様式 C-19

科学研究費補助金研究成果報告書

平成２３年　６月１７日現在

機関番号：３２６８９
研究種目：研究活動スタート支援
研究期間：2009～2010
課題番号： 21800061
研究課題名（和文）メニーコアＣＰＵ環境に適したアクセスコスト考慮型ファイルキャッシュ機構

研究課題名（英文）A File Cache Mechanism Considering Access Cost and Being Suitable for Many-core CPU Environment

研究代表者
上田　高徳（UEDA TAKANORI）
早稲田大学・メディアネットワークセンター・助手
研究者番号：90546863

研究成果の概要（和文）：
本研究では多数の計算コアを持つメニーコア CPU 環境に適切な、新しいファイルキャッシュアルゴリズムを開発することを目指した。新アルゴリズムは、アクセスコストが大きいデータを優先的にキャッシュすることで性能向上を試みる。アクセスコストとはセカンダリストレージから読み出すのに必要な時間のことであり、たとえばランダムアクセスされるデータはアクセスコストが高いといえる。関係データベースに対する標準的なベンチマークである TPC-H による評価では、実機上において 5%程度の高速化を達成できたが、実用に足ると考えられる数十%ないしは、数倍といったオーダでの高速化は達成できなかった。

研究成果の概要（英文）：
This research aims to develop a novel file cache algorithm suitable for many-core CPU environment that has many computation cores. The new method tries to cache high access cost data such as random accessed data that requires long time to be read from secondary storage devices. The evaluations by TPC-H, a de facto standard benchmark for DBMS, showed about five percent performance improvement on real computers. However, several dozen or several fold performance improvement enough for actual use was not achieved.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
2009 年度	1,070,000	321,000	1,391,000
2010 年度	970,000	291,000	1,261,000
年度			
年度			
年度			
総　計	2,040,000	612,000	2,652,000

研究分野：ストレージ・並列分散処理
科研費の分科・細目：「情報学」・「メディア情報学・データベース」
キーワード：キャッシュ・ストレージ・データベース・メニーコア

１．研究開始当初の背景

研究開始当初の 2009 年ごろは、CPU のマルチコア化が進み、Intel をはじめとする x86 アーキテクチャ CPU でも 4 コア CPU が一般的

になってきていた。周波数の向上によって処理性能を向上してきた時代は終わり、並列化により処理性能を向上させる時代となった。マルチコア化のトレンドが継続することは確実とみられ、1 つの CPU チップ上に数十から数百の計算コアを搭載したメニーコア CPU が登場し、CPU の並列処理能力が大幅に向上することが期待されていた。

メニーコア CPU の並列処理性能を活用することで、多数の計算機を用いなければ不可能だった大規模データ処理が、1 台の計算機で行えるようになると期待できた。しかし、このような大規模データ処理アプリケーションを滞りなく高速に動作させるためには、アプリケーションが要求するデータを高速に CPU に転送する必要がある。転送が間に合わない場合、データ転送が完了するまで CPU はアイドル状態になり、有益な処理を行うことができない。

このようなストレージボトルネックは、研究開始当初、そして現在でもますます深刻になると予想されている。なぜならば、メニーコア CPU の性能向上に対して、データを格納するセカンダリストレージであるハードディスク速度の向上が芳しくないからである。事実、CPU メーカの Intel 社は、1996 年から 2006 年の 10 年間で CPU の計算性能は 30 倍以上になったが、ハードディスクのアクセス性能は 1.3 倍程度の向上に留まっており、今後もこの傾向は続くとして問題視してきた。コア数の増加により、CPU とストレージ間の性能差がより開くことは疑いなく、ますますストレージボトルネックが深刻になると考えられる。本研究は、メニーコア CPU 環境におけるストレージボトルネックを軽減することを目標として開始された。

２．研究の目的

本研究の目的は「メニーコア CPU 環境に適したアクセスコスト考慮型ファイルキャッシュ機構」を開発することである。本研究のターゲットは、ランダムアクセスと排他制御が問題となるメニーコア CPU 環境における「ファイル」のキャッシュ手法である。

多くのオペレーティングシステムやアプリケーションが一度読み込んだファイルデータを物理メモリ上にキャッシュする機能を持っている。しかし通常、物理メモリは全データをキャッシュするには充分ではないため、どのデータをキャッシュに残すか選択するキャッシュ戦略が重要となる。これまで殆どの場合、LRU や 2Q といったキャッシュヒット率の向上を目標とするキャッシュ戦略が利用されてきた。

これに対して本研究は、ファイルのキャッシュ戦略においてはデータの「アクセスコスト」が重要であり、キャッシュヒット率を高めるのではなく、アクセスコストの総計を最小化する戦略が有効という立場をとる。ここで「アクセスコスト」とは、データにアクセスするために必要な実時間である。ストレージ、特にハードディスクに対するランダムアクセスは非常に低速であり、アクセスコストが高いといえる。よって、ランダムアクセスされるデータを優先的にキャッシュすることで、アクセスコストの総計を小さくできる。言い換えると、同じキャッシュヒットでも、シーケンシャルアクセスに対するヒットより、ランダムアクセスに対するヒットの方がシステムの実時間性能向上に大きな効果があるといえる。

メニーコア CPU 環境においては、アプリケーションの並列動作により、ファイルへの頻繁なランダムアクセスが発生する。よって、ランダムアクセスにおけるキャッシュミスがますます増え、アクセスコストの総計が増加し、深刻なストレージボトルネックが発生する。一方、キャッシュヒット率が高い場合は排他制御によるオーバーヘッドが問題になる。例えば、LRU リストをキャッシュ管理のデータ構造に用いると、LRU リストを更新する度にスレッド間の排他制御が必要となり、やはりボトルネックとなる。従ってメニーコア CPU 環境においては、同時実行性に優れたキャッシュ機構を用いると同時に、ヒット率を犠牲にしてでもランダムアクセスされるデータを優先的にキャッシュすることでストレージボトルネックを軽減し、システム性能の向上を達成できると考えられる。

よって本研究では、メニーコア CPU 環境におけるストレージボトルネックを軽減するため、キャッシュミスすると大きなアクセスコストが発生するファイルデータを優先的にキャッシュに残すことで総アクセスコストを削減し、システム性能を向上させることを目指した。この目標を実現するためには、ランダムアクセスでファイルから読み込まれるデータのように、アクセスコストが大きいデータを自動的に識別し、総アクセスコストをいかに小さくするかが課題となった。

３．研究の方法

実機で使える実用的なアルゴリズムの実現を目指し、理論的な考察によるアルゴリズムの検討と、開発したアルゴリズムの既存ソフトウェアへの実装という両面から研究を行った。

メニーコア CPU 環境では、多くのアプリケーションが並列に動作するため、従来の環境とは異なるファイルアクセス特性を示すと考えられる。そこでまず、ファイルアクセスログを採取して使用コア数に応じたアクセス特性を分析し、キャッシュアルゴリズム開発の参考にした。Linux のトレース機構を活

用して独自にトレースを取得すると共に、http://iotta.snia.org/ において公開されているファイルアクセスのトレースログも利用した。

収集したトレースログを利用してアクセスコストのモデル化を試みた。たとえば、ハードディスクのアクセスコストは、アクセスヘッドの移動距離で決まると考えられる。しかし、同じデータブロックに対するアクセスでも、直前のアクセスでどのブロックにアクセスしたかによってアクセスコストが変化するため、アクセスコストの見積もりは困難に思える。ただし、ファイルに対するアクセスにはアプリケーションが用いるデータ構造によって一定のアクセスパターンが生じる。たとえば木構造の場合、あるノードがアクセスされたあと子ノードがアクセスされる可能性が高い。このようにアクセス順序に一定のパターンがあると仮定し、各ファイルブロックを取得するのに掛った時間を実計測しておくことで、ブロックごとのアクセスコストの算出を目指した。

また、近年普及が進んでいる半導体ストレージである SSD による評価を行うために、PCI-Express 接続型の SSD を用いても実験を行った。SSD は稼働部品がないため、アクセスコストの特性がハードディスクとは大きく異なると考えられる。SSD、特にこのような PCI-Express 接続型の SSD は発展途上であるが、ハードディスクによるボトルネックが顕在化するにつれ、より一般的なデバイスになっていくと考えられる。

開発したアルゴリズムが理論の面からも適切であることを示すために、Metrical Task Systems をはじめとして、キャッシュアルゴリズム関係の理論を調査した。そして、取得したトレースログの統計を参考に、現実に即した理論の導出を目指した。

４．研究成果

(1) 2009 年度の研究成果

2009 年度は、「メニーコア環境におけるファイルアクセス特性の分析」および「理論に基づいた新しいキャッシュアルゴリズムの導出」を進めた。前述のように、本研究の基本的なアイデアは、セカンダリストレージからの読み込みに時間が掛かる、すなわちアクセスコストが大きいデータをより長くキャッシュに留めることでシステム性能を高めることである。

まず、「アクセス特性の分析」を行い、アクセスコストのモデル化を目指した。Linux カーネルのトレース機能を拡張し、DBMS のファイルアクセスログを取得して解析した。ここでは、MySQL と PostgreSQL を DBMS として用い、TPC-H を動作させた時のファイルアクセスログを採取した。

その結果、半導体ストレージである SSD をセカンダリストレージに用いた場合のアクセスコストは、読み込むデータサイズに大きく依存し、アクセスサイズからアクセスコストをモデル化することが可能であると分かった。一方、ハードディスクを用いた場合のアクセスコストは、前述のように時間を計測することで推定可能と考えられたが、誤差が大きく、モデル化は課題として残った。

並行して、「理論に基づいた新しいキャッシュアルゴリズムの導出」を進めた。LRU といった通常のキャッシュアルゴリズムは、将来アクセスされる可能性が最も低いと予想されるデータをキャッシュから追い出す。しかし、アクセスコストが大きいデータをより長くキャッシュに留めることで、読み込み待ち時間を削減できると考えられる。そこで本研究では、Metrical Task Systems と呼ばれるキャッシュモデルを拡張し、アクセスコストの総和の期待値を最小化できる、アクセスコスト考慮型キャッシュアルゴリズムを導出した。

2009 年度初期の研究成果は 2009 年 7 月に行われた iDB Workshop にて発表し研究の改善へ向けて議論を行った。PostgreSQL 8.3.7 に提案手法を実装し、TPC-H で評価を行ったところ、クエリ単体の実行時間で最大 18.3%の性能向上を確認できた（図 1）。その一方で 6.8%の性能低下があった。Power Test 全体の性能向上は 3.4%であった。

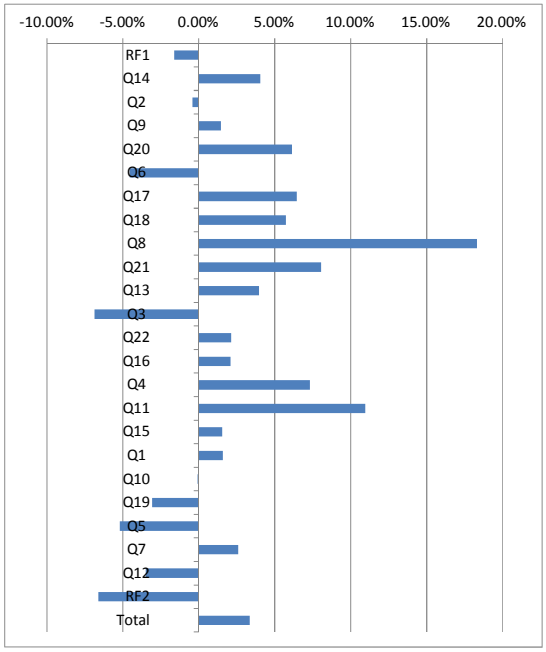

図 1：TPC-H Power Test における クエリごとの性能向上

(2) 2010 年度の研究成果

2010 年度は 2009 年度に構築したアルゴリズムの問題点の改良と実機における評価を続けた。いかにアクセスコストが大きいデータをリアルタイムで自動的に識別し、総アクセスコストを小さくするかが継続的な課題となった。

2010 年度はまず、標準的なデータベースベンチマークである TPC-H を主として、複数ベンチマークのファイルアクセスログを再度採取した。そして、アクセスログをもとにファイルアクセスをトレースドリブンで再現することで 2009 年度に構築したアルゴリズムの問題点を調査した。その結果、現実のハードディスクの動作に応じて正しくアクセスコストの期待値の総和を最小化するには、構築したアルゴリズムでは不十分であることがわかり、モデルの改良が必要となった。モデルの改良に伴いアルゴリズムの計算量が増加したため、動的計画法に近似的な計算を導入することで計算量を削減することを目指した。

また、ハードディスクのアクセスコスト算出のために、ストレージにリクエストを発行してからデータが到着するまでのレスポンスタイムを利用してアクセスコストを推定する手法を開発した。改良したアルゴリズムを用いて実機による評価を行ったところ、TPC-H ベンチマークの QphH 指標で 5%程度、またマイクロベンチマークにおいても性能向上を確認できたが、実用レベルにおいて有意義と考えられる数十%のオーダでの性能向上は確認できず、残念ながら成果発表には至らなかった。今後の課題として、アクセスコストの推定がより容易と考えられる SSD に特化した手法として改良していくことを考えている。

５．主な発表論文等

（研究代表者、研究分担者及び連携研究者には下線）

該当なし

６．研究組織

(1)研究代表者

上田　高徳（UEDA TAKANORI）
早稲田大学・メディアネットワークセンター・助手
研究者番号：90546863