情報処理学会論文誌 Vol. 49 No. 6 2054–2069 (June 2008)

負荷変動と応答性能維持を考慮した高可用並列ストレージシステムのための複製利用と更新要求制御

小林 大[†1,†2] 横田治夫[†1]

スケーラブルで堅牢な大規模ストレージシステム構築のため，高機能並列ストレージシステムが注目されている．当該システムでは要求される様々な性能品質を維持するためのデータ管理が必要であるが，システムの大規模化，複雑化にともない管理コスト上昇が問題となる．時間的に変動する負荷に対する可用性維持のためのアクセス負荷均衡化は要求応答性能維持のための重要な管理項目である．動的負荷均衡化をデータマイグレーションによるデータ再配置によって行う手法は管理コスト低減に有用である．しかしながら，再配置がシステム処理能力の一部を使用するため短期的に応答性能維持が困難となりシステム可用性低下を招く問題がある．これまで，複製データを利用した RM 手法により，WEB 等の読み出しアクセス環境においてマイグレーションと応答性能維持が達成されている．本論文では，より一般的な環境下においてデータマイグレーションによる負荷均衡化と応答性能維持の両立を目指す．提案する手法では，更新リクエスト扱いの効率化を目的とし，複製への更新のみライトバックキャッシュとし，RM 手法と適応的制御を組み合わせることで要求応答性能維持を達成する．シミュレーションによる実験により提案手法の有効性を検証した．その結果，提案手法を用いることで，応答性能を維持しつつ迅速なマイグレーションが可能となることが示された．

Control of Replica and Update Operations on High-availability Parallel Storage Systems with Service-awareness over Concept Drifting Workload

Dai Kobayashi[†1,†2] and Haruo Yokota[†1]

Network parallel storage systems have attracted attention over the years for constructing large scale dependable storage systems. In such systems, it becomes difficult to maintain required service level because of enlargement and complexity. Dynamic data replacement with data migration is useful to decrease management costs by avoiding serious performance degradation caused by load concentration when access patterns change. However, it also causes temporary violation of service quality, referred as the longer latency. We have considered a solution for the problems by simultaneous use of replica data for failure recovery, while it still has required capability of effective handling of write operations. In this paper, we achieves the simultaneous pursuit of data migration and service quality under the workload containing both read and write operations. The described approach includes a replica write-back cache. It also includes the adaptive control of data migration assisted by data replication to reduce and distribute the load caused by migration. This paper demonstrates the ability of the method to keep system service quality by doing storage simulation experimentation with both synthetic and file server workloads.

1. はじめに

大規模なデータを扱う IT システムにおけるストレージシステムの重要性が高まるなか，スケーラブルで大規模・堅牢なストレージシステムを構築する要求が高まっている．高機能ストレージノードのクラスタにより構成される高機能並列ストレージシステムは，ノード数に応じた高いスループットを達成できるため注目されている．また，要求される様々な性能品質を維持するためのシステム管理・データ管理が必要とされる．しかし，システムの大規模化，複雑化にともない管理コスト上昇が問題となっている．高機能並列ストレージシステムではデータ管理の一部を，ストレージノード上の計算資源を用いて自律的に行うことで管理コストの削減が可能である．

要求される応答性能を維持することは重要な管理項目の 1 つである．アクセス負荷傾向の時間的変動に起因するアクセス負荷偏りは応答性能低下を引き起こす．磁気ディスク装置やそれを含むストレージノードは負荷が集中すると極端に性能が悪化するため，分散格納されたデータに対するアクセスをノード間で均衡化する負荷均衡化は重要であるが，同時に管理コスト増加の一因となる．

動的データマイグレーションを用いることで，システムは変化するワークロード下においてもノードへの負荷集中とそれにともなう性能低下を自律的に回避することができる．デー

†1 東京工業大学
Tokyo Institute of Technology
†2 日本学術振興会特別研究員 DC
Research Fellow, Japan Society for the Promotion of Science

タマイグレーションでは，負荷集中ノード（ホットスポット）あるいはホットスポットになると予測されるノードから，アクセスが集中するデータの一部を別のノードに移動することで負荷を均衡化する．

しかし，マイグレーションは同時に，大量のデータ転送をシステム資源を利用して行うため，短期的な可用性低下を引き起こす．データマイグレーションによるデータ転送はシステムがクライアントアクセスに用いるI/Oやネットワークといった資源を利用して行われる．特に負荷均衡化のためのデータマイグレーションでは，負荷がすでに集中しているノードからのデータ移動が求められる場合がある．このようなデータマイグレーションは一時的な応答性能低下を引き起こし，システムがSLA（Service Level Agreement）として要求される応答性能を維持するのが難しくなる．

斯様な問題に対し，複製データを利用したReplica-assisted Migration（以下RM手法）による解決手法が提案されている[12),13)]．RM手法では，障害復旧のため他ノードに格納された複製データを用いてマイグレーションによる負荷を他のノードに分散する．RM手法ではまずデータ移動元ノードの選択を行う．マイグレーション戦略立案時に，より負荷の低いノード上の複製データをデータ移動元とするマイグレーション経路選択により，データマイグレーションによる応答性能低下を防ぐ．さらに，クライアントアクセスの一部を複製データへ回送し，複製保持ノード上の余剰処理能力を利用することで，負荷集中ノードからのデータ移動をデータ移動速度を低下させることなく行う．

これまでの研究成果では，RM手法によりWEB利用等の限定的な環境下において，データマイグレーションと応答性能維持の両立が達成された[13)]．しかしながら，より一般的な環境においてマイグレーションと応答性能維持を達成することは，ワークロードの変化に対し堅牢なストレージシステムを構築するうえで肝要である．

本論文の目的は，一般的な環境において，データマイグレーションによる負荷分散およびその速度維持と，クライアントアクセスに対する応答性能の維持を両立させる手法の構築である．我々は，更新リクエスト扱いの効率化に特に着目し，次に述べる2つのアプローチによりRM手法を拡張することで，この目的を達成する．

まず，複製に対する更新のみをライトバックキャッシュにより一時的に複製保持ノード上のメモリに保持する．これにより，永続メディア上のプライマリデータによりシステムの耐故障性能を維持しつつ，複製への更新アクセスを削減および遅延させることで，RM手法で期待される複製保持ノード上のアクセス処理能力を一時的に確保し，データマイグレーションおよびクライアントアクセスの処理に用いる．また，当該キャッシュの振舞いにより，これまでのRM手法で用いていたノード負荷見積り値と，実際のノード負荷の差が大きくなる．そこで，RM手法のうちマイグレーション経路選択を適応的制御へ拡張する．RM手法では，これまで静的な負荷見積り値を利用して経路を選択していた．これを，各データ断片の移動ごとにマイグレーション移動元ノードと複製保持ノードの負荷の関係から適応的に判断してマイグレーション経路選択を行うことで，データマイグレーションによる負荷が実際の負荷に対して均等に発生するように制御する．

本論文の後半では，提案する適応型RM手法と複製ライトバックキャッシュの組合せの効果をシミュレーションを用いた実験により確認する．人工的なワークロードによって，提案手法を用いたシステムが応答性能維持ととより短いマイグレーション時間を実現可能であることを確認した．また，ファイルサーバ利用を想定したワークロードでは提案手法により，20%の要求性能超過リクエストを削減でき，その効果が確認できた．

本論文の構成を以下に示す．つづく2章では，関連研究について論ずる．3章において前提とするシステムおよびデータマイグレーション戦略について述べる．4章ではRM手法について述べる．そして，5章において，RM手法を更新リクエスト向けに拡張する．6章において，本論文における実験環境の概要を述べる．7章では人工的なワークロード，8章では現実的なワークロードを用いて，提案手法の有効性をシミュレーションにより確認する．最後に9章でまとめと今後の課題について述べる．

2. 関 連 研 究

並列ストレージクラスタ，クラスタファイルシステム，ブリックベースストレージ等の，高機能並列ストレージノードのクラスタによるシステムが注目されている．高機能並列ストレージシステムは，インテリジェントなストレージノードで構成される．ストレージノードは，HDDに加え，CPUと大容量のキャッシュメモリを備え，Gigabit Ethernetのような高速なLANに接続される．HP Fab[4)]，Self-* Storage[5)]，自律ディスク[23)]といった研究や，近年ではLustre[1)]，Google File System[6)]，Panasas ActiveScale Storage Cluster[17)]といったシステムが実用化されている．

アクセス負荷均衡化は主に無共有並列データベースの分野における主要な課題であった．Scheuermannら[19)]は，分割されたテーブルを格納するシステムにおいて，各データ断片のアクセス負荷をもとにした動的データマイグレーションによる負荷均衡化を達成した．そして彼らは動的マイグレーションと性能保証の両立が次の重要な課題であると述べていた[21)]．また，Feelifl らは並列Btree索引構造の負荷分散をデータマイグレーションにより解決し

ている[3]．このとき彼らは速度制御による性能保証を導入している．これはデータ量が比較的少ない索引構造のマイグレーションとして効果が得られている．

データマイグレーションと応答性能保証両立に対する速度制御によるアプローチは階層構成ストレージシステムにおいて多く提案されている．階層構成では上位層と下位層の間でデータを移動する必要がある．これまでに，現在のレスポンスタイムをもとにした適応的速度制御[14]，負荷変化の予想による速度制御[2]，ストレージノードの許容最大性能見積り値をもとにした速度制御[24] 等が提案されている．しかし，負荷均衡化のためのデータマイグレーションに対し，速度制御による応答性能保証手法を適用すると，負荷を均衡するために必要なデータマイグレーションが速度制御手法により大きく遅延，あるいは止められてしまう．そのため負荷が均衡化できず，その結果負荷偏りによる性能低下が発生することが分かっている[13]．そのため，速度制御ではシステムが応答性能を維持するのは難しい．

自律データ管理と応答性能保証の問題は，ストレージアレイにおける障害回復時においても問題となっている．ディスク障害にともなう突発的な負荷不均衡や，データ冗長度回復のためのアクセスが応答性能を低下させるためである．Chained Declustering[8] は，前者の突発的な不均衡を複製データへのアクセス回送により解決している．一方，Façade[16] は突発的な不均衡や冗長度回復と応答性能保証を，クライアントとストレージの間のフローコントロールにより解決している．

負荷均衡化のためのデータマイグレーションでは，複製データの利用による RM 手法が提案されている[12),13]．RM 手法では複製データを介し複製保持ノード上の余剰処理能力を利用する．これにより読み出し環境が支配的な WEB 等のワークロード下では，応答性能保持とマイグレーションの両立を達成している．しかし，ワークロード中に更新リクエストが存在する場合，これまでの RM 手法では，プライマリを含む複製間の一貫性保持のため同期更新にする必要がある．複製をクライアントアクセスに用いるためである．この同期更新のため，更新リクエストの負荷が複製側にも同時に発生し，RM 手法で仮定している複製保持ノード上の余剰処理能力を利用できない．よって更新リクエストのある場合，これまでの RM 手法ではデータマイグレーションと応答性能保証の両立は実現できない．

複製データを障害回復だけでなくアクセスにも利用する手法は，WAN における長いネットワークレイテンシの隠蔽[10),18),25)]，ビデオ配信サーバの負荷均衡化[22] 等で長く利用されてきた．しかし，高機能並列ストレージにおいては，単純なアクセス回送は各ストレージノード上のキャッシュメモリ利用率を低下させる問題がある[11),15)]．よって複製へのアクセス回送による負荷均衡化は通常時の利用にはいまだ検討を要する．このため，本論文においてはデータマイグレーション発生時という条件でのみ複製の利用を考慮する．

3. 負荷変動を考慮した並列ストレージシステム

本論文では，Gigabit Ethernet 等の，高速な LAN 上に並列接続された高機能ストレージノードにより構成されたシステムを想定する．データは，ファイル，エクステント，ページ，ブロック等の粒度で分割を行い，各ストレージノードに分散格納する．分割されたデータ断片と格納されるストレージノードの対応は，メタデータサーバや分散ディレクトリを利用したストレージ仮想化機構により利用者から隠蔽される．そのため，システムは運用中の動的データ再配置を低い管理コストで行うことが可能となる．また，各ノードはアクセスのためのデータ断片（プライマリデータ）に加え，他のノードに格納されたプライマリデータの複製データを障害復旧用に保持する．2 章で述べたように，データマイグレーション発生時以外は複製をアクセスに利用せず，データの読み出しは基本的にプライマリデータから行うことを仮定する．データの更新はプライマリと同時にすべての複製データにも同期的に行われる．

時間的に変動するワークロード下でシステム全体の性能を維持するため，システムはデータ動的再配置（データマイグレーション）による負荷均衡化を自律的に行う．データマイグレーションは，各ノードごとのアクセス負荷が要求性能を維持可能な量を超えないデータ配置を目標とする．本論文では集約評価型の負荷均衡化アルゴリズムを前提とする．すべてのノードは，自ノードに格納されるデータの現在の負荷を表す負荷値を記録している．ここで負荷とはデータ提供のための資源（CPU，I/O 等）利用量を示す度合いである．READ によるプライマリデータアクセス，WRITE によるプライマリ，バックアップ両データに対するアクセスの量で決まる．負荷値は負荷を定量的に表現する値である．例として，最近一定時間のアクセス量の平均値や，最近一定回数のアクセスの到達時間間隔，あるいはそれらの値を各ノードの性能を表す値によって正規化した値が負荷値として利用される．

集約評価型の負荷均衡化では，システムはある定められたインターバル時間ごとに以下の作業を行う．まず全ノードは，ある 1 つのノードもしくは外部管理サーバ（以下では coordinator と呼ぶ）を選定する．そして coordinator に対し自ノードの負荷値を送信する．全ノードの負荷値を受け取った coordinator は，すべてのノード間で負荷値が均衡化するデータ配置とそのためのデータ移動戦略を計算する．移動戦略はマイグレーションタスクの集合とする．ここでマイグレーションタスクは（移動元ノード，移動先ノード，移動負荷値）とする．そして各ノードに移動戦略を送信する．各ノードは指定された移動先ノードに

必要なデータを送信する．

coordinator では集約負荷値から移動する負荷値のみを決定し，具体的な移動データの特定は各ノード内で行う．これは個々の格納データごとの負荷情報を coordinator ノードへ集約することはコストが高いためである．各ノードは，与えられた戦略に含まれる移動負荷値がある閾値を超えた場合，自ノードの個々の格納データに関する負荷値を走査し，与えられた移動負荷値を満たすデータセットを特定する．各ノードは指定された移動先ノードに必要なデータを送信する．移動負荷量が閾値以下の場合，負荷均衡化よりデータ配置の安定を重視し，マイグレーションを行わない．

ここで，データの移動は Silvering [14] と呼ばれる操作により行われることを前提する．Silvering ではまず移動するデータの複製を移動先ノードに作成する．次に複製作成開始時から終了時までの更新を複製に適用する．最後に元のデータを削除する．これにより移動中データへの排他ロック取得期間を短く抑えることができる．よって，以下の議論ではデータマイグレーションのための排他制御による応答性能への影響は考慮しないこととする．

4. Replica-assisted Migration

複製データを利用したマイグレーション中の応答性能維持手法として，Replica-assisted Migration（RM 手法）が提案されている．本章ではその概要を述べる，

4.1 2 つの複製利用法

データマイグレーションによる応答性能の低下は，データマイグレーション処理を行うストレージノードのアクセス処理能力飽和により起こる．我々の提案する RM 手法では，他のノードに配置された複製データを利用し負荷集中ノードから一時的に処理能力を確保する．これにより，たとえマイグレーション対象ノードに負荷が集中した後でも，複製データ保持ノードの処理能力を用いたデータ移動により負荷均衡化を行うことを可能とする．複製の利用方法として，マイグレーション経路選択とアクセス回送の 2 つが提案されている．

マイグレーション経路選択では，coordinator により，移動対象データの複製を保持するノードのうち最も負荷の低いノードをデータ移動元ノードとして経路を設定する．**図 1** にその様子を示す．ここではデータ $pj2$ をノード n へ移動する．データ移動元ノード j の負荷が複製保持ノード k よりも低ければ，ノード j のプライマリデータ（$pj2$）を移動元と設定する．そうでなければ，ノード k 上に格納された $pj2$ の複製データ（${}_{rep}j2$）を移動元とする．

アクセス回送では，負荷集中ノードに格納されたデータへのアクセスの一部を，当該データの複製データを保持するノードに回送する．データマイグレーション対象ノードが負荷集中ノードとなった場合，このアクセス回送を用いマイグレーション対象ノード上にマイグレーション処理能力を確保することができる．**図 2** では，ノード j 上のデータ $pj3$ をノード n へ移動する．ノード j の処理能力がデータマイグレーションにより飽和するため，データ $pj4$ へのアクセスをノード k 上にある複製データ ${}_{rep}j4$ へ回送している．ただし，図中 $pj5$ のように，書き込みアクセスはプライマリ，複製共に同期更新されるためアクセス回送の対象外となる．

アクセス回送は次のように行う．まず複製との一貫性保持が非同期であれば同期に切り替

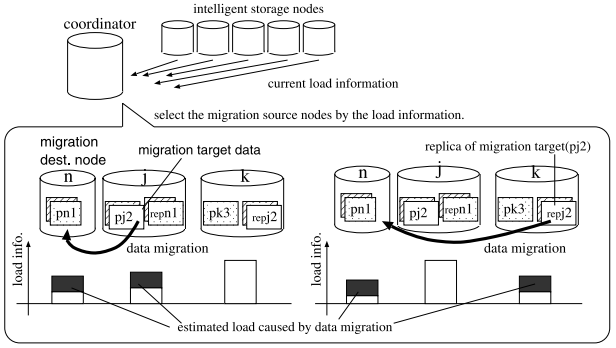

図 1　RM 手法における 2 種類の複製利用法 (1) マイグレーション経路選択
Fig. 1　Relica usage in RM method (1). Migration source node selection.

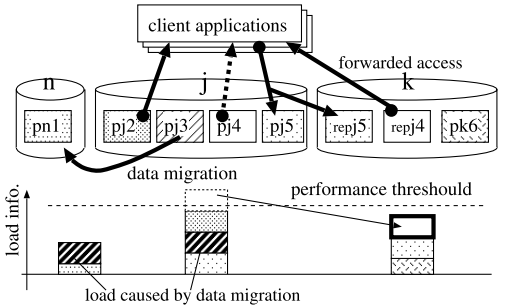

図 2　RM 手法における 2 種類の複製利用法 (2) アクセス回送
Fig. 2　Relica usage in RM method (2). Access forwarding.

え，複製に対してまだ適用されていない更新を適用する．移動元ノード i に対するアクセス要求を与えられた回送率 r_i の割合で，複製の存在するノード j へと回送するようにセットする．続いてマイグレーション戦略に基づきデータを移動する．その間のノード i へのアクセス要求の一部は回送率 r_i とアクセス回送判定手法[11] に基づき複製ノード j へ回送される．データ移動完了後，複製への要求回送率 r_i を 0 にする．

4.2　複製利用法の制御

RM 手法では，3 章で述べた coordinator によるマイグレーション戦略立案後，得られた戦略を利用して次のことを行う．

まずマイグレーション経路選択アルゴリズムにより，より負荷の低いノードにマイグレーションタスクを割り当てる．マイグレーション経路選択アルゴリズムでは戦略から得られるマイグレーションタスク 1 つごとに，そのデータ移動元ノードを一番負荷が低い複製ノードに割り当てる．

つづいて RM 手法のうち回送率計算アルゴリズムにより，アクセス回送率を算出する．回送率計算アルゴリズムでは，クライアントからのアクセスに対する処理能力を確保しつつ，マイグレーション対象タスクにより多くの処理能力を確保する．アクセス回送率計算アルゴリズムでは，マイグレーション対象ノード間で，空き負荷値（許容最大負荷値とノード負荷値の差）を割当てタスク数の比となるよう，アクセス回送量を決定する．マイグレーション対象ノード n の複製保持ノード $n+1$ へタスクが割り当てられていない場合，ノード n へマイグレーション用処理能力を確保するため，ノード n のすべての読み出しアクセスを $n+1$ へ回送する．ここでノード $n+1$ の許容最大負荷値を超えた場合，超過分に相当する $n+1$ へのアクセスをさらにその複製保持ノード $n+2$ へと回送する．ノード n，$n+1$ ともに M_n 個，$M_{(n+1)}$ 個のマイグレーションタスクの対象の場合，両ノードで空き負荷値（許容最大負荷値とノード負荷値の差）が $M_n : M_{(n+1)}$ となるように回送負荷値を決定する．以上を各ノードに対して行い，現在負荷値に対する回送負荷値の割合を回送率 r_i とする．

5.　複製利用を考慮した更新要求効率化

本論文では，RM 手法を拡張することで，更新リクエスト環境下での応答性能維持とマイグレーションの両立を実現する．

5.1　複製利用を考慮したキャッシュ制御

RM 手法の問題の 1 つが，同期更新リクエストによる複製保持ノード側の負荷上昇である．既存の複製を用いたシステムでは，プライマリと複製間で非同期更新を用い複製側の負荷上昇を抑えるものがある．しかし，我々の用いる複製はノード障害時のデータ復旧，回送されたアクセスの処理に利用されるためプライマリと同期更新される必要がある．

更新リクエストの影響を抑えるもう 1 つの方法にライトバックキャッシュがある．ライトバックキャッシュでは磁気ディスク等の低速な永続メディア上のデータを更新する前に，高速なキャッシュメモリ上に一時的に更新を保持する．これにより，同じブロックへの更新はキャッシュメモリ上で削減でき，更新リクエストが低速メディアへ適用されるタイミングをある程度制御できる．しかしながら，キャッシュ上で未適用の更新リクエストは，電源障害により喪失するという問題がある．プライマリ保持ノード，複製保持ノードの両方で同時に電源障害が起こることでデータが失われる．

そこで，本論文では複製への更新のみをライトバックキャッシュに保持する．これにより複製保持ノード上の更新リクエストを削減・遅延させることで一時的に処理能力を確保する．RM 手法を用いこの処理能力を利用し，応答性能を維持しつつデータマイグレーションを行う．電源障害時にもプライマリ側データは永続メディア上に保存されていることで，データ永続性が保たれる．

あるノードにメディア障害が発生した場合，ノード上のプライマリデータの冗長性は他のノード上のキャッシュもしくは永続メディア上に保持された複製データを用いることで回復される．電源障害が発生した場合，システムは一時的に停止する．電源回復後，プライマリ保持ノードは永続メディア上の最新データを用い，他のノード上の複製データを最新データに更新する．プライマリ保持ノード上にメディア障害が発生し，かつ複製保持ノードにメディア障害が発生した場合もしくは更新データを永続メディアへ書き戻す前に電源障害が発生した場合にのみデータは失われる．

N ノード，1 エンクロージャあたり n_e ノードのシステム（**図 3**）におけるデータ喪失確率の概算を**表 1** に示す．ただし複製データは他のエンクロージャ上のノードに配置するとする．ここで，P_S はストレージノード，P_P はエンクロージャ上電源，P_U は UPS の故障率をそれぞれ示す．また，t_R はデータ冗長度回復時間，t_c はキャッシュ上未適用更新の書き戻し時間を示す．ただし，データ冗長度回復を「故障したストレージノード上のデータの複製を保持するノードが，冗長性を復旧するため他のノードにデータを複製する」ことと定義し，この複製転送時間をデータ冗長度回復時間 t_R とした．計算の過程は付録 A.1 に記す．表より，P_S，P_P，P_U が同程度の値であった場合，上記 2 つは $O(P^2)$，下は $O(P)$ となる．UPS 故障に対して，提案するキャッシュ方式は同程度の信頼性となる．

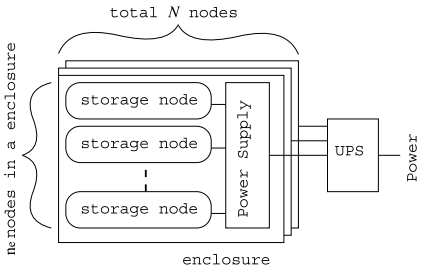

図 3　データ喪失確率計算のためのシステムモデル
Fig. 3　System model.

表 1　データ喪失確率の比較
Table 1　Probability to data lost.

Pri.	Rep.	データ喪失確率
WT	WT	$NP_S{}^2t_R$
WT	WB	$NP_S{}^2t_R + 2NP_SP_Pt_c$
WB	WB	$NP_S{}^2t_R + 3NP_SP_Pt_c + NP_p{}^2t_c/n_e + P_U$

※ WT は Write Through, WB は Write Back の略.

5.2　適応的経路選択の導入

当該キャッシュの振舞いにより，これまでの RM 手法で用いていたノード負荷見積り値と，実際のノード負荷の差が大きくなる．キャッシュ上での更新リクエスト削減および遅延はリクエストの傾向により大きく変化するため予測が難しいからである．そこで，RM 手法のうちマイグレーション経路選択を適応的制御へ拡張することで，より正確にノード上の資源へマイグレーションタスクを割り当てる．

4.1 節で述べた，マイグレーション経路選択では，その判断に静的な負荷見積り値を利用して選択していた．これを，各データ断片の移動ごとに複製保持ノードの負荷から判断してデータ移動元を選択することで，データマイグレーションによる負荷が実際の負荷に対して均等に発生するように制御する．

提案手法による改良は次のとおりである．複製データ保持ノードはある短いインターバル時間ごとにプライマリデータ保持ノードに対し短期的アクセス負荷に関する情報を送信する．短期的アクセス負荷とは次のインターバルまでのノード性能を表す値である．たとえばディスクキュー長等が短期的アクセス負荷を表す．

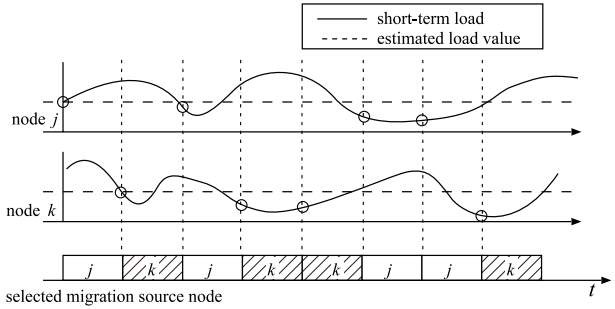

図 4　マイグレーション経路選択の適応的処理の概要
Fig. 4　Concept of Adaptive control of migration source node selection.

プライマリデータ保持ノード上のマイグレーションタスクは，1 つのデータのマイグレーションごと，あるいはあらかじめ定められたまとまったサイズのファイル群のマイグレーションごとに，マイグレーション経路を変更する．短期的アクセス負荷が小さいノードが移動元となるよう，データ移動元を選択する．**図 4** に概要を示す．ノード j,k がそれぞれデータ移動元データ群のプライマリ・複製保持ノードとする．提案する適応的制御では，各データ断片のマイグレーション開始タイミングで，現在の短期的アクセス負荷が低いノードをデータ移動元とする．

キャッシュ等により負荷見積り値に過誤が生じている場合，負荷見積り値とは異なる要因から算出される短期的アクセス負荷の値を用いることでこの過誤が是正される．また，プライマリデータ保持ノード，複製データ保持ノードが共にホットスポットとなってしまった場合でも，この適応的経路選択によりマイグレーションタスクが等分に割り振られる．その結果，経路選択を行わない場合に比べマイグレーション時間の削減が期待できる．

6. 実 験 概 要

本章以降ではシミュレーションによる実験結果を示す．実験の目的は，本論文で提案した RM 手法の 2 つの拡張の有効性を検証することにある．まず本章で，実験設定の概要を説明する．

並列ストレージシミュレーションプログラム上に高機能並列ストレージシステムの 1 つである自律ディスク[23] システムを構築し，そのうえで実験を行う．シミュレーション内のシステムに対し時間的に変化するワークロードを発生させ，ある時点でデータマイグレーショ

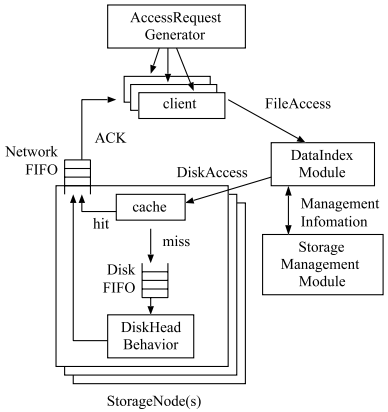

図 5　ストレージシミュレーションの構成
Fig. 5　Structure of a system on the simulation.

ンによる負荷均衡化処理を発生させた．そのうえで，各手法の応答性能，マイグレーション時間を観測することで，提案手法とその他の手法の応答性能保持能力を比較する．

本実験ではまず人工的なワークロードを用いて，異なるパラメータ下での挙動を観察し，性能維持効果とマイグレーション速度減少率について検証する．実験では，Zipf 分布に従うアクセス分布とポワソン到着に従うアクセス到着率を持つワークロードを発生させマイグレーションの挙動と応答性能の変化を観測する．2 つ目の実験として，更新リクエストを含む現実的なワークロード下での挙動を観察することで手法の現実システムへの適用可能性を考察する．ここでは，大規模なファイルサーバから採取されたアクセスログをもとにしたワークロードを用いる．

6.1　システム構成

待ち行列を利用したイベント駆動型のストレージシミュレーションプログラムを構築し，実験に用いた．シミュレーションの構成を**図 5** に示す．1 つのストレージノードは，cache，ディスク FIFO，DiskHeadBehavior，ネットワーク FIFO により構成される．シミュレーション内におけるディスクアクセス時間は，図中 DiskHeadBehavior の部分で，HGST Deskstar T7k500[7] をもとにした**表 2** に示すパラメータと前回アクセス時のヘッド位置から計算する．各ストレージノードのネットワークは 800 Mbps で処理可能な FIFO で表現した（NetworkFIFO）．ディスクへのアクセスは buffer size 単位に分割して行う．キャッシュの追い出しアルゴリズムは LRU とした．

表 2　シミュレーション構成設定
Table 2　Configuration of storage simulation.

simulation parameter	value
ディスク回転速度 (RPM)	7,200
ディスクサーフェイス数	6
シリンダあたりのセクタ数	4,320 to 9,000
ゾーン数	29
シークタイム（最長，最短）(msec)	15.1, 0.8
バッファサイズ (KB)	8,196
ディスクノード数	4
ノードキャッシュサイズ (MB)	512
ネットワーク速度 (Mbps)	800

ディスクノード数については 4 とした．これは，8 章の実験において用いるファイルサーバワークロードで，高負荷時に十分な応答時間の悪化が見込まれる規模を基準に選択した．一般に並列ストレージシステムははるかに大きな規模で運用される．しかしながら，格納コンテンツは多数の独立したドメインから独立して利用される場合が多く，ファイルシステム等のワークロードには自己相似性が見られることが報告されている[9]．また，本提案手法の処理量はノード数 N に依存しない．したがって，より大規模なシステム構成については，本実験規模のシステムの集合であると考えることで，提案手法の有効性は示される．なお，同一傾向のワークロード下における，ノード数の差異と提案手法の効果の関係については，7.6 節において，ノード数を変更した実験により議論する．

負荷均衡化のためのデータマイグレーション戦略については 3 章に示す戦略を実装した．負荷値は，時間による幅固定のスライディングウィンドウ方式で計算した．負荷値を計算するために，各ファイルごとにアクセス履歴として過去一定時間内のアクセスについて要求サイズとリクエスト到達時刻を保持する．本実験ではこれを 600 秒間と設定した．到達時刻は，到達してから 600 秒以上経過したアクセスを履歴から削除するために用いる．ストレージノードのアクセス処理性能は要求サイズごとに異なるため，要求サイズは，予備実験によって求めた各ストレージノードの性能をもとに負荷を示す負荷量に変換し，サイズによる負荷の違いを考慮する．戦略立案時には，アクセス履歴中の合計負荷量を計測期間 600 秒で割った，単位時間あたりの負荷量を負荷値として用いる．

提案する適応的マイグレーション経路選択で用いる短期的負荷としては，各ノード上のディスク FIFO，ネットワーク FIFO 中の待ちリクエスト数の和とした．

7. 人工的なワークロードによる実験

実験ではまず人工的なワークロード下での挙動を観察する．

7.1 環　　境

表 3 にワークロード緒元を示す．

実験では，Zipf 分布に従うアクセス分布とポワソン到着に従うアクセス到着率を持つアクセスをシャッフル関数を用いて格納ファイルに分散する．Zipf 分布は一部のファイルに多くのアクセスが集中するモデルである．表中の式 f は，N 個のファイルのうち k 番目に多くアクセスされるファイルへのアクセス割合を示し，パラメータ s の値が大きいほど偏りが大きい．そして実験開始一定時間後にシャッフル関数の乱数種を変化させ，アクセス傾向を変化させる．ここでは，アクセス履歴を過去 600 秒間保持することから，実験開始後 600 秒後と設定した．通常のシステムでは一定時間ごとの負荷評価によりアクセス傾向の変化が感知された時点をマイグレーション開始時間とするが，今回の実験では各実験の条件を合わせるため，アクセス傾向変化後一定時間後にマイグレーションによる負荷均衡化を実行する．その後実験終了までの応答性能およびマイグレーション速度の変化を観察する．以上の実験を，全リクエスト数に占める更新リクエスト数の比率（以下，更新比率）が 50%，25%について，異なるアクセス到着率 [req/sec] において計測を行った．なお，今回の実験では簡単化のため，更新は格納ファイルの書き換えのみとし，新規ファイルの挿入は行わないものとした．

そのうえで，各マイグレーション性能保持手法使用時のアクセス応答時間を測定する．本実験では，まずキャッシュ方式としてライトスルーキャッシュ（write through），複製側のみライトバックキャッシュ（replica write-back）を用い，応答性能保証制御なしの通常のマイグレーション（NORMAL）と，参考として提案する適応的経路選択（Adap-RM）と組み合わせた実験結果を示す．また，マイグレーション経路選択比較として，NORMAL, Adap-RM，および RM 手法の静的経路選択（Static-RM）の 3 種類を replica write-back 下で比較する．また，実験の結果では同時に NORMAL においてプライマリ，複製ともにライトバックにする手法（NORMAL + all write-back）の結果も参考までに併記した．ただし前述のとおりこの手法には電源障害時の耐故障性において問題がある．

表 3　人工的ワークロード実験緒元
Table 3　Specification of experiments with synthetic workload.

workload parameter	value
時間 (hour)	2
更新比率	50%, 25%
格納ファイルサイズ (GB)	100 × 2 (Primary & Backup)
格納ファイル数	100,000 (1 MB each)
アクセス分布	Zipf $\left(f = \frac{1/k^s}{\sum_N 1/n^s}\right)$
Zipf 係数 s	1.5
アクセス到着分布	ポワソン到着

今回の実験における必要性能としては 1 リクエストあたりの目標レスポンスタイムを 0.2 秒以下と設定した．

7.2 実験結果 1：応答性能

まず，アクセス傾向変化後 60 秒後をマイグレーション開始時刻と設定した実験の結果を示す．アクセス傾向変化の影響が，負荷評価のためのアクセス履歴にある程度反映されるためのインターバルとして 60 秒と設定した．このマイグレーション開始時刻が変化した場合の影響については 7.5 節で述べる．ワークロードは 50%が更新アクセスとした．

図 6 に各実験における平均応答時間を示す．write through を用いた 2 本の曲線はともに 300 req/sec を超える負荷に対し急激な応答時間の上昇を示している．write through で

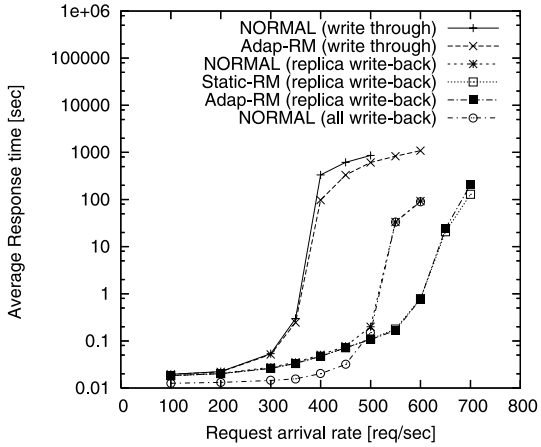

図 6　アクセス傾向変化後 60 秒後にマイグレーションを開始した場合の平均応答時間（read : write = 1 : 1）
Fig. 6　Average response time (read : write = 1 : 1). The migration started 60 seconds after the workload trend was changed.

は RM 手法は効果がないことが分かる．これは複製データ保持ノード上にも大きな更新負荷が存在し，マイグレーション処理用の資源を確保できないためである．

一方，replica write-back では，NORMAL が 500 req/sec で応答時間が大きく上昇しているのに対し，2 つの RM 手法ではより大きな負荷に対してもゆるかに応答時間が上昇している．この結果より，RM 手法と replica write-back の併用が，データマイグレーションによる応答性能低下に対し効果的であることが分かる．

なお本実験では Adap-RM と Static-RM の間に性能差が見られなかった．この理由については 7.6 節における考察で触れる．

NORMAL + all write-back の組合せは，低い負荷に対しては他のいずれの手法よりも良い応答性能を示した．しかしながら，500 req/sec を超えると，NORMAL + replica write-back と同様に応答時間が大きく上昇している．これより，複製保持による性能向上はデータマイグレーションと応答性能の維持の両立に効果があるが，加えてプライマリへの更新をメモリ上に保持することは効果がないことが分かる．この傾向は，以降の実験結果においても同様に見られる．

負荷の上昇にともない，応答時間が急激に増加する理由について考察する．各リクエストの応答時間は FIFO 待ち時間＋ FIFO 内サービス時間（ディスクアクセス，ネットワーク転送）である．アクセス到着率の上昇にともない FIFO 内リクエスト数（キュー長）が増え，FIFO 待ち時間が上昇する．本実験のワークロードでは，4 台のストレージノードのうち，1 台のディスク（以下 disk1）がボトルネックとなっていた．**図 7** に，Replica write-back 適用をした NORMAL，Adap-RM 各手法に対するアクセス到着率が 500 および 550 req/sec の場合の，disk1 上のネットワーク FIFO のキュー長の変化を示す．キュー長が変化せず一定量または減少傾向であることは，ストレージノードのアクセス処理能力が要求処理アクセス量を上回っていることを示す．キュー長が上昇傾向であることは，アクセス処理能力を上回る量のアクセスが到達していることを表す．図 7 より，いずれの手法も $t < 660$ では一定キュー長を保っている．$t > 660$，すなわちマイグレーション開始後は大きく振舞いが異なる．ここで，各手法でキュー長が高頻度で上下しているのは，アクセス到着率をポワソン到着としているためであることに注意されたい．Normal では，$t > 660$ でキュー長が増加傾向に転じている．これはマイグレーションによる負荷が発生したため，クライアントサービスとマイグレーションをあわせた負荷が，ストレージノードの処理可能量を上回ったためである．ただし，500 req/sec の場合では，マイグレーション処理が進み負荷均衡化が進んだため，disk1 上のアクセス量が減り，$t > 750$ でキュー長が減少傾向に転じている．よって，平均応答時間の上昇が少なかった．一方，アクセス到着率 550 req/sec における Normal では，キュー長が増加し続けている．500 req/sec よりも大きなアクセス負荷によりマイグレーション処理が進まず，負荷均衡化が進まないためである．このような増加し続けるキュー長が大きな平均応答時間上昇につながる．一方，Adap-RM では，500 req/sec の場合と 550 req/sec の場合いずれにおいても，キュー長が一定であり，アクセスを処理できている．これはマイグレーションに起因するアクセスを disk2 上の複製を用いて処理したためであると考えられる．このように，各手法における平均応答時間の急激な上昇は，負荷集中ノード上の負荷均衡化が進まずアクセス集中がさらにキュー長を増加させた結果である．

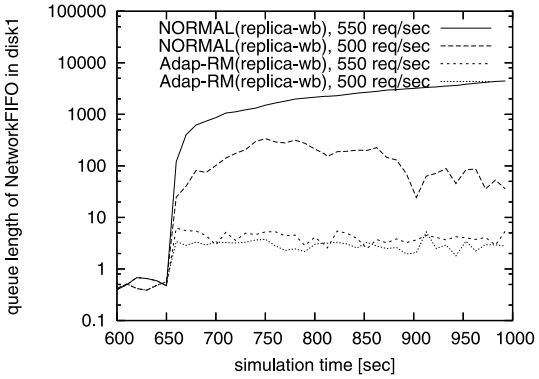

図 7　disk1 内 NetworkFIFO のキュー長の変化．グラフは 10 秒ごとの FIFO 内リクエスト数の平均の時間推移を表す

Fig. 7　Queue lenght of NeworkFIFO in disk 1 (10 seconds mean).

また，Write Through Cache ではいずれの手法でも disk1 のディスク FIFO がボトルネックとなっていた．アクセス到着率 400 req/sec では disk1 上ディスク FIFO のキュー長は秒間 17 リクエスト増加しており，実験中減少に転ずることはなかった．これが平均応答時間悪化の原因である．また，その複製保持ノード disk2 のディスク FIFO キュー長も秒間 6 リクエスト増加しており，このため RM 手法の効果が得られなかったと考えられる．

7.3　実験結果 2：応答性能維持能力

つづいて，各手法の応答性能維持能力を見る．ここでは応答性能維持能力の指標として，要求最大応答時間を超えたリクエスト（超過リクエスト）の割合を用いる．**図 8** は 7.2 節の実験において，要求最大応答時間である 0.2 秒を超えた超過リクエスト数を示す．同図は

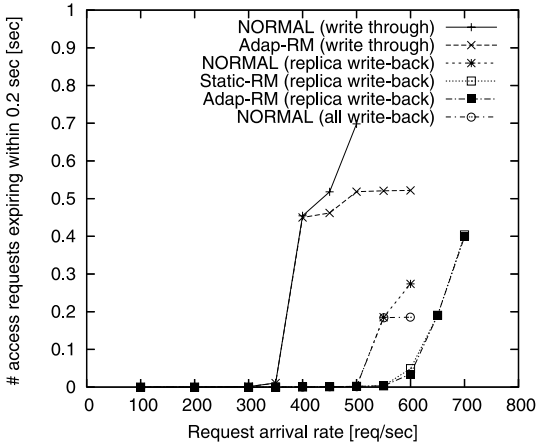

図 8　要求応答時間である 0.2 msec を超過したレスポンスの全体に占める割合（read : write = 1 : 1）
Fig. 8　The ratio of the number of access requests expiring within 0.2 sec. (read : write = 1 : 1).

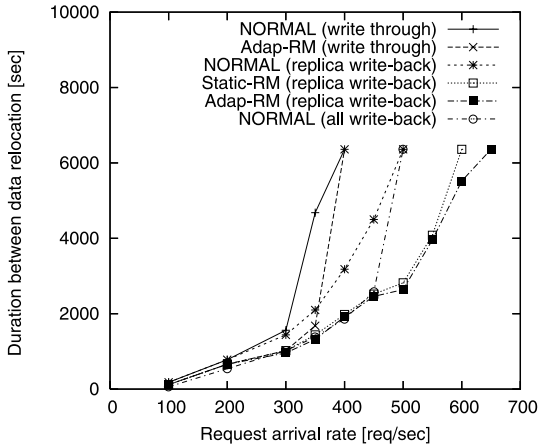

図 9　マイグレーションに要した時間（read : write = 1 : 1）
Fig. 9　Effect of reducing migration duration (read : write = 1 : 1).

応答時間の絶対値（図 6）と同様の傾向を示している．RM 手法は，replica write-back と併用することで大きな負荷においても超過リクエスト数を低く維持できている．

また，図では，NORMAL + replica write-back では 500 req/sec を超えた時点で初めて超過リクエストが現れている．RM 手法のアルゴリズムは超過リクエストの数を減らすように設計されている．よって，500 req/sec 以下では RM 手法は応答性能維持機能が働かない．図 6 において 500 req/sec 以下では RM 手法と NORMAL の間に差が現れないのはこのためである．

要求応答時間超過レスポンス数が増加するのも，アクセスの集中によりキュー長が増加した結果である．前節で解析したとおり，あるアクセス到着率を超えると，大量のクライアントアクセスがキュー内を占拠しマイグレーション処理が遅滞されるため，キュー長が減少せず，アクセス応答時間が回復しない．よって，超過レスポンスが急増する．

7.4　実験結果 3：マイグレーション時間

負荷均衡化のためのマイグレーションでは，マイグレーション時間は重要な要素である．**図 9** に 7.2 節の実験においてマイグレーションに要した時間を示す．負荷が高くなるにつれ，マイグレーション時間が増加しているのが分かる．これはマイグレーションに利用可能な資源が減るためである．図から，RM 手法は応答性能に加え，マイグレーション時間の削減にも貢献していることが分かる．これは，一部のマイグレーションタスクをホットスポットから他のノードへ割当て変更したためクライアントアクセスとデータマイグレーションによる資源競合が減ったためである．

図 9 の曲線は応答性能とは異なる傾向を示している．1 つは，write through においても Adap-RM 手法が優位である点である．これは，アクセスが集中し資源が飽和するプライマリ，複製側両ノードに対して適応的 RM が均等にマイグレーションタスクを割り当てているためである．もう 1 つは，replica write-back において，500 req/sec 以下においても RM 手法の効果が見られる点である．これは，低負荷時においては RM 手法は，クライアントアクセスへ十分な資源を確保したうえで，マイグレーションに対しても最大限に資源を確保するために複製を用いるように設計されているためである．

以上の実験結果から，提案する RM 手法を導入することでストレージシステム管理者は同じ構成のシステムを 10%から 20%高い負荷のシステムにおいても，応答性能を維持しつつより少ないマイグレーション時間で運用することが可能となることが分かった．

7.5　実験結果 4：提案手法の有効性とキャッシュ容量

キャッシュメモリのサイズは有限であるため，キャッシュオーバフローが起こりうる．**図 10** はマイグレーション開始時点を変化させたときのマイグレーション時間の変化を

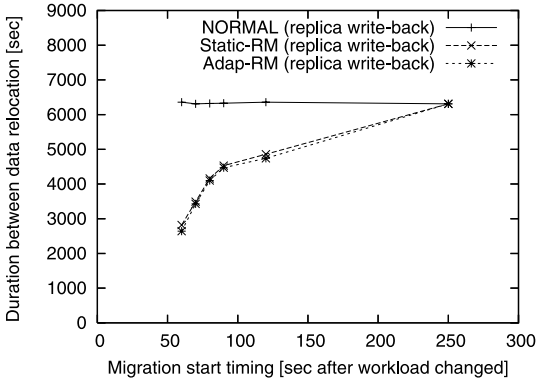

図 10　異なるマイグレーション開始時間に対するマイグレーションに要した時間（read : write = 1 : 1）
Fig. 10　The relationship among migration duration and migration start timing (read : write = 1 : 1).

表している．ここで，アクセス到着率は 500 req/sec で，更新比率は 50%である．図より 60 秒後の開始では手法間に大きな差異が見られるが，開始時刻が遅い実験では RM 手法と NORMAL の差が小さくなることが分かる．

これはマイグレーション開始時刻が遅れることで，アクセス偏りによる高負荷の更新アクセスがキャッシュからディスクに書き戻され始めるからである．アクセス傾向の変化の結果，アクセス偏りにより負荷集中ノードができる．負荷集中ノードの複製保持ノード上のキャッシュには多量の更新リクエストが発行されディスクへ書き戻されない更新データが溜まり始める．キャッシュ上が更新データでいっぱいになると，複製保持ノードは更新データをディスク上に書き戻し始める．ディスクに書き戻すよりも早く新しい更新リクエストが到達する場合，新しい更新リクエストはライトスルーで処理されるようになり，ライトバックキャッシュの効果がなくなる．これまでの実験により，性能保持をしつつマイグレーションを行うためにはライトバックキャッシュの効果が重要であることが分かる．複製保持ノード上のキャッシュが更新データで埋まる前に，マイグレーションにより負荷均衡化をする必要がある．

一方，データマイグレーションによる負荷均衡化は，時間のかかる処理であるため，長期的な負荷変動にのみ利用すべきである．現在の負荷集中が短期的なものか見極めるため，マイグレーションの発動は要求性能が維持可能な範囲で遅い方が良い．よって，データマイグレーション戦略およびマイグレーション開始時間の決定においては，更新アクセス混在環境

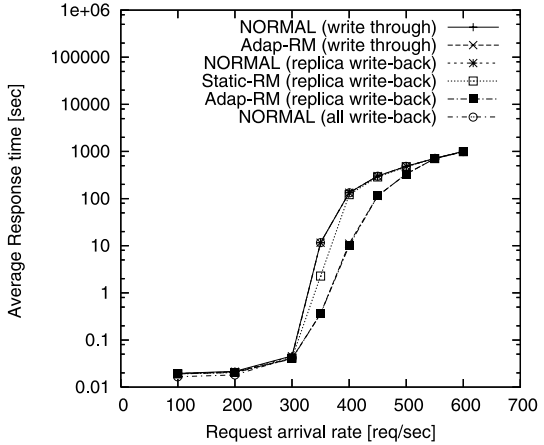

図 11　平均応答時間（read : write = 3 : 1）
Fig. 11　Average response time (read : write = 3 : 1).

下で RM 手法を使用するにはライトバックキャッシュの効果を持続させつつ，要求性能が維持可能なタイミングを，キャッシュ容量や更新アクセス比率等を考慮する必要がある．

7.6　実験結果 5：適応的制御による応答性能維持機能

これまでの実験結果では Adap-RM と Static-RM について差が見られなかった．マイグレーション経路選択の効果は，RM 手法やマイグレーション戦略に用いる負荷見積り値の誤差が大きい場合に現れる．そこで，これまでの更新比率 50% よりも読み込み比率を増加させた実験を行った結果，キャッシュ上での書き込みアクセス量が変化し，Adap-RM と Static-RM について差が現れた．

図 11 および，**図 12** は更新比率が 25%の実験における，平均応答時間および超過レスポンス数を表す．この結果では Adap-RM と Static-RM の間に有意な差が見られる．これは，適応的経路選択により負荷見積りの誤差による性能低下を回避した結果である．この負荷見積りの差は，本実験では性能のボトルネックがライトキャッシュ上で除去された冗長な更新リクエストの割合の変化によりストレージノード上の HDD とネットワークの間で揺れ動いたためである．

また，更新比率 25%の実験では，いずれの手法も 300 req/sec 程度で性能が悪化しており，更新比率 50%よりも性能が低い．これは，ライトバックキャッシュ上で除去される同じ

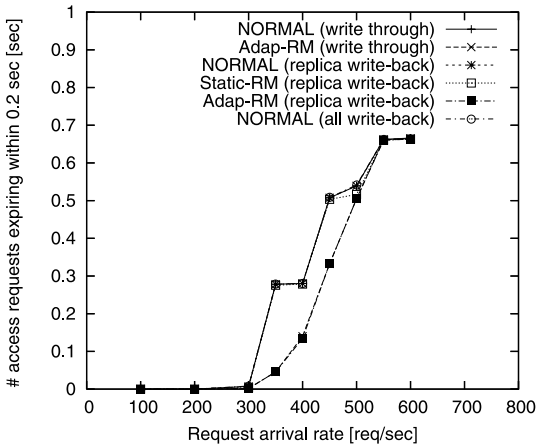

図 12　要求応答時間超過リクエスト数の割合（read : write = 3 : 1）
Fig. 12　The number of access requests expiring within 0.2 sec. (read : write = 3 : 1).

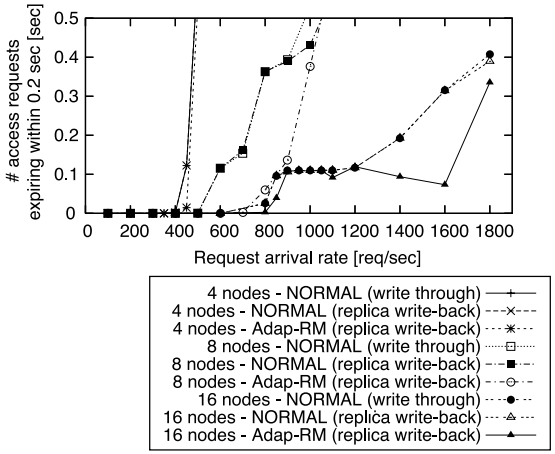

図 13　ノード数ごとの要求応答時間超過リクエスト数の割合（read : write = 3 : 1，zipf $s = 1.2$）
Fig. 13　The number of access requests expiring within 0.2 sec when the number of nodes are changed (read : write = 3 : 1, zipf $s = 1.2$).

ブロックへのアクセス数が減少したためである．また，同様の理由により，ライトバックとライトスルーの性能差もずっと小さくなっている．

7.7　実験結果 6：スケーラビリティ

最後に，ノード数が増加した場合の適応的 RM の効果について考える．**図 13** に，ノード数 4，8，16 の場合の要求応答時間超過リクエストの割合を示す．格納データ量はこれまでの実験の 1 倍，2 倍，4 倍とした．ただし，この実験に限り本節後述の理由により Zipf 係数を 1.2 とした．更新比率は 25%である．

図より，どのノード数においても Adap-RM により超過リクエスト数削減効果が見られる．ノード数 8 の場合はノード数 4 の場合よりも，Adap-RM による効果はより高いアクセス到着率に対しても示されている．これは，同じアクセス偏り度合いでノード数を増加させたため，複製保持ノードの負荷が負荷集中ノードに比べ相対的に低下したため，より多くの複製保持ノード資源（キャッシュやアクセス能力）がマイグレーションに利用できたためである．本論文における提案手法のうち，複製のみライトバックキャッシュについても，適応的制御についても，各ノードとその複製保持ノード間の通信のみを利用しており，処理量はノード数 N に依存しない．そのため，ノード数増加による提案手法の有効性低下はない．

一方，図 13 におけるノード数 16 の場合，アクセス到着率 800 req/sec 程度でどの手法においても，超過レスポンス数の割合が 11%程度まで上昇している．これは，マイグレーションをする前の状態ですでにアクセス偏りが起きているためである．今回用いた単純な Zipf 分布ではわずかなファイルにより多くのアクセスが集中する．$s = 1.2$ の設定の場合，最も多くアクセスされる単一のファイルに，アクセス全体の約 1/8 が集中してしまう．そのため，ノード数 9 以上では，最もアクセスの集中するノードのアクセス負荷を全体の 1/8 以下にできず，データ再配置のみでは負荷の均衡化が達成できない．$s = 1.5$ 等のより大きなパラメータではこのアクセス集中度合いが顕著であるため，本実験に限り $s = 1.2$ を用いた．これは，極度なアクセス偏りに対応できないというデータマイグレーションの問題と，実際の大規模なストレージシステムにおいて単一ファイルにアクセスが集中することは稀であることから単純な Zipf 分布によるモデリングの限界と考えられ，提案手法のスケーラビリティの問題ではないと考える．実際，提案手法を組み合せた RM 手法の場合，急激な超過レスポンス数上昇はアクセス到着率 1,600 req/sec と 8 ノード時の 2 倍まで抑えられており，性能は線形に増加している．

以上の実験結果から，適応的経路選択および複製のみライトバックキャッシュを利用することで，更新リクエストが含まれる環境下においても，システムのアベイラビリティを保ちつつデータマイグレーションを効率良く行うことが可能となる．

8. ファイルサーバワークロードによる実験

つづいて，提案手法の現実における有効性を示すため，現実的なワークロードを用いた実験結果を示す．ここでは，HP 社により公開されている実際のファイルサーバから採取されたトレースログ[20] をもとにワークロードを生成する．

本実験では，32 KB ブロックあたり 0.1 秒を要求最大応答時間とする．これは通常時の約 10 倍の値である．

格納データセットおよびワークロードは次のように作成する．まず `ls` コマンドの結果，およびトレースログの offset 値よりデータセットを作成する．各データにはログ登場順に一意の ID（Raw Id）を振る．つづいて，Raw Id に下記の変換を行うこと．この File Id 順に並ぶファイルを 4 等分し，4 台のストレージノードに分散格納する．

$$fileid = rid \oplus ((\overline{rid} \wedge \texttt{0xff}) \times 2^{14}) \quad (1)$$

ここで，ストレージシステムが階層構造となっていることを想定し，格納するデータはアクセスログ中でアクセスのあったデータのみとする．

ワークロード中の各アクセスはステートレスとし，トレースログ中の `READ`，`READV`，`WRITE` および `WRITEV` システムコールから作成する．よって `MMAP` によるアクセスは本実験では再現しない．以上で作成されたアクセス系列を 1 日単位で分割する．そして複数の日の同時間帯のアクセスをまとめて同時に利用することで，複数の異なる大きさのワークロードを実現する．

8.1 実験結果

平均応答性能，および超過レスポンス数の割合に関する実験結果を**図 14** に示す．同図は，Adap-RM + replica write-back および Static-RM + replica write-back の結果を，NORMAL + replica write-back の結果により正規化した値を示す．横軸は，同時に利用したワークロード日数，つまりワークロードの大きさを表す．各棒は，実験開始後 66 分，68 分，70 分後の 3 つの異なるマイグレーションスタート時間を用いた場合の実験の平均である．実験開始後 75 分後ごろに比較的大きなアクセス傾向の変化があり，その 10 分程度前を負荷均衡化開始時点とした．しかし負荷均衡化開始時点によって，結果が異なったため，複数の実験開始時点での結果の平均で評価を行った．

図より，Static-RM では，人工的ワークロードと異なりあまり改善が見られないことが分かる．これは，ファイルサーバワークロードではアクセス傾向の短期的な変化が激しいためであると考えられる．負荷評価時にマイグレーション移動元ノードを静的に設定する

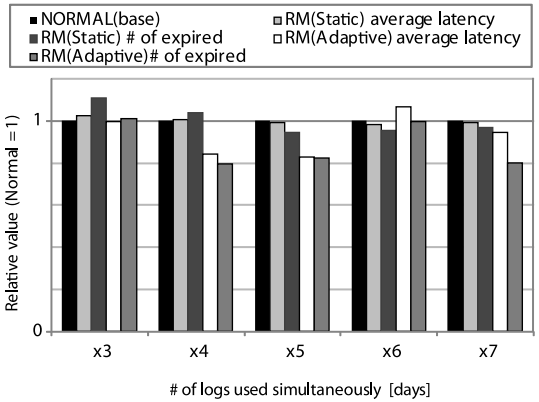

図 14　ファイルサーバワークロードにおける Replica write-back 各手法の性能（NORMAL を 1 とした相対値）
Fig. 14　Performance result of replica write-back method on the experiments with file server workload (Normalized by the value of the NORMAL).

Static-RM では，その後の負荷変化に対応できないため，マイグレーション経路変更が性能を悪化させる場合も存在する．

一方多くの場合で Adap-RM が応答性能においても超過レスポンス数についても削減されていることが分かる．しかしながら，削減率は約 20%と，人工的ワークロードによる実験の結果よりも小さい．この理由の 1 つに，ファイルサーバワークロードが Zipf ワークロードに比べアクセス集中が少ない点があげられる．よって，超過リクエストの増加が負荷の増加に対して大きくない．それに加え，ワークロードの傾向変化の度合いが Zipf に比べ小さい．

もう 1 つの理由として，ワークロード変化の予測が難しく，データマイグレーション戦略がワークロード変化を正確にとらえて立案されていない点があげられる．図 14 では 6 日分の log を用いた実験では RM 手法が有効に動作していないのが分かる．実験結果の解析により，この実験において立案されたマイグレーション戦略の誤りによるものであることが分かった．7.4 節における実験の結果のとおり，RM 手法を用いるとマイグレーション時間が削減される．その結果，負荷均衡化プロセスは NORMAL や Static-RM とは異なるタイミングで次のデータマイグレーションのための負荷評価を行う．同実験では，この異なるタイミングでのマイグレーション評価後に大きなワークロード傾向の変化が起こってしまったため，結果的に誤ったデータマイグレーションが性能を低下させてしまった．このように，

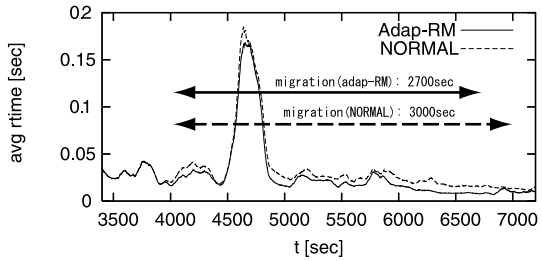

図 15　平均応答性能の時間変異
Fig. 15　Transitional average respons time.

負荷評価・予測とマイグレーション戦略アルゴリズムの精査は重要な課題である．

しかしながら，適応的経路選択と replica write-back を併用した RM 手法により，更新リクエストが含まれるファイルサーバ環境下においても，超過レスポンス数を 20%削減しつつデータマイグレーションを行うことが可能であり，手法の有用性が示された．**図 15** は 7 日間のトレースログを同時に用い，68 秒後に負荷均衡化を開始した実験における 1 分あたりの平均応答性能の時間変化を表した図である．図のように，Adap-RM はマイグレーション期間中，NORMAL よりも良い応答性能を示しており，かつマイグレーション時間を 300 秒短縮している．図 15 では，$4{,}500 < t < 5{,}000$ においていずれの手法にも大きな応答性能の悪化が見られる．これはワークロード中に過去の傾向を逸脱した突発的で大きなアクセスが含まれていたためである．現実的アクセスは先述の負荷予測の難しさに加え，このような突発的なアクセスも含まれる．今回対象としているデータマイグレーションによる負荷均衡化やその改善である提案手法では，負荷均衡化に比較的長い時間を要するため，このような短期的な負荷変動には対応できない．より堅牢なストレージシステム実現のためにはこのような突発的負荷上昇への対応が課題である．

9. おわりに

高い可用性と低い管理コストが求められるストレージシステムにおいて，データマイグレーションによる負荷分散およびマイグレーション速度維持と，クライアントアクセスに対する応答性能の維持の両立は重要である．本論文では，更新リクエスト扱いの効率化に特に着目し，既存手法である複製を利用した応答性能維持手法に対し，複製のみライトバックキャッシュ，経路選択の適応的制御の 2 つを加えることで，この目的を達成した．実験では，更新を含む Zipf ベースの人工的ワークロード，ファイルサーバ利用を想定した現実的なワークロードいずれに対しても，提案手法によりマイグレーション時間の削減，応答性能の維持を達成できたことを確認した．

今後の課題として，データマイグレーション戦略自体の改良，負荷評価・予測アルゴリズムの精査があげられる．また，より多くの適応対象，たとえば OLAP やビデオ配信サーバ，あるいは科学計算アプリケーション等で提案手法を用いた場合の有効性の評価を検討している．より堅牢なストレージシステム実現のためには，データマイグレーションが対象とする長期的な負荷変動による負荷集中に加え，短期的で予測の難しい突発的負荷上昇への対応も重要な課題である．以上の課題を解決したうえで，堅牢でディペンダブルな並列ストレージシステムの構築に関する検討を進めていきたい．

謝辞　本研究の一部は，独立行政法人科学技術振興機構戦略的創造研究推進事業 CREST「ディペンダブルで高性能な先進ストレージシステム」，独立行政法人日本学術振興会科学研究費補助金（特別研究員奨励費），情報ストレージ研究推進機構（SRC），文部科学省科学研究費補助金特定領域研究（18049026，19024028）および東京工業大学 21 世紀 COE プログラム「大規模知識資源の体系化と活用基盤構築」の助成により行われた．

参 考 文 献

1) Cluster File Systems Inc.: Lustre: A Scalable, High-Performance File System, Technical Report, Cluster File Systems Inc. (2002).
2) Dasgupta, K., Ghosal, S., Jain, R., Sharma, U. and Verma, A.: QoSMig: Adaptive Rate-Controlled Migration of Bulk Data in Storage Systems., *The 21st International Conference on Data Engineering* (*ICDE2005*), pp.816–827 (2005).
3) Feelifl, H., Kitsuregawa, M. and Ooi, B.: A Fast Convergence Technique for Online Heat-balancing of Btree Indexed Database over Shared-nothing Parallel Systems, *11th Int'l Conf. on Database and Expert Systems Applications*, pp.846–858 (2000).
4) Frølund, S., Merchant, A., Saito, Y., Spence, S. and Veitch, A.: FAB: Enterprise storage systems on a shoestring, *HOTOS 2003*, Kauai, HI (2003).
5) Ganger, G.R., Strunk, J.D. and Klosterman, A.J.: Self-* Storage: Brick-based Storage with Automated Administration, Technical Report CMU-CS-03-178, Carnegie Mellon University (2003).
6) Ghemawat, S., Gobioff, H. and Leung, S.-T.: The Google file system, *SOSP*, Boton Landing, NY, pp.29–43 (2003).
7) Hitachi Global Storage Technologies: Deskstar T7K500 Hard Disk Drive Specification ver.1.2 edition (2006). http://www.hitachigst.com

8) Hsiao, H.-I. and DeWitt, D.J.: Chained Declustering: A New Availability Strategy for Multiprocessor Database Machines, *Proc. 6th International Conference on Data Engineering*, Los Angeles, CA, pp.456–465, IEEE Computer Society (1990).
9) Hsu, W.W. and Smith, A.J.: Characteristics of I/O traffic in personal computer and server workloads, *IBM Systems Journal*, Vol.42, No.2, pp.347–372 (2003).
10) Kistler, J.J. and Satyanarayanan, M.: Disconnected operation in the Coda File System, *ACM Trans. Comput. Syst.*, Vol.10, No.1, pp.3–25 (1992).
11) 小林　大，渡邊明嗣，田口　亮，上原年博，横田治夫：データ移動コストとキャッシュを考慮した複製へのアクセス分散制御，日本データベース学会 Letters，Vol.4, No.1, pp.125–128 (2005).
12) 小林　大，渡邊明嗣，山口宗慶，田口　亮，上原年博，横田治夫：複製データを併用した効率的なデータマイグレーションの検討，日本データベース学会 Letters，Vol.3, No.2, pp.65–68 (2004).
13) Kobayashi, D. and Yokota, H.: Comparison of Replica Assisting and Speed Adjustment for Service-aware Horizontal Data Migration on Autonomous Disks, *International Workshop on Advanced Storage Systems 2007* (*ADSS 2007*) (2007).
14) Lu, C., Alvarez, G.A. and Wilkes, J.: *Aqueduct:* online data migration with performance guarantees, *Conference on File and Storage Technologies* (*FAST'02*), Monterey, CA, pp.219–230 (2002).
15) Lumb, C.R., Golding, R. and Ganger, G.R.: DSPTF: Decentralized Request Distribution in Brickbased Storage Systems, *Proc. ASPLOS'04*, Boston, MA (2004).
16) Lumb, C.R., Merchant, A. and Alvarez, G.A.: Façade: Virtual Storage Devices with Performance Guarantees, *2nd USENIX Conference on File and Storage Technologies* (*FAST'03*), pp.131–144 (2003).
17) Nagle, D., Serenyi, D. and Matthews, A.: The Panasas ActiveScale Storage Cluster: Delivering Scalable High Bandwidth Storage, *Proc. 2004 ACM/IEEE Conference on Supercomputing* (*SC '04*), Washington, DC, USA, p.53, IEEE Computer Society (2004).
18) Saito, Y. and Levy, H.M.: Optimistic Replication for Internet Data Services, *Distributed Computing, 14th International Conference* (*DISC*), Toledo, Spain, pp.297–314 (2000).
19) Scheuermann, P., Weikum, G. and Zabback, P.: Data Partitioning and Load Balancing in Parallel Disk Systems, *VLDB J.*, Vol.7, No.1, pp.48–66 (1998).
20) Storage Systems Lab, Hewlett-Packard Labs.: Publicly-available storage traces ("File System Traces"). http://tesla.hpl.hp.com/public_software/
21) Weikum, G., Mönkeberg, A., Hasse, C. and Zabback, P.: Self-tuning Database Technology and Information Services: from Wishful Thinking to Viable Engineering, *VLDB*, pp.20–31 (2002).
22) Wolf, J.L., Yu, P.S. and Shachnai, H.: Disk Load Balancing for Video-On-Demand Systems, *Multimedia Systems*, Vol.5, No.6, pp.358–370 (1997).
23) Yokota, H.: Autonomous Disks for Advanced Database Applications, *Proc. International Symposium on Database Applications in Non-Traditional Environments* (*DANTE'99*), pp.441–448 (1999).
24) Zhang, J., Sarkar, P. and Sivasubramaniam, A.: Achieving completion time guarantees in an opportunistic data migration scheme, *SIGMETRICS Perform. Eval. Rev.*, Vol.33, No.4, pp.11–16 (2006).
25) 原　隆浩，春本　要，塚本昌彦，西尾章治郎，奥井　順：広帯域ネットワーク上のデータベース移動に基づく動的複製配置法，電子情報通信学会論文誌，Vol.J-82-D-I, No.8, pp.1049–1058 (1999).

付　　録

A.1　データ喪失確率の計算

5.1 節で用いたデータ喪失確率の計算過程を示す．なお，各変数の定義は 5.1 節を参考にされたい．

まず，いずれのデータもライトスルーで永続メディアに書き込む場合のデータ喪失を考える．データが失われるのは，(あるストレージノードが故障）かつ（当該ノード上のデータの複製を保持するノード（以下複製保持ノード）が，冗長性を復旧するため他のノードにデータを複製する（以下，冗長度回復）前に故障）のときである．ノード数 N のため，データ喪失確率 P_{TT} は式 (2) となる．

$$\begin{aligned} P_{TT} &= N \times (P_S \times P_S t_R + O({P_S}^3)) \\ &\approx N {P_S}^2 t_R \end{aligned} \tag{2}$$

つづいて，複製データのみライトバックキャッシュ上に更新を保持する場合のデータ喪失を考える．データが失われるのは，((あるノードが故障）かつ（(複製保持ノードが冗長度回復前に故障）または（複製保持ノードがキャッシュ上の更新データを永続メディアに書き戻す（以下キャッシュ書き戻し）前に複製保持ノードの電源が故障)))または（(あるエンクロージャの電源が故障）かつ（複製更新のキャッシュ書き戻し前にそのシャーシ中の n_e 個のノードすべての複製保持ノード n_e 個のいずれかが壊れる)）ときである．ノード数 N，エンクロージャ数 N/n_e より，データ喪失確率 P_{TB} は式 (4) となる．

$$\begin{aligned} P_{TB} &= N \times \left\{ P_S \times (P_S t_R + P_P t_c + O({P_S}^2) + O(P_P P_S) + O({P_P}^2)) \right\} \\ &\quad + N/n_e \times P_P \times (n_e P_S t_c + O(P_S^2)) \\ &\approx N {P_S}^2 t_R + 2 N P_S P_P t_c \end{aligned} \tag{3}$$

最後に，いずれのデータもライトバックキャッシュ上に更新を保持する場合のデータ喪失を考える．データが失われるのは，((あるノードが故障）かつ（(複製保持ノードが冗長度回復前に故障）または（キャッシュ書き戻し前に複製保持ノードの電源が故障)))または((あるエンクロージャの電源が故障）かつ（(キャッシュ書き戻し前にそのシャーシ中の n_e 個のノードすべての複製保持ノード n_e 個のいずれかが壊れる）または（複製，プライマリ両更新のキャッシュ書き戻し前にそのシャーシ中の n_e 個のノードすべての複製保持ノード n_e 個の電源)))または（システムの電源供給が故障する）ときである．ノード数 N，エンクロージャ数 N/n_e より，データ喪失確率 P_{BB} は式 (4) となる．

$$\begin{aligned} P_{BB} &= N \times \left\{ P_S \times (P_S t_R + P_P t_c + O({P_S}^2) + O(P_P P_S) + O({P_P}^2)) \right\} \\ &\quad + N/n_e \times P_P \times (P_P t_c + n_e P_S 2t_c + O({P_S}^2) + O(P_S P_P) + O({P_P}^2)) + P_U \\ &\approx N{P_S}^2 t_R + 3N P_S P_P t_c + N{P_p}^2 t_c/n_e + P_U \end{aligned} \tag{4}$$

(平成 19 年 10 月 7 日受付)
(平成 20 年 3 月 4 日採録)

小林　大

2003 年東京工業大学工学部情報工学科卒業．2005 年東京工業大学大学院情報理工学研究科計算工学専攻修士課程修了．現在，同専攻博士後期課程在学中．2006 年より日本学術振興会特別研究員 DC．並列ストレージシステム，データ工学等の研究に従事．日本データベース学会学生会員．

横田　治夫（正会員）

1980 年東京工業大学工学部電子物理工学科卒業．1982 年東京工業大学大学院理工学研究科情報工学専攻修士課程修了．同年富士通（株）．同年 6 月（財）新世代コンピュータ技術開発機構研究所（ICOT）．1986 年（株）富士通研究所．1992 年北陸先端科学技術大学院大学情報科学研究科助教授．1998 年東京工業大学大学院情報理工学研究科助教授．2001 年東京工業大学学術国際情報センター教授．工学博士．主として分散インデキシング，データ工学向けアーキテクチャ，高機能ストレージシステム，ディペンダブルシステム等に関する研究に従事．電子情報通信学会フェロー．日本データベース学会理事．人工知能学会，IEEE，ACM 各会員．