以下のように与えられる。

条件 1: 各 $p \in U$に対し、$\bigcup_i Q_p^i \subseteq \bigcup_j G_j$A, 但し、$p \in G_j$。

これは、各プロセス p の資源獲得に関与するプロセスを、p と資源を共有しているプロセスのみにする条件である。逆を言えば、p と資源を共有していないプロセスに p は関与しない、という条件である。問題は、このような条件の下で定義1を満たすようなコーラム Q_p^iの集合で自然なものが存在するか否かである。一般に、各プロセスは複数個の資源を利用できるが、プロセス p が用いるコーラム Q_p^iに対して実際に割り当てられる資源が静的に定まっていると、Q_p^iを選ぶことは利用する資源を陽に指定することとなり、好ましくない。このようなコーラムの集合は不自然と考える。以下では、共有構造が包含構造となっているときは自然なコーラム集合が存在することが示される。

3.1 共有構造が包含構造の場合

より一般的な共有構造について考える前に、図2のように共有構造が包含構造にある場合について考える。

共有構造が次の条件を満たす時、その構造を包含構造と呼ぶ。

- 資源 $R_i (1 \leq i \leq k)$ に対し $G_i \subseteq U$を R_iを共有するプロセス集合とする時、$G_1 \supseteq G_2 \supseteq \ldots \supseteq G_k$.

図2を考える。図2では2つの資源 R_1, R_2 があり、R_1を共有しているプロセスの一部で R_2を共有している。このような包含構造のとき、各プロセスの用いるコーラムは以下の条件を満たさねばならない。

条件 2:

(a) 各 $p \in G_2$の用いるコーラムの全体集合を $C_2 = \{Q_p^1, Q_p^2, \ldots | p \in G_2\}$ とするとき、C_2には互いに交わりのない 2 つのコーラムの対が存在するが、任意の 3 つのコーラムは必ず交わりを持つ。

(b) 各 $p \in G_1 - G_2$の用いるコーラムの全体集合を $C_1 = \{Q_p^1, Q_p^2, \ldots | p \in G_1 - G_2\}$ とするとき、任意の 2 つの $Q, Q' \in C_1$に対して Q と Q'は必ず交わりを持つ。

(c) 任意の $Q_1 \in C_1$と $Q_2 \cap Q_2' = \emptyset$ であるような $Q_2, Q_2' \in C_2$に対し、$Q_1 \cap (Q_2 \cup Q_2') \neq \emptyset$。即ち、$G_2$で 2 つの資源が用いられている時は、$G_1$のプロセスは資源を利用できない。

以上の 3 つの条件が満たされるコーラム集合の集まりを構成できれば、コーラムコンセンサス型の分散相互排除アルゴリズム (例えば [5], [8]) を用いて、本稿で一般化された分散相互排除を解くことができる。以下では具体的な構成法を考える。上での条件 (b) に於いて、k-コータリーの概念を利用できるので、これを用いることにする。k-コータリーの定義は、以下の通りである [3]。

定義 2 (k-コータリー) 集合 Uの下の k-コータリー C とは、

1. $C \subseteq 2^U, C \neq \emptyset$
2. 各 $Q \in C$に対して、$Q \neq \emptyset$
3. Non-intersection Property
各 $h(< k)$ に対しある $Q_1, Q_2, \ldots, Q_h \in C$が存在し、これらの内の任意の Q_i, Q_jに対し $Q_i \cap Q_j = \emptyset$ であれば、ある $Q_{h+1} \in C$が存在してすべての $i(1 \leq i \leq h)$ に対し $Q_{h+1} \cap Q_i = \emptyset$ かつ $Q_{h+1} \neq Q_i$となる
4. Intersection Property
任意の $Q_1, Q_2, \ldots, Q_k, Q_{k+1} \in C$に対し、ある Q_i, Q_jが存在して $Q_i \cap Q_j \neq \emptyset$
5. Minimality
任意の $Q_1, Q_2 \in C$に対し $Q_1 \not\subseteq Q_2$ □

以下に、共有構造が 2 つの資源の包含構造であるときのコーラム集合の構成法を示す。

構成法 1: プロセス集合 Uの下で 2 つの資源 R_1, R_2がそれぞれ $G_1, G_2 \subseteq U$で共有され、$G_2 \subseteq G_1$であるとする。各プロセス $p \in U$の用いるコーラム集合を以下のように定める:

- 各 $p \in G_2$に対して、G_2の下での任意の 2-コータリーを、C^2とする。各 $p \in G_2$は C^2をコーラム集合に用いる。
- 各 $p \in G_1 - G_2$に対して、$G_1 - G_2$の下での任意の 1-コータリーを C^1 とする。各 $p \in G_1 - G_2$は $C^1 \oplus C^2$をコーラム集合に用いる。但し、$C \oplus C' = \{Q \cup Q' | Q \in C, Q' \in C'\}$ である。
□

構成法 1 で構成されたコーラム集合が先述の条件 (a),(b),(c) を満たすことを以下に示す。

補題 1 $C_i(i = 1,2)$ を U_iの下での k_i-コータリーとし、$U_1 \cap U_2 = \emptyset$ とする。このとき、$k = min\{k_1, k_2\}$ とおけば、$C_1 \oplus C_2$は k-コータリーである。

(証明) 一般性を失うことなく、$k = k_1 \leq k_2$と仮定できる。各 $h \leq k-1$ に対し、$P_1, P_2, \ldots, P_h \in C_1 \oplus C_2$が互いに素とする。$\oplus$の定義より、各 P_iに対し $P_i = A_i \cap B_i, Ai \in C_1, B_i \in C_2$なる A_i, B_iが存在する。C_1は $k_1(= k)$-コータリー,C_2は、$k_2(\geq k)$-コータリーであることにより、ある $A_{h+1} \in C_1$と $B_{h+1} \in C_2$が存在し、$\forall i(1 \leq i \leq h)[A_i \cap A_{h+1} = \emptyset \wedge B_i \cap B_{h+1} = \emptyset]$ となる。故に non-intersection property が成立する。次に intersection property を示す。ある $P_1, P_2, \ldots, P_{k+1} \in C_1 \oplus C_2$が存在して、それらが互いに素と仮定する。$P_i = A_i \cup B_i, A_i \in C_1, B_i \in C_2$なる A_i, B_iが各 p_i に対し一意に定まることにより、$A_1, A_2, \ldots, A_{k+1} \in C_1$が互いに素となる。これは、矛盾。最後に minimality を示す。ある $P_1, P_2 \in C_1 \oplus C_2$が存在して、$P_1 \subseteq P_2$と仮定する。$P_i = A_i \cup B_i (i = 1,2, A_i \in C_1, B_i \in C_2)$ とすると、$A_1 \subseteq A_2$または $B_1 \subseteq B_2$。これは矛盾。 □

補題 2 構成法 1 で構成されたコーラム集合は、条件 2 を満たす。

(証明) コーラム集合の構成で用いられた 1-コータリー, 2-コータリーを各々C^1,C^2とする。

(a) $C_2 = C^2$であり、C^2は 2-コータリーなので、成立。

(b) $C_1 = C^1 \oplus C^2$であり、補題 1 より C_1は 1-コータリーである。

(c) ある $Q_1 \in C^1 \oplus C^2$及び $Q_2, Q'_2 \in C^2$が存在し、$Q_1 \cap (Q_2 \cup Q'_2) = \emptyset$ かつ $Q_2 \cap Q'_2 = \emptyset$ と仮定する。$Q_1 = Q_1^1 \cup Q_1^2, Q_1^1 \in C^1, Q_1^2 \in C^2$とする。$(G_1 - G_2) \cap G_2 = \emptyset$ なので、仮定より $Q_1^2 \cap (Q_2 \cup Q'_2) = \emptyset$。即ち、$Q_1^2, Q_2, Q'_2$は互いに素である。これは、矛盾。

又、条件 1 を満たすことは明らか。 □

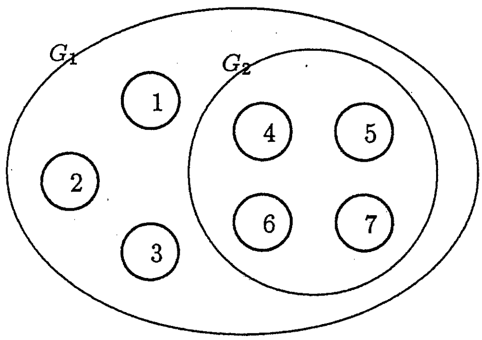

図 2: 包含構造:G_1={1,2,3},G_2={4,5,6,7}

例えば図 2において、$G_1 - G_2$についての 1-コータリー C^1と G_2についての 2-コータリー C^2として、

C^1 = {{1,2},{2,3},{1,3}}
C^2 = {{4,5},{6,7},{4,6},{5,7}}

を用いることにすれば、$C^1 \oplus C^2$は、

$C^{k_1} \oplus C^{k_2}$ =
{{1,2,4,5},{1,2,6,7},{1,2,4,6},{1,2,5,7},
{2,3,4,5},{2,3,6,7},{2,3,4,6},{2,3,5,7},
{1,3,4,5},{1,3,6,7},{1,3,4,6},{1,3,5,7}}

となる。

$G_1 - G_2$に含まれるプロセスはコータリー $C^1 \oplus C^2$を、G_2に含まれるプロセスはコータリー C^2を各々のコーラム集合とする。

構成法 1 では、資源が 2 つのみであったが、包含構成になっている任意の数の資源の場合の構成方法に拡張できる。

構成法 2: プロセス集合 Uの下で k個の資源 $R_1, R_2, \ldots, R_k$がそれぞれ $G_1, G_2, \ldots, G_k \subseteq U$によって共有されており、$G_k \subseteq G_{k-1} \subseteq \ldots \subseteq G_1$とする。各プロセス $p \in U$の用いるコーラム集合を、以下のように定める:

- $p \in G_k$に対して、G_kの下での任意の k-コータリーを C^kとする。各 $p \in G_k$は C^kをコーラム集合に用いる。
- $p \in G_L$かつ $p \notin G_{L+1} (1 \leq L < k)$ に対して、$G = G_L - G_{L+1}$ とおく。G の下での L-コータリーを C^Lとする。各 $p \in G$ は、$C^L \oplus C^{L+1}$をコーラム集合に用いる。 □

構成法 2 で構成されたコーラム集合が定義 1 での相互排除を保障するためには、(条件 2 を拡張した) 以下の条件が満たされないといけない。
条件 2':

(a) 各 $p \in G_k$の用いるコーラムの全体集合は G_kの下での k-コータリーである。

(b) 各 $p \in G_L (p \notin G_{L+1}, 1 \leq L < k)$ の用いるコーラムの全体集合は $G_L - G_{L+1}$の下での L-コータリーである。

(c) 各 $l (1 \leq l \leq k), L$ に対し、$C^k, C^k \oplus C^{k-1}, C^k \oplus C^{k-1} \oplus \cdots \oplus C^{L+1}$のいずれかに含まれる任意の $Q_1, Q_2, \ldots, Q_l$と任意の $Q_1^L, Q_2^L, \ldots, Q_{k-l+1}^L \in C^L$ に対し、
$(\bigcup_{i=1}^{l} Q_l) \cap (\bigcup_{i=1}^{k-l+1} Q_i^L) \neq \emptyset$。

補題 3 構成法 2 で構成されたコーラム集合は条件 2' を満たす。

(証明) コーラム集合の構成で用いられた G_Lに対するコータリーを C^Lとする。

(a) 各 $p \in G_k$の用いるコーラム集合 C^kは kコータリーなので成立。

(b) 各 $p \in G_L(p \notin G_L, 1 \leq < k)$ の用いるコーラム集合 $C_L \oplus C_{L+1}$は補題1より L-コータリーであるので成立。

(c) ある $l(1 \leq l \leq k), L(1 \leq L < K)$ と、C^k,$C^k \oplus C^{k-1}$, $C^k \oplus C^{k-1} \oplus \cdots \oplus C^{L+1}$のいずれかに含まれる $Q_1, Q_2, \ldots, Q_l$ と $Q_1^L, Q_2^L, \ldots, Q_{k-l+1}^L \in C^L \oplus C^{L+1} \oplus \cdots \oplus C^k$が存在し、$(\bigcup_{i=1}^{l} Q_i) \cap (\bigcup_{i=1}^{k-l+1} Q_i^L) = \emptyset$ と仮定する。任意の i について $Q_i^L = A_i \cup B_i, A_i \in C^L, B_i \in C^k \oplus C^{k-1} \oplus \cdots \oplus C^{L+1}$とする。包含構造の定義と仮定より、$(\bigcup_{i=1}^{l} Q_i) \cap (\bigcup_{i=1}^{k-l+1} B_i) = \emptyset$ となり、$k+1$個の互いに素なコーラム集合が存在する。これは、矛盾。 □

3.2 その他の共有構造の場合

ここでは、図3で示されるような共有構造について考える。3つの資源 R_1, R_2, R_3 が与えられ、その共有構造は、$G_1 \supset G_2, G_1 \supset G_3$ で与えられるものとする。

補題1の構成方法にならって、G_2, G_3それぞれの下での2-コータリー C_2, C_3を構成し、また $G_1 - G_2 - G_3$の下での1-コータリー C_1を構成する。G_2, G_3のプロセスはコーラム集合としてそれぞれ C_2, C_3を用いる。また、$G_1 - G_2 - G_3$に含まれるプロセスはコーラム集合として $C_1 \oplus C_2 \oplus C_3$を用いる。しかし、この構成方法では、図3の場合システム内に3つの資源があるにもかかわらず、高々2つの独立なコーラムしか取り得ない。これは、G_2と G_3にとって、それぞれ互いに共有していない資源が1つあるが、前記のコータリー構成方法によるコータリーでは、G_2のプロセスがコーラムを取ると同時に G_3の C_3のコーラムを取ること (または、その逆) になってしまうためである。このことから考えると、補題1のコータリーの構成方法の応用は、システム内のすべての資源を利用できるプロセスが存在する場合、すなわち星型構造 (図4に含まれる共有構造にある場合のみと予想される。また、図3の場合のコータリーの構成法として、次のようなものが考えられる。
構成法 3:

- i=2,3 について、各 G_iの下での任意の 1-コータリーを C_iとする。各 $p \in G_i$はコーラム集合として、C_i をコーラム集合として用いる。
- $p \in G_1 - G_2 - G_3$に対して、$G_1 - G_2 - G_3$ の下での任意の 1-コータリーを C_1とし、各 $p \in G_1 - G_2 - G_3$は、$C_1 \oplus C_2 \oplus C_3$をコーラム集合として用いる。 □

しかし、構成法3では、G_2, G_3のプロセスは選ぶコーラムによって、どの資源を要求するかが静的に定まることになり、これはどの資源を要求するかを陽にしていすることになるので、不自然である。図3の場合に対するより自然なコーラムの構成法が存在するか否か、また存在するならばそれはどのようなものかについては今後の課題として残る。

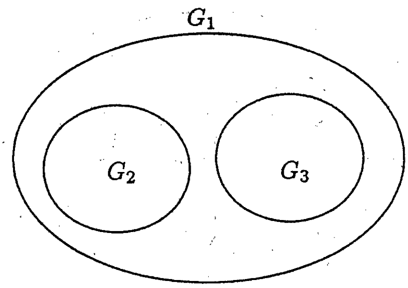

図 3:

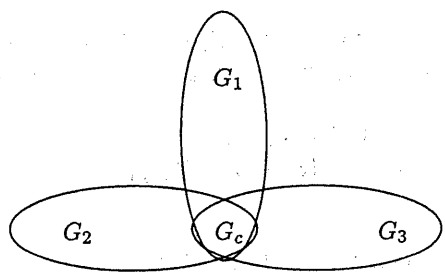

図 4:

4 おわりに

本稿で我々は、これまで多く研究されてきた分散1-, あるいは k-相互排除問題のより一般的な問題である拡張された分散 k-相互排除問題を定義した。さらに、その問題に対し、コーラムコンセンサス型の分散相互排除アルゴリズムを設計するための準備として、共有構造が包含構造である場合のコータリーの構成方法の1つを示した。今後の研究課題として、より一般的な共有構造に対するコータリーの構成方法、および、そのコータリーを用いた分散相互排除アルゴリズムの開発が挙げられる。

参考文献

[1] G. Ricart, A. K. Agrawala, "An Optimal Algorithm for Mutual Exclusion in Computer Networks," *Communications of the ACM*,Vol.24,No.1,pp.9-17,Jan.1981.

[2] H. Garcia-Morina, D. Barbara, "How to Assign Votes in a Distributed System," *Journal of the Association for Computing Machinery*, Vol.32, No.4, pp.841-860, Oct.1985.

[3] H. Kakugawa, S. Fujita, M. Yamashita, T. Ae, "Availability of k-Coterie," *IEEE Transactions on Computers* (to appear).

[4] I. Suzuki, T. Kasami, "A Distributed Mutual Exclusion Algorithms." *ACM Transactions on Computer Systems*,Vol.3,No.4,pp.344-349,Nov.1985.

[5] 角川裕次,藤田聡,山下雅史,阿江忠, "分散 k-相互排除のプロトコル," 信学技報 COMP91-28, 電子情報通信学会,6 1991.

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

[7] L. Lamport, "Time, clocks, and the ordering of events in a distributed system," *Communications of the ACM*, Vol.21, No.7, pp.558-565, Jul. 1978.

[8] M. Maekawa, "A $\sqrt{N}$ Algorithm for Mutual Exclusion in Decentralized Systems," *ACM Transactions on Computer Systems*, Vol.3, No.2, pp.145-159, May 1985.

[9] M. Mizuno, M. L. Neilsen, R. Rao, "A Token Based Distributed Mutual Exclusion Algorithm based on Quorum Agreements," Proc. of 11th International Conference on Distributed Computing Systems, pp.361-368, May 1991.