情報処理学会論文誌 Vol. 49 No. 6 2081–2091 (June 2008)

非連携チェックポインティングにおいて同時多重メディア障害を考慮したチェックポイント・ミラーリング手法

大原 衛[†1,†2] 上杉賢弘[†3]
新井雅之[†2] 福本聡[†2]

従来のチェックポインティング技術の多くは，ストレージメディアでの障害（メディア障害）の発生を想定していないため，チェックポイントデータを保存するストレージにメディア障害が発生した場合，システムは障害回復を行えない．本論文では，非連携チェックポインティングにおけるチェックポイント・ミラーリング手法を提案する．各プロセスはチェックポイントデータを自ノードに加えて他ノードにも保存する．メディア障害が発生した際は，他ノードからこれらのデータを取得することで障害回復を行う．非連携チェックポインティングでは，各プロセスが複数世代のチェックポイントデータを保持し，本質的に冗長性を備えているといえる．本研究は，この冗長性を利用して，比較的低いストレージオーバヘッドで高い耐メディア障害性を得ることを目的とする．本論文では，複製を配置するノードの選択方法によって，冗長度を増すことなく耐メディア障害性を向上させることができることを示す．プロセスごとに複製を配置するノードを固定する固定ミラーリング手法（FM）と，チェックポインティングのたびにこれを変更するローテーショナル・ミラーリング（RM）手法について，シミュレーションによって同時多重メディア障害発生時における回復能力の評価を行った．数値例から，ドミノ効果の発生頻度が比較的低い場合に，RM 手法はより高い冗長度の FM 法と同等の回復能力を有し，耐メディア障害性を向上させることができることが示される．

Checkpoint-mirroring Techniques for Uncoordinated Checkpointing toward Tolerating Concurrent Multiple Media Failures

Mamoru Ohara,[†1,†2] Takahiro Uesugi,[†3]
Masayuki Arai[†2] and Satoshi Fukumoto[†2]

Most conventional distributded checkpointing techniques assume the storage media to be stable. A system cannot recover from failures when some failures occur in the storage media because the checkpoint data are lost. In this paper, we propose checkpoint-mirroring techniques for uncoordinated checkpointing, in which each process saves its checkpoint data in its own node as well as in some of the other nodes. When a media failure occurs, the failure will be recoverable by loading these copies of checkpoint data. In uncoordinated checkpointing, processes usually hold a number of their checkpoints, that means, uncoordinated checkpointing techniques naturally have some redundant checkpoints. We aim to improve the media-failure-tolerance of uncoordinated checkpointing by utilizing the redundancy with adding a little extra storage overheads. In this paper, we show that we can improve the media-failure-tolerance by properly selecting nodes holding the copies of checkpoint data. We performed simulations to evaluate the recoverability of fixed mirroring (FM) technique and rotational mirroring (RM) technique which changes the mirror nodes at every checkpointing. The simulation results show the RM technique can effectively reduce mean rollback distances on concurrent multiple media failures.

1. まえがき

分散チェックポインティングは，分散システムにおける代表的なフォールトトレランス技術の 1 つである[1)–5)]．システムの大域的な状態を定期的に不揮発性のストレージメディアに保存し，障害の発生に備える．システムの大域状態は，システム中の各プロセスがそれぞれ局所的なチェックポイント（ローカルチェックポイント）を設定することで保存される．各プロセスの局所状態はプロセス間で交換されるメッセージによって相互に依存性を持つため，異なるプロセスのローカルチェックポイントどうしにも依存性が生じる．分散チェックポインティング手法は，この依存性に関して無矛盾な大域状態を取得する方法によって，連携手法と非連携手法に大別される[6)]．

連携手法は，チェックポインティング時にプロセスが協調して無矛盾な大域状態を保存する[1),2)]．このため，リカバリの基点が決定論的に定まるという利点があり，障害からの高速な回復が期待できる．また，各プロセスはそれぞれ 1 つずつのローカルチェックポイントを保持すればよく，空間オーバヘッドが小さい．しかし，プロセス間協調のオーバヘッドが大きいため，大規模分散システムにおいてはその適用が難しい場合がある．

†1 東京都立産業技術研究センター
Tokyo Metropolitan Industrial Technology Research Institute
†2 首都大学東京システムデザイン学部
Faculty of System Design, Tokyo Metropolitan University
†3 首都大学東京大学院システムデザイン研究科
Graduate School of System Design, Tokyo Metropolitan University

非連携手法は，各プロセスが協調しあうことなく独自に複数世代のローカルチェックポイントを設定し，リカバリ時に無矛盾な大域状態を形成できるチェックポイントの組合せを探索することで，通常実行時のチェックポインティングオーバヘッドを小さく抑える．通常時オーバヘッドを極力抑える必要のある並列シミュレーションなどの科学技術計算，分散データベースの排他制御などが，非連携手法の主な応用例として報告されている[7)–11)]．

従来研究の多くは，ストレージメディアに障害が発生する可能性を考慮していない．チェックポイントデータが保存されたメディアに障害が発生した場合，チェックポインティング・リスタートによる障害からの回復は不可能である．Plank は，このようなメディア障害に対する耐性を得る手法として，連携チェックポインティングにミラーリングの概念を導入する MIR 手法を提案した[12)]．また，Nakamura らは MIR 手法におけるオーバヘッドを改善した Skewed Checkpointing（SC）手法を提案した[13),14)]．しかし，非連携手法へ耐メディア障害性を付加する研究についての報告は見あたらない．

連携手法では，通常，各プロセスは最新のローカルチェックポイントだけを保持する．MIR，SC 手法は，ともに連携手法に空間オーバヘッドを追加して，メディア障害への耐性を付加する．これらの手法では，同時多重メディア障害への耐性を向上させることは，空間オーバヘッドの増加とトレードオフの関係にある．一方，非連携手法においては，各プロセスが複数世代のチェックポイントデータを保持するのが通常である．連携手法に比べて，チェックポイントデータの冗長性が高いといえる．非連携手法が元来有する冗長性を利用することで，空間オーバヘッドを大きく増加させることなく，チェックポイント・ミラーリングの手法を導入できる可能性がある．

このようなチェックポイント・ミラーリング手法を構築するための第 1 段階として，本研究では，非連携手法におけるチェックポイント・ミラーリング手法を提案する．本論文では，チェックポイントデータの複製を保持するノードを選択するアルゴリズムを検討する．このためのアルゴリズムとして，複製を各プロセスごとに定められた固定のノードに保存する固定ミラーリング（FM）法と，複数のノードに分散して保存するローテーショナル・ミラーリング（RM）法を提案する．RM 法は，各プロセスがチェックポインティングのたびに複製を保存するノードを変更するという点で SC 法と類似のアイディアを用い，複製の配置が非連携手法に適するように修正を加えている．

RM 法は，チェックポインティング時は SC 法と似た挙動を示すが，両者のリカバリ時の振舞いは大きく異なる．これまでに，非連携手法のためのチェックポイント・ミラーリングのリカバリに関する議論はあまり行われていない．本論文では，提案手法のリカバラビリティ（回復能力）を定量的に評価する．リカバラビリティを計る尺度としては，平均ロールバック間隔を用いる．非連携手法において無矛盾なチェックポイント集合の探索に要するオーバヘッドやこれを取得できる確率は，システム中のプロセス数，メッセージ頻度などの影響を受け，解析的な評価は簡単ではない．このため，本研究では，メディア障害が発生した際の平均ロールバック間隔をシミュレーションによって評価した．数値例から，ドミノ効果が発生しにくい条件下では，RM 法が FM 法よりも高いリカバラビリティを持つことが示された．

本論文の構成は以下のとおりである．2 章では関連する研究について述べる．3 章は提案手法について述べ，2 種の複製ノードの選択アルゴリズムについて論じる．提案手法の多重メディア障害に対する回復能力を評価するためにシミュレーションを行った．4 章は，シミュレーションから得られた数値例を示す．5 章は本論文のまとめを与える．

2. 関 連 研 究

分散システムのチェックポインティングでは，個々のプロセスの局所状態の集合としてシステム全体の大域状態が形成される．各プロセスは，その局所状態をローカルチェックポイントとして保存する．システム内の全プロセスについて，各プロセスから 1 つずつのローカルチェックポイントを集めた集合をグローバルチェックポイントと呼ぶ．各プロセスの局所状態はメッセージ交換を通じて相互に依存性を持つため，グローバルチェックポイントは，この依存性に関して矛盾を含む場合がある．

図 1 に，矛盾を含むグローバルチェックポイントと無矛盾なグローバルチェックポイントの例を示す．グローバルチェックポイントの無矛盾性は，これに含まれるチェックポイントとメッセージ交換の時間的関係によって決定される[15),16)]．チェックポイント $C_{0,3}$，$C_{1,2}$，および $C_{2,3}$ からなるグローバルチェックポイントは，プロセス P_2 がメッセージ $m_{1,2}$ を受信した後の状態であるにもかかわらず，P_1 はこれを送信する前の状態にあるため，矛盾を含む．一方，$\{C_{0,2}, C_{1,1}, C_{2,1}\}$ からなる大域状態は無矛盾である．

分散チェックポインティングは，グローバルチェックポイントの無矛盾性の確保の方法によって，連携手法と非連携手法に大別される[6)]．連携手法では，各プロセスにおいてローカルチェックポインティングを行う時機をプロセス間の協調によって調整し，同期的にローカルチェックポイントを設定する[1),2),10)]．このため，同世代のローカルチェックポイントの集合からなるグローバルチェックポイントは，つねに無矛盾である．協調動作を必要とするため，チェックポインティングオーバヘッドは大きいが，リカバリ動作は単純でそのための

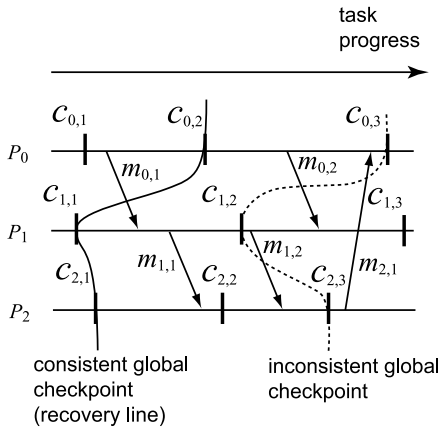

図 1　無矛盾なグローバルチェックポイントと矛盾を含むグローバルチェックポイント
Fig. 1　Consistent and inconsistent global checkpoints.

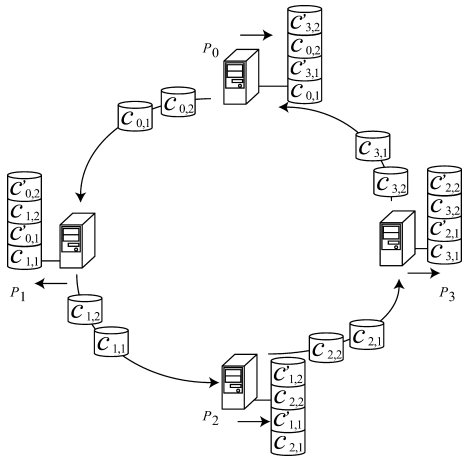

図 2　MIR 手法の動作
Fig. 2　An example illustration for MIR behavior.

オーバヘッドは小さい．

非連携手法では，各プロセスが独立してローカルチェックポイントを取得する．これらのチェックポイントの任意の集合は，必ずしも無矛盾ではない．各プロセスはローカルチェックポイントを複数世代保持しておき，リカバリ時にこれらの中から無矛盾な集合を探索する[17),18)]．チェックポインティング時に協調動作を必要としないため通常時のオーバヘッドは小さい．しかし，無矛盾な大域状態を探索するため，連鎖的にロールバックを繰り返す可能性がある．この連鎖的ロールバックは，ドミノ効果と呼ばれる[6),19)]．たとえば，図 1 の例では，最新のチェックポイント集合は矛盾を含むため，各プロセスは順次ロールバックを繰り返し，最終的に $\{C_{0,2}, C_{1,1}, C_{2,1}\}$ で無矛盾なグローバルチェックポイントを発見する．このような無矛盾なグローバルチェックポイントは，再実行の起点となるため，リカバリラインと呼ばれる．

従来のチェックポインティング手法の多くは，チェックポイントデータを保存するストレージメディアを故障することのない安定記憶と仮定する．このような手法では，チェックポイントデータが保存されているメディアに障害が生じると，チェックポインティング・リスタートによる障害からの回復は不可能になる．本論文では，このようなストレージメディアの障害をメディア障害と呼ぶ．これまでに，メディア障害に耐性を持つ連携チェックポインティング手法がいくつか提案されている．

チェックポイント・ミラーリングは，連携チェックポインティングに RAID アレイで用いられるミラーリングの手法[20)]を導入して，耐メディア障害性を付加する手法である．この手法では，各ノードが，チェックポインティング時にチェックポイントデータの複製を他の m 個のノードに保存する．本論文では，m を冗長度と呼ぶ．**図 2** は，Plank が提案したチェックポイント・ミラーリング手法である MIR 手法の動作例である[12)]．図で $C_{i,j}$ は，プロセス P_i の j 番目のチェックポイント，$C'_{i,j}$ は $C_{i,j}$ の複製を表す．冗長度 m を大きくするほど同時多重メディア障害への耐性を向上させることができるが，空間オーバヘッドが増大する．

Nakamura らによって，$m = 1$ とした MIR 手法の複製配置先を工夫した Skewed Checkpointing が提案されている[13)]．この手法では**図 3** のように，チェックポインティングのたびに複製の保存先を変更することで，複数世代のチェックポイントデータをいくつかのノードに分散して配置する．この手法では，$n = \lfloor \log_2 N \rfloor$ 重メディア障害が発生した際も，直前 n 世代以内のグローバルチェックポイントが少なくとも 1 つ残存する．ただし，N はシステム中のプロセス数である．各プロセスが保持するチェックポイントデータ量は $(n+1)$ 世代分であり，連携手法としては空間オーバヘッドが大きい．

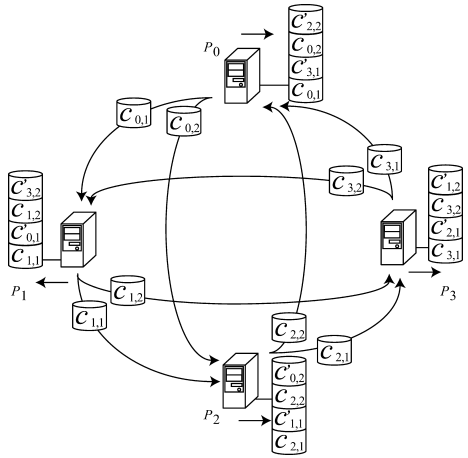

図 3　Skewed Checkpointing の動作
Fig. 3　An example illustration for the behavior of Skewed Checkpointing.

3. 提 案 手 法

3.1 システムモデル

本論文では，相互に双方向接続された N 個のノード $P_0, P_1, \ldots, P_{N-1}$ からなる分散システム D について論じる．各ノードは，それぞれ個別の不揮発性ストレージメディアを有する．プロセスは障害の発生に備えて，その内部状態を，自身を実行するノードのストレージメディアに保存する．本論文では簡単のため，各ノードは 1 つのプロセスだけを実行するものとし，以後，ノードとこれが実行するプロセスとを区別せずに P_i（$i = 0, 1, \ldots, N-1$）と呼ぶ．

本研究では，メディア障害とプロセス障害の 2 種類の障害を考慮する．メディア障害は，チェックポイントデータが保存されるストレージメディアに発生する障害を指す．ノード P_i にメディア障害が発生した場合，プロセス P_i は，ストレージメディアを利用したチェックポイントデータの読み出しおよび書き込みができなくなる．

これに対して，プロセスが何らかの原因によって実行を継続することができなくなった状態をプロセス障害と呼ぶ．プロセス障害の原因として，たとえばメモリなどのハードウェアの過渡的な故障に起因するものや，並列シミュレーションや分散データベースにおける整合性の喪失などの論理的要因が考えられる[7)–9),11),21)]．プロセスを実行するノードは，プロセス障害を検出すると，当該プロセスの回復処理を行う．本論文では，プロセス障害はチェックポインティング・リスタートによって回復されるが，メディア障害は永続的な障害であり，チェックポインティング手法による回復の対象とならないものと仮定する．メディア障害の回復には，ストレージメディアの交換が必要である．

3.2 チェックポイント・ミラーリング

本研究では，非連携チェックポインティングにメディア障害への耐性を与えるチェックポイント・ミラーリング手法を提案する．提案手法では，正常動作時は各プロセスが独自にチェックポインティングを行う．各プロセスは，チェックポイントデータを自ノードのストレージメディアに保存し，同時にその複製を一定のアルゴリズムに従って他の m（$0 \leq m < N$）個のノードに配置する．プロセス P_i の j 番目のチェックポイント $C_{i,j}$ の複製を保持するノードを $C_{i,j}$ のミラーノードと呼び，ミラーノードの集合 $R_{i,j}$ を $C_{i,j}$ のミラーノード集合と呼ぶ．本論文では，簡単のため，プロセスがチェックポイントを生成してから，この複製をすべてのミラーノードにコピーし終わるまでの時間は障害の発生時間間隔と比べて十分に短く，この間に障害は発生しないものとする．

本論文では，ミラーノードの選択アルゴリズムとして，以下の 2 つを検討する．

- 固定ミラーリング（FM: Fixed Mirroring）法
  FM 法は，単純なミラーノード選択方法である．各ノードが，つねにあらかじめ定められた m（$< N$）個のノードに，そのすべてのチェックポイントデータをコピーする．ノード間の負荷の分散を図るため，各ノードがそれぞれ他の m 個のノードのチェックポイントデータをミラーリングする．ノードの故障率に空間的偏りがないと仮定すると[⋆1]，以下のようにミラーノードを選択すればよい．すなわち，
  $$R_{i,j} = \{P_k \mid k = (i+1) \bmod N, (i+2) \bmod N, \ldots, (i+m) \bmod N\} \quad (1)$$
  である．ただし，FM 法におけるミラーノードは，チェックポイント世代 j によらず，プロセスごとに一定であるため，以後 FM 法に限って，プロセス P_i のミラーノード集合を単に R_i と記述する．
- ローテーショナル・ミラーリング（RM: Rotational Mirroring）法
  RM 法は，Skewed Checkpointing（SC）法[13)] と類似の，チェックポインティングごとにミラーノードを変更する複製配置手法を非連携チェックポインティングに導入する．

⋆1 故障率に空間的な偏りがある場合の解析は文献 14) などに見られる．

SC 法は冗長度 m を 1 としたチェックポイント・ミラーリング手法であり，プロセス P_i は j 番目のチェックポイント $C_{i,j}$ の複製を $P_{\{i+2^{(j-1) \bmod n}\} \bmod N}$ に保存する．ただし，$n = \lfloor \log_2 N \rfloor$ である．各プロセスは n 個のミラーノードを，チェックポインティングごとに順次変更して複製を行う．

これに対して，RM 法では，各プロセスはミラーノード集合を以下のように変更する．すなわち，

$$a_{i,j} = [i + \{j \bmod (N-1)\} + 1] \bmod N \tag{2}$$

を用いて

$$d_{i,j} = (i - a_{i,j}) \bmod N \tag{3}$$

とすれば，

$$R_{i,j} = \{P_{a_{i,j}}, P_{(a_{i,j}+1) \bmod N}, \ldots, P_{(a_{i,j}+m-1) \bmod N}\} \ (m \leq d_{i,j}) \tag{4}$$

または，

$$\begin{aligned} R_{i,j} = \{&P_{a_{i,j}}, P_{(a_{i,j}+1) \bmod N}, \ldots, \\ &P_{(a_{i,j}+d_{i,j}-1) \bmod N} \ (= P_{(i-1) \bmod N}), \\ &P_{(a_{i,j}+d_{i,j}+1) \bmod N} \ (= P_{(i+1) \bmod N}), \\ &\ldots, P_{(a_{i,j}+m) \bmod N}\} \quad (m > d_{i,j}) \end{aligned} \tag{5}$$

である．

図 4 は，$m = 1$ とした RM 法のチェックポインティング時の動作例である．横軸は時間の経過を，2 つのプロセスを結ぶ矢印はメッセージを表す．非連携手法では各プロセスが独自にチェックポイントを取得するため，ある時点でのチェックポイント保持数はプロセスによって異なる場合がある．図で，プロセス P_0 は，まずローカルチェックポイント $C_{0,1}$ を生成し，そのデータを $R_{0,1}$（$= \{P_2\}$）にコピーする．次いで，P_0 は，$C_{0,2}$ を P_3 に，$C_{0,3}$ を P_1 にコピーする．図の明瞭さのために省略されているが，この後，P_0 は $C_{0,4}$ を P_2 に，$C_{0,5}$ を P_3 というように，チェックポイントデータの複製を自身を除く他のすべてのノードに順番に保存する．また，他のプロセスも P_0 と同様に，ミラーノード集合をチェックポインティングごとに変更しながら複製を配置する．このように，チェックポインティングのたびに，あるプロセスのミラーノード集合が，他のすべてのノードを結んでできた円上を回るように変更される様子から，本論文ではこの手法をローテーショナル・ミラーリングと呼ぶ．

RM 法と SC 法のミラーノード選択法の差異は，連携および非連携チェックポイントの特性の相違を反映している．連携手法では，グローバルチェックポイントをすべて同世代のローカルチェックポイントから形成する必要がある．冗長度 $m = 1$ である SC 法では，プロセス P_i とそのチェックポイント $C_{i,j}$ のミラーノードに同時にメディア障害が発生した場合，他のプロセスのメディアが残存していても第 j 世代のグローバルチェックポイントが失われる．同時多重メディア障害の発生に際してグローバルチェックポイントを利用できるようにするために，SC 手法では複数世代のグローバルチェックポイントをそれぞれ異なる組合せのノードに複製させる．また，2 つのノード P_α と P_β について，P_β が $C_{\alpha,j}$ の複製を保持し，P_α が $C_{\beta,k}$（$j \neq k$）の複製を保持するような複製配置法では，これらのノードに同時にメディア障害が発生した際に，第 j，第 k 世代の両方のグローバルチェックポイントが失われる．ノード間にこのような相互関係を生じさせないために，SC 手法では，各ノードはそれぞれ $n = \lfloor \log_2 N \rfloor$ 個のノードだけをミラーノードとして利用する．

一方，非連携チェックポインティングでは，グローバルチェックポイントを形成するロー

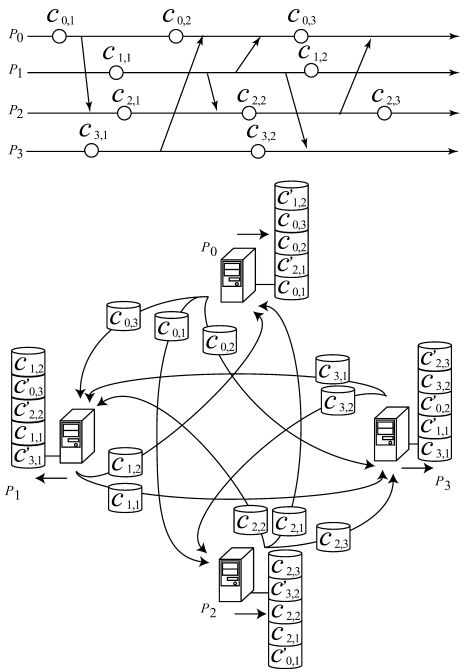

図 4　冗長度 $m = 1$ の RM 手法の動作例
Fig. 4　An example of the checkpoint-mirroring behavior of RM method ($m = 1$).

カルチェックポイントの世代に関する制約はなく，すべてのプロセスのローカルチェックポイントを 1 つずつ含めばよい．しかし，非連携手法のグローバルチェックポイントは必ずしも無矛盾でないため，リカバリラインが存在する確率を高めるためには，メディア障害発生時にできるだけ多数のグローバルチェックポイントを残存させる必要がある．このため，RM 法では，各プロセスが他のすべてのプロセスにいずれかの世代のチェックポイントデータの複製を保持させる．RM 法では，1 つでもメディアが稼働していれば，少なくとも 1 つ以上のグローバルチェックポイントが残存する．しかし，これらのうちにリカバリラインが存在するか否かは確率的な問題である．

3.3　リカバリ動作

提案手法におけるプロセス障害からの回復動作は，以下のように行われる．ロールバックするプロセスは，自ノードにメディア障害が発生していない場合，通常の非連携手法と同様にしてロールバックを行う[7),17),22)]．メディア障害が発生している P_i にロールバックの必要が生じた場合，P_i は以下のアルゴリズムを実行する．プロセス P_i が l_i 個のチェックポイントを取得しているとすると，

(1)　for $j = l_i$ to 1
　(a)　ミラーノード集合 $R_{i,j}$ に含まれるノードの 1 つから $C_{i,j}$ の複製の取得を試みる．
　(b)　取得できた場合，$C_{i,j}$ へロールバックし，アルゴリズムを終了する．
(2)　$C_{i,j}$（$j \in [1, l_i]$）のいずれも取得できなかった場合，初期状態から再実行を行う．

P_i のロールバックは，他のプロセスにおいて連鎖的にロールバックを引き起こす場合がある[7)]．

図 5 は，RM 法を用いたチェックポイント・ミラーリングにおけるリカバリ動作の例である．ノード P_0 と P_1 でメディア障害が発生した後，P_1 でプロセス障害が発生し，システムがロールバックを開始したとする．P_1 が保持していたチェックポイントデータはメディア障害で失われているため，P_1 は最新のチェックポイントデータである $C_{1,2}$ の複製を P_0 から取得しようと試みる．しかし，P_0 でもメディア障害が発生しているため，P_1 は P_3 から 1 世代前のミラーチェックポイントデータである $C_{1,1}$ を取得する．ここで，チェックポイント $C_{1,1}$ と P_0，P_2 および P_3 の現在の状態からなる大域状態は矛盾を含む．この矛盾を解消するため，P_1 以外のノードもロールバックを行い，リカバリラインの探索を行う．この例では，最終的に $\{C_{0,2}, C_{1,1}, C_{2,1}, C_{3,1}\}$ をリカバリラインとして大域状態を回復する．

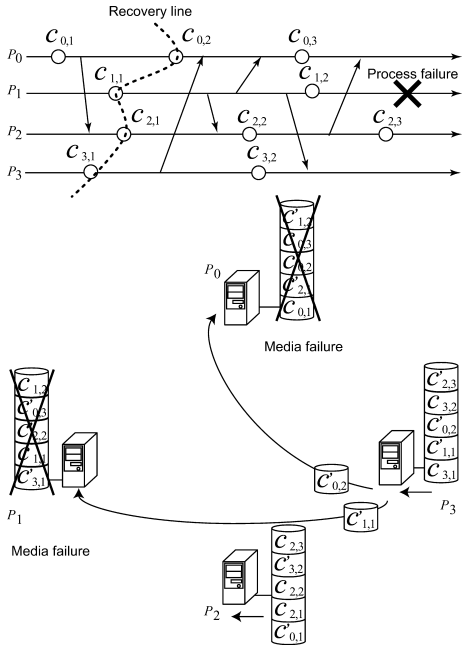

図 5　ローテーショナル・ワンミラー手法の回復時の動作
Fig. 5　An example of the recovery procedure of RM method ($m = 1$).

3.4　同時多重メディア障害への耐性

SC 手法などの連携手法のためのチェックポイント・ミラーリングでは，冗長度と障害の多重度から，リカバリの規模を決定論的に定めることができた[12),13)]．一方，提案手法は非連携チェックポインティングをその適用対象とするため，リカバリ動作は非決定的である．本節では，提案手法の同時多重メディア障害への耐性について議論し，いくつかの定理を示す．

プロセス $P_0, P_1, \ldots, P_{N-1}$ がそれぞれ $l_0, l_1, \ldots, l_{N-1}$ 個のローカルチェックポイントを取得しているとすると，全ローカルチェックポイントの集合は

$$L = \left\{ C_{0,1}, C_{0,2}, \ldots, C_{0,l_0}, C_{1,1}, \ldots, C_{1,l_1}, \ldots, C_{N-1,1}, \ldots, C_{N-1,l_{N-1}} \right\} \tag{6}$$

である．L から得られるグローバルチェックポイントの集合を $G(L)$ とすると，

$$G(L) = \bigcup_{j_0,\ldots,j_{N-1}} \{C_{0,j_0}, C_{1,j_1}, \ldots, C_{N-1,j_{N-1}}\} \quad (j_i \in [1, l_i], i \in [0, N-1]) \tag{7}$$

である．

補題 1　2 つのローカルチェックポイントの集合 L_1，L_2 が $L_1 = L_2$ を満たせば，これらのローカルチェックポイント集合から得られるグローバルチェックポイントの集合 $G(L_1)$ と $G(L_2)$ は等しい．（証明省略）

メディア障害が生じているノードの集合を F としたとき，FM 法および RM 法は，$|F| \leq m$ である任意の F に対して，その影響を受けずにロールバックができる．

定理 1　メディア障害が発生していないときのグローバルチェックポイント集合を $G_\emptyset$ とし，障害ノードの集合 F に対して残存するグローバルチェックポイントの集合を G_F とすると，冗長度 m の FM 法によるチェックポイント・ミラーリングでは，$|F| \leq m$ であれば $G_F = G_\emptyset$ である．

証明　F に含まれないノードの集合 $\overline{F}$ に保存されたローカルチェックポイントの集合を L_F とする．任意のプロセス P_i およびそのミラーノード集合 R_i について，$|P_i \cup R_i| = m+1$ かつ $|\overline{F}| \geq N - m$ から，明らかに $|P_i \cup R_i| + |\overline{F}| > N$ である．よって，$P_i \cup R_i \cap \overline{F} \neq \emptyset$ である．FM 法では，P_i および R_i に含まれるノードはすべて，$C_{i,1}, C_{i,2}, \ldots, C_{i,l_i}$ のデータを保持しているから，$C_{i,1}, \ldots, C_{i,l_i} \in L_F$ である．システムの対称性から，$L_F = \bigcup_{i=0}^{N-1} \{C_{i,1}, C_{i,2}, \ldots, C_{i,l_i}\} = L = L_\emptyset$ であり，補題 1 から $G_F = G_\emptyset$ が導かれる．　□

定理 2　RM 法によるチェックポイント・ミラーリングでは，$|F| \leq m$ である任意の F に対して，$G_F = G_\emptyset$ である．

証明　$C_{i,j}$ は，P_i および $R_{i,j}$ に保存される．$|P_i \cup R_{i,j}| = m+1$，$|\overline{F}| \geq N - m$ から，定理 2 と同様にして $\forall i, j$ $(1 \leq j \leq l_i)$，$C_{i,j} \in L_F$，すなわち，$L_F = L_\emptyset$ が導かれる．よって，$G_F = G_\emptyset$ である．　□

$|F| > m$ である場合，FM 法では $F = P_i \cup R_i$ となる F が存在しうる．この場合，P_i のすべてのチェックポイントデータがシステムから失われるため，P_i は初期状態から全タスクを再実行する．この際，ドミノ効果によって，ほとんどのプロセスに大規模なロールバックの必要が生じることが予想される．逆に，すべてのプロセスに関して，そのミラーノードのいずれかにメディア障害が発生していなければ，FM 法の回復能力はメディア障害の影響を受けない．

定理 3　FM 法において，$\forall i$，$P_i \cup R_i \cap \overline{F} \neq \emptyset$ であれば，$G_F = G_\emptyset$ である．（証明省略）

一方，RM 法では，メディア障害によって C_{i,l_i} のデータが取得できない場合でも，C_{i,l_i-1} 以前のいずれかのチェックポイントが残存している可能性がある．しかし，残存するグローバルチェックポイントからリカバリラインが発見できるか否かは確率的な問題である．次章では，これをシミュレーションによって評価する．

定理 4　RM 法によるチェックポイント・ミラーリングでは，各プロセスが $N-1$ 回以上のチェックポインティングを行っていれば，$|F| < N$ の任意の F に対して $G_F \neq \emptyset$ である．

証明　整数 N を法とした剰余類群 $\mathbf{Z_N}$ はアーベル群である．$\mathbf{Z_N}$ の代表元の集合を $S = \{[0], [1], \ldots, [N-1]\}$ とすると，$\forall i \in S$，$\bigcup_{x \in (S-[0])} i + x = S - i$ である．ただし，$S - x \equiv \{y \mid y \in S, x \neq y\}$ である．ここで，式 (2) はこの準同型写像であるから，任意の P_i について $\bigcup_{j=1}^{N-1} P_{b_{i,j}}$ は P_i 以外の全プロセスを含む．システムの対称性から，すべてのプロセスが $N-1$ 回のチェックポインティングを終えた後には，全プロセスが P_0〜P_{N-1} のチェックポイントデータを少なくとも 1 つずつ保持する．よって，定義より $G_F \neq \emptyset$ である．　□

4. 数　値　例

本研究では，平均ロールバック間隔を評価尺度として提案手法のリカバラビリティを評価した．平均ロールバック間隔は，1 つのプロセス障害を回復するためのリカバリ処理において，リカバリラインを発見するまでに各プロセスがロールバックによって遡るチェックポイント世代数の平均値である．ロールバック間隔が小さいほど，プロセスがリカバリ動作後に再実行すべきタスク量が少ないため，平均ロールバック間隔が小さい手法は，リカバラビリティが高いといえる．

これまでに，非連携チェックポインティングにおけるミラーリング手法のリカバラビリティに関する評価は報告されていない．前章で述べたように，ミラーリングの冗長度を超える多重メディア障害に対するリカバラビリティは，ミラーノード選択アルゴリズムによって異なる．本研究では，これをシミュレーションによって評価した．シミュレーションでは，各プロセスは内部イベントが $T = 100$ 回発生するごとにチェックポインティングを行うこととし，内部イベントは確率 q でメッセージを送信するイベントであるとした．また，各プロセスがそれぞれ 100 回のチェックポインティングを終えた状態ですべての障害が同時に発生するものとした．

まず，メッセージの送信頻度と平均ロールバック間隔の関係について計測を行った．N 個のノードから無作為に 2 個のノードを選択してメディア障害を発生させ，さらに，無作

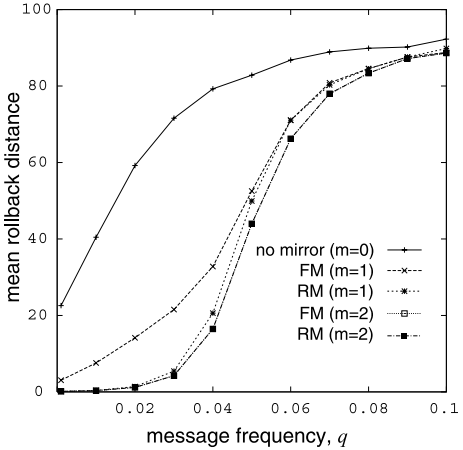

図 6　FM 手法と RM 手法の平均ロールバック間隔（$N = 8$）
Fig. 6　Average rollback distance of FM and RM methods with varying message frequency q ($N = 8$).

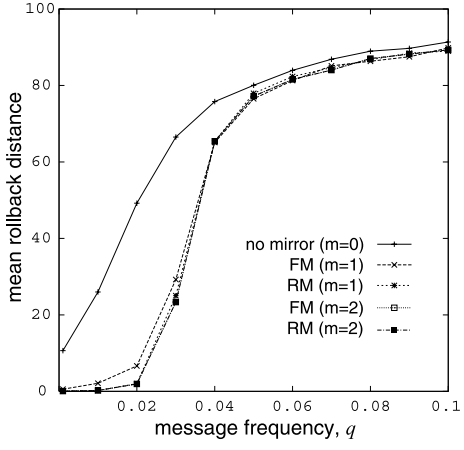

図 7　FM 手法と RM 手法の平均ロールバック間隔（$N = 16$）
Fig. 7　Average rollback distance of FM and RM methods with varying message frequency q ($N = 16$).

為に抽出した 1 ノードにプロセス障害を発生させてリカバリ処理を行わせた．$N = 8$ および $N = 16$ のそれぞれの場合について，平均ロールバック間隔の数値例を**図 6**，**図 7** に示す．図中で，横軸はメッセージを送信する確率 q を表す．また，“no mirror（$m = 0$）” は，チェックポイント・ミラーリングをともなわない通常の非連携チェックポインティングを示す．両図において，プロセス数 N にかかわらず，$m = 1$ の RM 法の平均ロールバック間隔は，$m = 1$ の FM 法のそれよりもつねに小さい．

ドミノ効果は，プロセス間で送受信されたメッセージがロールバックによって取り消されることに起因しておこるため[7]，メッセージ頻度が高いほど平均ロールバック間隔は大きくなる傾向がある[23]．図 6，図 7 の数値例は，これを反映し，q が大きいほど平均ロールバック間隔も大きくなっている．

本シミュレーションでは，障害の多重度 $|F| = 2$ であるので，冗長度 $m = 2$ である FM，RM 両手法は，メディア障害の影響を受けることなくロールバックを行うことができる．つまり，$m = 2$ の両手法の平均ロールバック間隔は，メディア障害がない場合のそれと等しい．特に図 6 において，q が比較的小さい場合には，$m = 1$ の FM 手法の平均ロールバック間隔が $m = 2$ のそれより大きいのに対して，$m = 1$ の RM 手法は $m = 2$ のそれとほぼ同等の値を示している．q が小さい場合には，RM 法は優れた耐障害性を持つといえる．一方，q が比較的大きい場合には，冗長度にかかわらず，各手法がほぼ同様の平均ロールバック間隔を示した．このような場合，メディア障害の有無にかかわらずドミノ効果の影響が顕著となるため，チェックポイント・ミラーリングは，必ずしも耐メディア障害性の向上に寄与しないといえる．

また，$N = 16$ の数値例では，$N = 8$ のものに比べて各手法間の平均ロールバック間隔の差異は小さく，より低い q で大きな平均ロールバック間隔をとっている．これは，プロセス数 N が増加すると，ドミノ効果が発生しやすくなることを意味する．本論文のシミュレーションでは，各プロセスが対称的な動作を行うため，N が大きいほど単位時間あたりに送信されるメッセージの総数が増加する．これが，プロセス間の推移的な依存関係[15] を増加させ，無矛盾なグローバルチェックポイントが減少するものと考えられる．多くのプロセスからなる分散システムでは，やはり平均ロールバック間隔へのドミノ効果の影響が支配的になる．

このように，多くのプロセスが含まれ，メッセージが頻繁に送信される分散システムでは，平均ロールバック間隔へのドミノ効果の影響が支配的になる．このような環境では，非連携手法とメッセージ・ロギングの併用が有効であると報告されている[6),24]．メッセージ・ロギングを用いる非連携手法におけるチェックポイント・ミラーリング手法の検討は，今後の課題である．

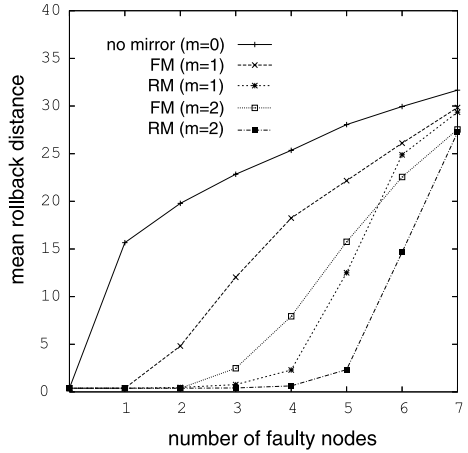

図 8　同時多重メディア障害発生時の FM 法と RM 法の平均ロールバック間隔（$N = 8$）
Fig. 8　Average rollback distance of FM and RM methods on concurrent multiple media failures ($N = 8$).

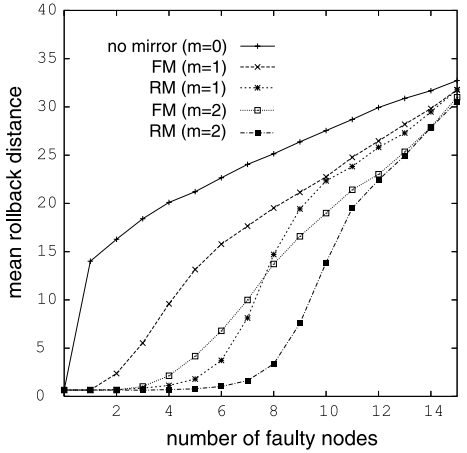

図 9　同時多重メディア障害発生時の FM 法と RM 法の平均ロールバック間隔（$N = 16$）
Fig. 9　Average rollback distance of FM and RM methods on concurrent multiple media failures ($N = 16$).

次に，各手法の同時多重メディア障害への耐性を評価した．上述のシミュレーションとほぼ同様の条件下で，メディア障害の多重度を変化させ，平均ロールバック間隔を測定した．メッセージ送信確率 q は，0.02 とした．**図 8**，**図 9** に $N = 8$ および 16 とした際の数値例を示す．横軸はメディア障害の多重度を表す．どちらの数値例でも，ミラーリングを行わない通常の非連携手法では，メディア障害が発生すると，平均ロールバック間隔が顕著に増加する．通常の非連携手法では，メディア障害時にグローバルチェックポイントが残存するのは，プロセス障害が生じたノードにメディア障害が発生しておらず，かつドミノ効果によってロールバックの必要が生じるすべてのプロセスにおいてメディア障害が生じていない場合に限られる．これは非常に強い制約であり，通常の非連携手法は，多重メディア障害への耐性を持たないといえる．

FM 法の平均ロールバック間隔は，メディア障害の多重度 $|F|$ が冗長度 m に対して $|F| \geq m + 1$ となる多重度から増加が始まる．これに対して，RM 法の平均ロールバック間隔は，同一の冗長度の FM 法に比べてつねに小さい．また，$|F|$ が m を超えてもすぐに平均ロールバック間隔が増加しない．たとえば $m = 1$ とした RM 法では $|F| \leq 3$ 程度まで，$m = 2$ では $|F| \leq 5$ 程度までの多重障害に対して，平均ロールバック間隔を非常に小さく保つことができる．さらに，$m = 1$ の RM 法の平均ロールバック間隔は，障害の多重度が比較的低い場合には，$m = 2$ の FM 法のそれよりも小さい．これらのことは，非連携手法では，あるプロセスのチェックポイントデータが数割程度失われても，リカバラビリティに及ぼす影響が小さいことを示している．RM 法は，非連携手法が元来有するチェックポイントデータの冗長性を利用し，効率的に耐メディア障害性を高めるといえる．

5. ま と め

本論文では，非連携チェックポインティングにメディア障害に対する耐性を付加するためのチェックポイント・ミラーリング手法を提案し，2 種のミラーノード選択アルゴリズムについて検討した．シミュレーションによって提案手法の同時多重メディア障害に対するリカバラビリティを評価し，ローテーショナル・ミラーリング手法が優れた回復能力を持つことを示した．この結果から，ドミノ効果の比較的起こりにくい環境下では，ミラーノードの選択方法の工夫によって，冗長度を増加させることなく耐多重メディア障害性を向上させることができることを明らかにした．

平均ロールバック間隔の解析的評価は，今後の主要な課題の 1 つである．また，メッセージ・ロギングをともなう非連携手法に適したチェックポイント・ミラーリング手法の検討は，大規模分散システムにおける非連携手法のディペンダビリティを向上させるために重要な課題である．

本論文では，複製を保持するノードの選択アルゴリズムによって，比較的低い冗長性で多重メディア障害への耐性を得ることを試みたが，チェックポイントごとに複製を行うか否かを選択することによっても同様の効果が得られる可能性がある．非連携チェックポインティングでは，チェックポイントとメッセージの時間的配置によって，決して無矛盾なグローバルチェックポイントを形成できないローカルチェックポイントが存在しうる[16]．今後，このようなチェックポイントの複製を省略することで，より低い空間オーバヘッドで同等の回復能力を有するミラーリング手法についても検討を行う予定である．

謝辞　本研究の一部は文部科学省科研費補助金（課題番号 15500046）による．

参 考 文 献

1) Chandy, K.M. and Lamport, L.: Distributed Snapshots: Determining Global States of Distributed Systems, *ACM Trans. Comput. Syst.*, Vol.3, No.1, pp.63–75 (1985).
2) Koo, R. and Toueg, S.: Checkpointing and Rollback-Recovery for Distributed Systems, *IEEE Trans. Softw. Eng.*, Vol.SE-13, No.1, pp.23–31 (1987).
3) Plank, J.S.: An Overview of Checkpointing in Uniprocessor and Distributed Systems, Focusing on Implementation and Performance, Technical Report UT-CS-97-372, University of Tennessee (1997).
4) Higaki, H., Tachikawa, T. and Takizawa, M.: Checkpoint and Rollback in Asynchronous Distributed Systems, *IPSJ J.*, Vol.39, No.4, pp.1058–1066 (1998).
5) 平山秀昭，白木原敏雄，金井達徳：分散チェックポイント方式との組合せによりフォールトトレラントシステムを実現する分散レプリケーション方式，電子情報通信学会論文誌，Vol.J82-D-I, No.3, pp.496–507 (1999).
6) Elnozahy, E.N., Alvisi, L., Wang, Y.-M. and Johnson, D.B.: A Survey of Rollback-Recovery Protocols in Message-Passing Systems, *ACM Computing Surveys*, Vol.34, No.3, pp.375–408 (2002).
7) Jefferson, D.: Virtual time, *ACM Trans. Prog. Lang. Syst.*, Vol.7, No.3, pp.404–425 (1985).
8) Fujimoto, R.: Parallel Discrete Event Simulation, *Comm. ACM*, Vol.33, No.10, pp.30–53 (1990).
9) 松本幸則，瀧　和男：並列論理シミュレーション向きタイムワープ機構の効率的な実現手法，情報処理学会論文誌（ジャーナル），Vol.37, No.4, pp.654–665 (1996).
10) 守屋　宣，櫟　粛之：インターネットエージェントのための動的スナップショットアルゴリズムと部分ロールバックアルゴリズム，電子情報通信学会論文誌，Vol.J86-D-I, No.5, pp.301–317 (2003).
11) 山口佳紀，本多弘樹，弓場敏嗣：タイムワープ法並列離散事象シミュレーションにおけるロールバックの雪崩現象，情報処理学会研究報告，1998-HPC-075, pp.151–156 (1999).
12) Plank, J.S.: Improving the Performance of Coordinated Checkpoints on Networks of Workstations Using RAID Techniques, *Proc. SRDS'96*, pp.76–85 (1996).
13) Nakamura, H., Hayashida, T., Kondo, M., Tajima, Y., Imai, M. and Nanya, T.: Skewed Checkpointing for Tolerating Multi-Node Failures, *Proc. SRDS'04*, pp.116–125 (2004).
14) 東美和子，近藤正章，今井　雅，中村　宏，南谷　崇：空間的に障害率が異なる計算機クラスタシステムにおけるチェックポインティング，電子情報通信学会論文誌，Vol.J89-D, No.8, pp.1705–1716 (2006).
15) Lamport, L.: Time, clocks, and the ordering of events in a distributed system, *Comm. ACM*, Vol.21, No.7, pp.558–565 (1978).
16) Netzer, R.H.B. and Xu, J.: Necessary and Sufficient Conditions for Consistent Global Snapshots, *IEEE Trans. Parallel Distrib. Syst.*, Vol.6, No.2, pp.165–169 (1995).
17) Manivannan, D., Netzer, R.H.B. and Singhal, M.: Finding Consistent Global Checkpoints in a Distributed Computation, *IEEE Trans. Parallel Distrib. Syst.*, Vol.8, No.6, pp.623–627 (1997).
18) Manivannan, D.: Determining Consistent Global Checkpoints of a Distributed Computation, *IEICE Trans. Inf. & Syst.*, Vol.E87-D, No.1, pp.164–174 (2004).
19) Agbaria, A., Attiya, H., Friedman, R. and Vitenberg, R.: Quantifying Rollback Propagation in Distributed Checkpointing, *Proc. SRDS'01*, pp.36–45 (2001).
20) Patterson, D.A., Gibson, G. and Katz, R.H.: A Case for Redundant Arrays of Inexpensive Disks (RAID), *Proc. ACM SIGMOD Int'l Conf. Management of Data*, pp.109–116 (1988).
21) Agrawal, D. and Agre, J.R.: Recoverying from Multiple Process Failures in the Time Warp Mechanism, *IEEE Trans. Comput.*, Vol.41, No.12, pp.1504–1514 (1992).
22) 田中勝也，桧垣博章，滝沢　誠：分散オブジェクト環境における非同期なロールバック復旧方式，情報処理学会研究報告，1999-DPS-095, pp.49–54 (1999).
23) Ohara, M., Suzuki, R., Fukumoto, S. and Iwasaki, K.: Analytical Model on Hybrid State Saving with a Limited Number of Checkpoints and Bound Rollbacks, *IEICE Trans. Fund.*, Vol.E89-A, No.9, pp.2386–2395 (2006).
24) Elnozahy, E.N. and Zwaenepoel, W.: Manetho: Trasparent Roll Back-Recovery with Low Overhead, Limited Rollback, and Fast Output Commit, *IEEE Trans. Comput.*, Vol.41, No.5, pp.526–531 (1992).

(平成 19 年 10 月 8 日受付)
(平成 20 年 3 月 4 日採録)

大原　　衛

平成 15 年東京都立大学大学院工学研究科修士課程電気工学専攻修了．平成 16 年東京都立産業技術研究所入所．平成 18 年東京都立大学大学院工学研究科博士課程電気工学専攻修了．現在，東京都立産業技術研究センター研究員．組み込みシステム，分散システムの高信頼化に関する研究開発に従事．博士（工学）．電子情報通信学会会員．

上杉　賢弘

昭和 59 年生．平成 19 年東京都立大学工学部電子情報工学科卒業．現在，首都大学東京大学院システムデザイン研究科在籍．分散システムの高信頼化研究に興味を持つ．

新井　雅之

平成 13 年東京都立大学大学院工学研究科電気工学専攻修士課程修了．同専攻助手を経て，現在，首都大学東京システムデザイン学部助教．ディペンダブルコンピューティングおよび VLSI テスト技術に関する研究に従事．博士（工学）．電子情報通信学会，IEEE 各会員．

福本　　聡（正会員）

昭和 62 年広島大学工学部第二類（電気系）卒業．平成 4 年同大学大学院博士後期課程修了．愛知工業大学助手，講師，助教授，都立大学大学院工学研究科助教授を経て，現在，首都大学東京システムデザイン学部准教授．博士（工学）．ディペンダブルコンピューティング，並列分散システム，ネットワーク，VLSI テスト等に興味を持つ．著書『コンピュータアーキテクチャ（昭晃堂）』等．電子情報通信学会，日本オペレーションズリサーチ学会，IEEE，ACM 各会員．