社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2009－DPS－138 (2)
2009－CSEC－ 44 (2)
2009／3／5

多段キャッシュ型ネットワークストレージへのアクセスの時間的局所性を考慮したメモリキャッシュ制御

宮野晋平† 山口実靖‡ 淺谷耕一†‡

†工学院大学大学院 工学研究科 電気・電子工学専攻 ‡工学院大学 工学部 情報通信工学科

ネットワークストレージを用いることによりユーザが使用するデータを集約して一元管理することが可能となる．これによりデータ管理コストが大幅に削減されるため，ネットワークストレージは多くの組織で使用されるようになった．ネットワークストレージ上のデータへのアクセスは，サーバ計算機上のキャッシュメモリとストレージ機器上のキャッシュメモリの二重のキャッシュを介して行われる．我々はまず，二重構造キャッシュ使用時における両キャッシュのデータの重複度，キャッシュのヒット率について調査を行った．結果，多重構造キャッシュに起因する負の参照の時間的局所性が存在し，それによりキャッシュヒット率が極めて低いことが分かった．そして，LRU を用いない解決手法の性能評価を行い，これによりキャッシュヒット率と性能が改善されることを確認した．

A Study of Negative Temporal Locality of Reference on Network Storage System

Shimpei Miyano† Saneyasu YAMAGUCHI‡ Koichi ASATANI†‡

†Graduate School of Electrical and Electronic Engineering, Kogakuin University ‡Kogakuin University

Storage management cost is one of the most import issues of recent computer systems. Storage consolidation using network storages remedy this problem, thus they are widely deployed. Accesses to data in network storages are executed through a cache memory in a server computer and one in a storage appliance. We explored temporal locality of reference in network storage, and found that cache hit ratio is severely low according to negative temporal locality of reference, which is caused by these two caches. We discussed methods to utilize these caches, and demonstrated that cache controlling based on this locality could provide better hit ratio and performance than cache replacement using LRU.

1. はじめに

近年，ストレージの大容量化が進み，計算機システムの扱うデータ量が爆発的に増加している．それに伴いバックアップなどのストレージ管理コストが増大し，計算機システムの大きな問題の一個となっている．この問題に対してSAN(Storage Area Network)や NAS(Network Attached Storage)を用いてネットワーク上にストレージを配置し，データを集約する手法が提案された．ユーザが使用するデータを集約し一元管理することによりストレージ管理コストの大幅な削減が可能であり，ネットワークストレージは多くの組織で使用されるようになった．

ネットワークストレージ上のデータへのアクセスは，サーバ計算機上のキャッシュとストレージ機器上のキャッシュの二層のキャッシュを介して行われる．このような二重のキャッシュ構造においては，両キャッシュに同一のデータが重複して格納される可能性が考えられる．通常の階層化されたキャッシュシステムでは階層間に非常に大きな容量の違いがあるため，上位層キャッシュにてミスヒットしたデータアクセスが下位層キャッシュにてヒットする確率は低くない．しかし，サーバ計算機とストレージ機器のキャッシュサイズには極端な差が無い場合が多く，両キャッシュの保持データがほぼ同一となると，サーバ計算機のキャッシュにてミ

スヒットしたデータアクセスはストレージ機器のキャッシュにても高確率でミスヒットとなり，ストレージ機器のキャッシュが効果的に機能しないことが考えられる．これをストレージ機器の視点で見ると，一度アクセスされたデータへの再アクセス要求は（サーバ計算機のキャッシュにて処理されるため）近い将来にはストレージ機器に送られないことを意味し，逆向きのデータ参照の時間的局所性が存在することを意味する．

本稿ではこの多重キャッシュに起因する負の時間的局所性とそれによる低いキャッシュヒット率の問題を示し，その解決手法と性能評価について述べる．

2. ネットワークストレージのキャッシュ処理

2.1. ネットワークストレージ

記憶装置として DAS（Direct Attached Storage：計算機に直接接続された通常のストレージ）を用いた場合，データのバックアップを行うには各個人の計算機のデータを個別にバックアップしなくてはならない．これに対してネットワークストレージを使用した場合は，各計算機に分散していたデータをネットワーク上で集約し管理できるため，管理者は集中管理が可能であり，大幅な管理コストの削減が可能となる．

ネットワークストレージは主に，ファイルレベルプロトコルでアクセスされる NAS と，SAN を介してブロックレベルプロトコルでアクセスされるストレージに分類される．前者の場合，ファイルシステム機能も含めてストレージ機器内に存在し，ファイルの共有が可能，導入が容易などの利点がある．通常，ネットワークとしては TCP/IP と Ethernet が，アクセスプロトコルとしては NFS や CIFS が使用される．後者では，ファイルシステム機能はサーバ計算機上に存在し，ストレージ機器はブロックレベルの処理のみを行う．よって，より高い性能が得られると期待される．ネットワークとブロックレベルプロトコルとしては FC (Fibre Channel) と SCSI や， TCP/IP over Ethernet と iSCSI などが使用される．

2.2. 参照の局所性

多くのアプリケーションプログラムでは，メモリやストレージへのデータ参照の分布は一様の分布ではなく，偏りのある局所化された参照が行われるとされている[1]．具体的には，多くのプログラムにおいて参照の時間的局所性と空間的局所性が存在するとされている．時間的局所性とは最近参照されたデータが近い将来に再度参照される可能が高いことを意味し，空的局所性とはある参照されたデータの近傍のデータが近い将来に参照される可能性が高いことを意味する．

多くのコンピュータシステムのアプリケーションにおいて参照の時間的局所性が存在することが確認されており，ほとんどの OS やコンピュータシステムにおいてキャッシュの置換アルゴリズムには LRU(Least Recently Used) [1]が採用されている．

2.3. ネットワークストレージにおける二重ディスクキャッシュ構造

ネットワークストレージ環境におけるストレージアクセス処理について説明する．サーバ計算機内のアプリケーションプログラム(以下，単に“アプリケーション”と記す)からのネットワークストレージ上のデータへのアクセスは，サーバ計算機上のキャッシュと，ストレージ機器内のキャッシュの二重のキャッシュを介して行われる．一般に両キャッシュともそれぞれの機器の OS の機能として提供され，それぞれの機器の主記憶（半導体メモリ）を用いて実現される．

以下に，ブロックレベルネットワークストレージとファイルレベルネットワークストレージにおけるキャッシュ機能の詳細について述べる．iSCSI SAN などのブロックレベルアクセスのネットワークストレージの場合は，ファイルシステム，ブロックデバイス機能がサーバ計算機側に存在し，これらがサーバ計算機内にてブロックデータのキャッシュ，先読み機能を提供する．NFS などのファイルレベルプロトコルによりアクセスする NAS を用いる場合，サーバ計算機のファイルシステムクライアントにキャッシュ機能が存在し，これがサーバ計算機の主記憶を用いてストレージへのキャッシュを実現する．ストレージ機器内のディスクキャッシュ機能はその機器の実装に依存するが，多くの場合ストレージ機器は CPU を搭載し OS が稼動している計算機により実現されており，組み込まれた OS が搭載されている主記憶を用いてディスクに対するキャッシュや先読みを行う．HDD デバイスにもキャッシュ用メモリが搭載されているが，本稿で主に考察する計算機主記憶を用いるキャッシュと比較し規模が非常に小さく影響も限定的となるため，本稿では考察の対象としない．

2.4. 二重キャッシュの動作

次に，ブロックレベルプロトコルアクセスのネットワークストレージでのデータアクセス動作を基に，データ参照要求発生時のサーバ計算機とストレージ機器のキャッシュの動作について説明する．動作は図 1 の**“(A)サーバ計算機キャッシュヒット”，“(B)ストレージ機器キャッシュヒット”，“(C)HDD デバイスアクセス”**の三種に分類される．ファイルレベルアクセスのネットワークストレージの場合も，アクセスプロトコルのレイヤは異なるが，その動作は類似のものとなる．

サーバ内のアプリケーションからストレージ上のファイルに対する参照要求が発行されると，参照要求を受け取ったサーバ計算機の OS のキャッシュ機能が自機の主記憶内に該当データが存在するか否かを調査する．存在する場合は，サーバ計算機の OS が該当データをアプリケーションに返す．これが(A)サーバ計算機キャッシュヒットとなる．ブロックレベルプロトコルを用いている場合，I/O 要求はサーバ計算機内部で処理され，ストレージ機器に要求や情報は転送されない．

サーバ計算機の主記憶内に該当データが存在しない場合は，サーバ計算機はネットワークを介してストレージ機器に参照要求を送信する．参照要求を受信したストレージ

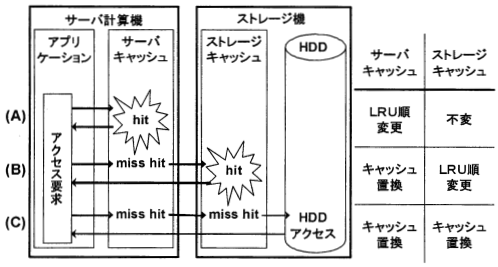

図 1　キャッシュ動作

機器の OS は，自機の主記憶内に該当データが存在するか否かを調査する．存在する場合はストレージ機器の OS が該当データをサーバ計算機に返し，データを受信したサーバ計算機の OS がアプリケーションにデータを返す．これが(B)ストレージ機器キャッシュヒットであり，ストレージ機器の HDD デバイスへの物理的なアクセスは伴わない．

ストレージ機器の主記憶内に該当データが存在しない場合は，ストレージ機器の OS は HDD デバイスに物理的にアクセスし，要求されたデータを取得する．そして，そのデータがサーバ計算機に転送され，アプリケーションに渡される．これが(C)HDD デバイスアクセスとなる．

アクセス時間は短い順に，(A)サーバ計算機キャッシュヒット，(B)ストレージ機器キャッシュヒット，(C)HDD デバイスアクセスとなり，(A)が多く(C)が少ないことが好ましい．

次に，(A)～(C)の各動作の発生時の両キャッシュ状態の変化，ストレージ機器のキャッシュの重複度の変化，各動作の発生確率に関する考察を行う．ただし，両キャッシュの置換アルゴリズムとしては LRU が使用されているとする．また，ストレージ機器のキャッシュの重複度とは，両キャッシュに登録されているデータが同一であるか否かを示す指標であり，ストレージ機器のキャッシュに登録されているデータがサーバのキャッシュにも登録されている確率のこととする．

まず，(A)サーバ計算機キャッシュヒットが発生した場合について述べる．この場合，サーバ計算機内のキャッシュの LRU の順のみが更新され，キャッシュの置換は発生しない．ストレージ機器のキャッシュに関しては一切の変化が発生しない．重複度にも変化はない．次に(B)ストレージ器キャッシュヒットが発生した場合について述べる．この場合，ストレージ機器のメモリからサーバ計算機にデータが転送され，そのデータがサーバ計算機の主記憶内にキャッシュされる．よって，サーバ計算機においてはデータの主記憶への新規の追加と同サイズのデータの削除が行われ，ストレージ機器ではキャッシュ LRU 順の更新が行われる．該当データが両キャッシュに同一の状態で登録されるため，このデータに関しては両キャッシュの重複度を増加させることになる．削除されるデータに関してはそのデータが重複データであったか否かにより分れ，重複データであった場合は重複度が減少し，そうでなかった場合は重複度は変化しない．以上まとめると，削除データが重複データであった場合は同サイズの重複増加(新規登録)と減少(削除)が発生し重複度は変化せず，削除データが重複データでなかった場合は重複度増加(新規登録)のみが発生し重複度は増える．最後に，(C)HDD デバイスアクセスについて述べる．この場合は，HDD デバイスから読み込まれた該当データがストレージ機器の主記憶とサーバ計算機の主記憶に新規に追加されることとなる．追加の登録であるため，両キャッシュにて LRU 順で最古のデータが削除される．新規に登録されたデータに関しては，両キャッシュの重複度を高める．それぞれのキャッシュにて削除されるデータに関しては，削除対象のデータがもう片方のキャッシュに存在している場合と存在していない場合に分けられるが，存在していない場合は重複度に変化は無く，存在している場合は重複度を下げることになる．削除データが両方とも重複データであった場合は，新規登録データの倍のサイズの重複データが削除されるため結果的に重複度が減少する．

以上より，両キャッシュデータの重複度が減少するのは HDD デバイスアクセス時において両削除対象データが重複データであった場合のみであり，両キャッシュのデータ内容は常に重複率の高い状態になると考えられる．よって，サーバ計算機とストレージ機器のキャッシュ容量が同程度の場合には，サーバ計算機でキャッシュミスヒットしたデータアクセスはストレージ機器のキャッシュでも必ずミスヒットすると予想される．ストレージ機器内である時刻に参照されたデータが近い将来に再度アクセスされる可能性は(それ以外のデータと比較して)低く，通常とは逆向きの負の参照の時間的局所性が存在していると考えられる．そのため，LRU によるキャッシュの置き換えを行うとキャッシュヒット率が極めて低くなると考えられる．

3. 二重キャッシュ構造の効率の評価

3.1. ヒット率

二重キャッシュ構造における両キャッシュの効果を調査するため，シミュレーションにより評価を行なった．シミュレーション条件は，データ領域サイズが 10,000 ブロック，サーバ計算機キャッシュサイズが 100 または 1000 ブロック，サーバ計算機キャッシュとストレージ機器キャッシュのサイズの比が1対0．5～1対4，アクセスパターンが一様分布ランダム，参照試行回数が 1,000,000 回とした．ただし，データ領域サイズとはユーザからアクセスされうるストレージ全体のサイズであり，ブロックとはキャッシュ置換が行われる最小単位である．

シミュレーション結果を図 2，図 3 に示す．図 2 ではサーバ計算機のキャッシュ容量は 100 であるため，一様分布アクセスに対するサーバ計算機でのキャッシュヒット率は 1%となっている．同様に，図 3 ではサーバ計算機のキャッシュヒット率は 10%となっている．これに対してストレージ機

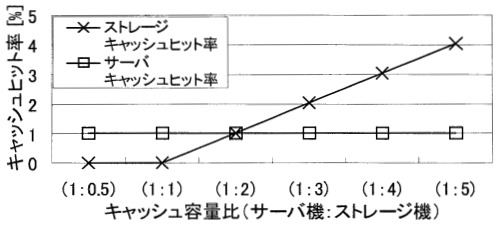

図2　キャッシュヒット率（一様分布）
(サーバ機キャッシュ容量：100)

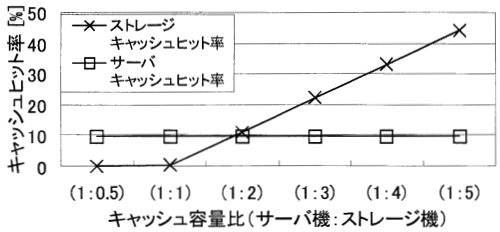

図3　キャッシュヒット率（一様分布）
(サーバ機キャッシュ容量：1000)

器のキャッシュヒット率は低くなっていることが分かる．特に，サーバ計算機のキャッシュ容量がストレージ機器のキャッシュ容量の同等以上の場合には，ストレージのキャッシュヒット率が極めて低くなっている．また，ストレージ機器のキャッシュ容量をサーバ計算機の 2〜5 倍としても，1〜4 倍から期待されるヒット率しか得られていないことが分かる．すなわち，ストレージ機器キャッシュのうちサーバ計算機キャッシュの1倍分のサイズは有益な効果を生めていないことが分かる．ストレージ機器は複数のサーバ計算機からアクセスされることも多く，その場合，相対的なストレージ機器の搭載メモリ量は減少しヒット率はさらに減少すると考えられる．また参考のために，アプリケーションから発行される参照要求に偏りを持たせた場合の測定結果を付録に記す．

3.2. キャッシュ重複率

メモリキャッシュ容量比が(1:0.5)(1:1)(1:2)(1:3)(1:4)の場合におけるキャッシュデータの重複率を図 4 に示す．本結果より，ストレージ機器キャッシュサイズがサーバ計算機キャッシュサイズの同等以下の場合は，ストレージ機器のキャッシュの重複率がほぼ 100%であることがわかる．つまり，ストレージ機器キャッシュに保存されているデータはサーバ計算機のキャッシュにも保持されており，サーバ計算機でキャッシュミスヒットが発生すると，ストレージ機器のキャッシュでもほぼ必ずミスヒットとなる．また，サイズ比が(1:2), (1:3), (1:4)の場合の重複率がほぼ 1/2, 1/3, 1/4 となっていることから，サーバ計算機キャッシュのデータのほぼ

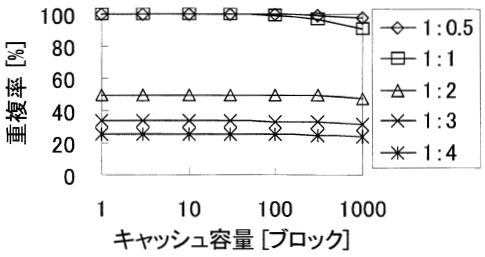

図4　キャッシュデータ重複率

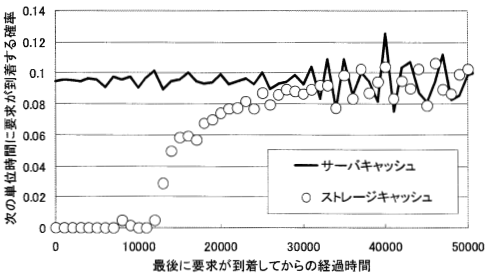

図5　アクセス後経過時間とアクセス確率の関係

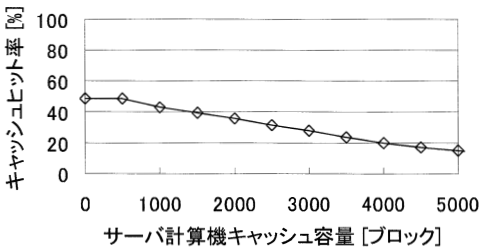

図6　ストレージキャッシュヒット率

すべてがストレージ機器キャッシュにも重複して登録されていることが分かる．

3.3. 負の参照の時間的局所性

次に参照の局所性について考察する．アプリケーションの要求発行が一様分布，データ領域サイズ 10,000，両キャッシュサイズ 5,000 における「最後に要求が到着してからの経過時間」と「次の単位時間に要求が到着する確率」の関係を図 5 に示す．マルコフ過程であるためサーバ計算機における到着確率は均等となったが，ストレージ機器における到着確率は，経過時間が短いアドレスのみ低くなった．これより，アプリケーションから時間的局所性の無い参照要求が発行されても，サーバ計算機のキャッシュを経ることにより，ストレージ機器にとどく要求には負の時間的局所性が生じることが分かる．ストレージ機器のキャッシュヒッ

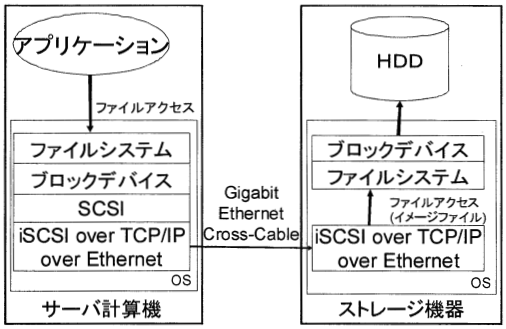

図 7　ネットワークストレージシステム

ト率を図 6 に示す．局所性が無ければヒット率 50%，正の局所性があれば 50%以上が期待されるが，50%未満となり負の局所性が確認された．

4. LRU を用いないキャッシュ制御

本章では，実際のネットワークストレージ環境における効果的なキャッシュの使用方法の考察と，評価を行なう．

4.1. 実験環境

実験環境として図 7 のネットワークストレージシステムを構築した．両実験機の仕様は OS Linux 2.6.18.8. (Fedora 7), CPU Intel Xeon 2.32GHz, Memory DDR2 ECC 1GB, HDD WESTERN DIGITAL WD800AAJS-18 (80GB, SATA 接続, 7200 rpm, バッファ 8MB)である．ストレージ機器は 1.5[GB]のディスクイメージを提供する iSCSI ターゲットであり，サーバ機がこれに iSCSI を用いて接続する．iSCSI ターゲットは iSCSI Enterprise Target 0.4.12 を，iSCSI イニシエータは Open-iSCSI-2.0-869.2 を用いて構築した．ファイルシステムとブロックデバイスはサーバ計算機側に，HDD デバイスがターゲット側に存在し，両者間でブロックレベルのデータ送受信が行われる．iSCSI ターゲットはファイルモードで動作し，ストレージ機器のファイルシステム上に作成されたファイルがディスクイメージとなる．よって，ストレージ機器の OS のファイルシステムやブロックデバイスが「イメージファイルへのアクセス」を仲介し，結果としてディスクアクセスに対するキャッシュや先読みの機能を提供する．サーバ計算機ではファイルシステムやブロックデバイスが「iSCSI ディスクへのアクセス」に対するキャッシュや先読みを提供する．両計算機は 1[GB]のメモリを積載しており，このうち約 800[MB]がディスクキャッシュに利用される．実験では，iSCSI ディスク(1.5GB)内に，1[MB]のファイルを 1400 個（ファイル名 0～1399，総容量 1.4[GB]）作成し，これらのファイルに対して一様分布ランダムに読込処理を 10000 回行った．

4.2. メモリ固定割付け

参照の時間的局所性を用いない(LRU でない)制御として，ディスク内の固定領域をメモリ内にキャッシュする手法について評価する．具体的にはストレージ機器にてディスクイメージの先頭 700[MB]に継続的にアクセスし，LRU キャッシュアルゴリズムにより置換されないように制御した．ディスクイメージの先頭 700[MB]はファイル 0～700[MB]の存在位置とほぼ一致する．参照の時間的局所性を用いた通常の LRU ではストレージ機のメモリキャッシュは完全にミスヒットするが，制御を行なうことにより 700[MB]/1.4[GB](2 分の 1)の確率でキャッシュがヒットすることが期待できる．ただし，本制御そのものに負荷があるためキャッシュヒット率に改善が見られない場合は性能を下げることとなる．キャッシュ制御を行なう手法を“制御あり”，制御を行なわない手法を“制御なし”と記す．

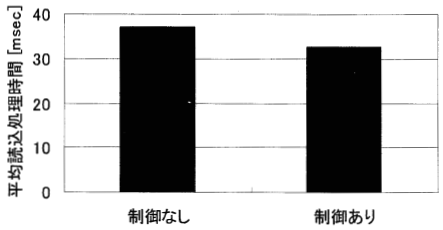

図 8　各読込処理時間

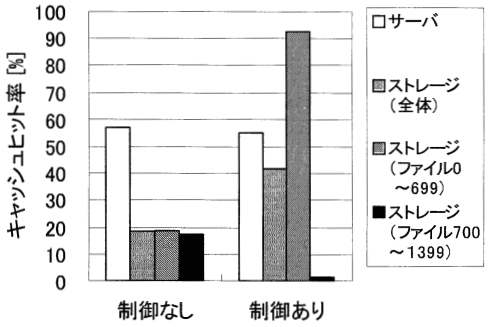

図 9　キャッシュヒット率

4.3. 性能評価

性能計測結果を図 8 に，キャッシュヒット率を図 9 に示す．図より本実験環境の例では，キャッシュがほぼ全てミスヒットとなる LRU よりも，固定的に割り当てる手法の方が約 13%性能が優れることが確認された．

次に，両手法の各読込の個別の処理時間を図 10,11 に示す．図 10,11 において，読み込み時間が約 1ms の処理と，9ms の処理が多数存在しているのが，それぞれサーバ計算機キャッシュヒット，ストレージ機器キャッシュヒットである(1[MB]ファイルを速度 1[Gbps]で転送するのに約 8[ms]要する)．図 9 よりストレージ機器でのキャッシュヒット率は制御を行わない状態では極めて低いこと，制御を行うことにより制御対象のファイル(0～699)のキャッシュヒット率が大幅に改善され，結果として全体のキャッシュヒット率も大きく改善されることがわかる．

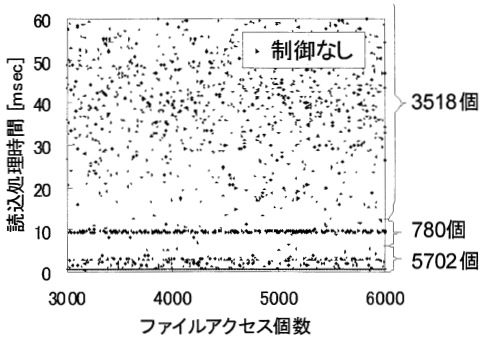

図 10　各読込処理時間（制御なし）

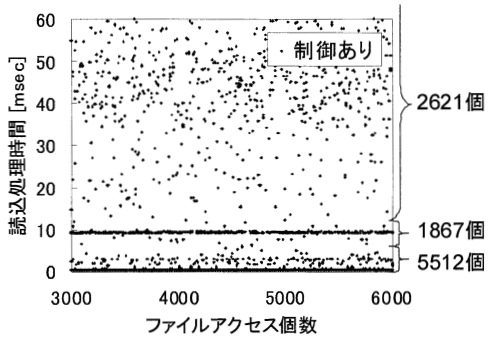

図 11　各読込処理時間（制御あり）

5. まとめ

本稿では多重キャッシュに起因する負の参照の時間的局所性とそれによる低いキャッシュヒット率の問題をシミュレーションにより示した．そして解決手法について考察し，評価実験を行い，性能向上が可能であることを示した．

今後は，サーバ計算機とストレージ機器のキャッシュが重複しない方法の考察，固定割付よりもヒット率が高い方法の考察，アプリケーションが発行する参照要求に偏りが有る場合の考察，書込処理に関する考察，キャッシュ置換手法のOSへの実装，既存のOSのランダムキャッシュ置換手法との比較などを行う予定である．

参考文献

[1] アンドリュー・S. タネンバウム，“オペレーティングシステム 設計と実装”，ピアソン・エデュケーション，2007 年 12 月．

[2] 田中 淳裕，“メモリ参照の局所性に関する定量的な評価”，情報処理学会研究報告，ハイパフォーマンスコンピューティング，Vol.96, No.81 pp. 39-44, 1996.

[3] 田中 淳裕，“メモリ参照の局所性に着目したソフトウェア性能評価手法”，コンピュータソフトウェア，Vol.16, No.2(1999), pp. 32-46.

[4] P. Denning. Working sets past and present. IEEE Trans. on Software Engineering, SE-6(1):64 84, 1980.

[5] P. J. Denning, "The Working Set Model for Program Behavor, "CACM, Vol. 11, No. 5, pp, 323-333, May 1968.

[6] P. J. Denning and S. C. Schwartz, "Properties of the Working Set Model, CACM, Vol. 15, No. 3, March 1972, pp. 191-198.

[7] Saneyasu Yamaguchi, Masato Oguchi, Masaru Kitsuregawa, "iSCSI Analysis System and Performance Improvement of iSCSI Sequential Access in High Latency Networks," In Proceedings of 12th IEEE International Conference on High Performance Computing (HiPC 2005)

付録

「平均がデータ領域サイズ/5 である指数分布」を確率密度関数とする参照要求発生分布の場合のキャッシュヒット率を図 12，13 に示す．両図より，アプリケーションの参照に偏りがある場合も同様に，ストレージ機のキャッシュヒット率が大きく低下することが分かる．

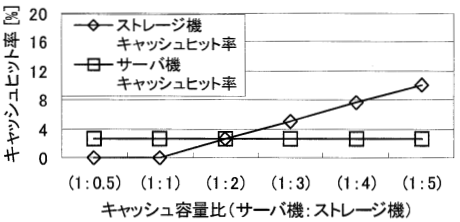

図 12　キャッシュヒット率（指数分布）
(サーバ機キャッシュ容量：100)

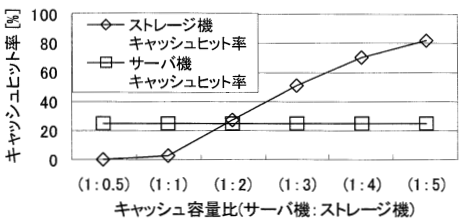

図 13　キャッシュヒット率（指数分布）
(サーバ機キャッシュ容量：1000)