回帰分析を用いたストレージシステムのベンチマーク結果推定

曽我　樹大†　　大西　真晶†　　首藤　一幸†

† 東京工業大学 情報理工学院 数理・計算科学系
〒 152-8552 東京都目黒区大岡山 2-12-1-W8-43
E-mail: soga.t.aa@m.titech.ac.jp, ohnishim@gmail.com, shudo@is.titech.ac.jp

あらまし　現在のストレージシステムにおけるメモリ階層において，メインメモリに用いられる DRAM とストレージに用いられる NAND フラッシュメモリには大きなアクセス性能の差がある．そのギャップを埋める新しいストレージデバイスとして Storage Class Memory（SCM）が開発されてきているが，SCM を用いた将来のストレージシステムに対するベンチマーク結果を直接得ることは難しい．そこで，本研究ではいくつかの既存のストレージデバイスを用いたシステムに対するベンチマーク結果を回帰分析することで，ストレージデバイスの性能からベンチマーク結果を生成する式を求める．また，その生成式を用いて SCM を用いた将来のストレージシステムに対するベンチマーク結果の推定を行う．実際に実験を行い，一定条件においては高い精度で回帰分析を行うことができ，またストレージシステムの設定次第で回帰分析の当てはまりの良さが変わることを確認した．

キーワード　ストレージ, Storage Class Memory, ベンチマーク, 回帰分析

1. はじめに

コンピュータで利用されるストレージデバイスとして HDD・SSD などが一般的であるが，技術の発展によって新しいストレージデバイスが開発されてきている．その中で代表されるのが，相変化メモリ（PRAM）・抵抗変化型メモリ（ReRAM）・磁気抵抗メモリ（MRAM）などの不揮発性メモリである．

現在のストレージシステムにおけるメモリ階層において，メインメモリに用いられる DRAM とストレージに用いられる NAND フラッシュメモリには大きなアクセス性能の差があり，データ処理のボトルネックとなってしまっている．そのアクセス性能のギャップを埋められるメモリは Storage Class Memory（SCM）と呼称され，その役割を担いうるデバイスの必要性が叫ばれている．SCM の役割を担いうるデバイスとして現在注目されているのが，先程述べた不揮発性メモリである．

SCM の開発において，アクセス遅延性能と高容量を両立することはできないことが知られている．したがって，SCM を用いたシステムの設計を考えるにあたってコストと遅延と容量のバランスを考えることが必要になるが，どのようなバランスを採用するかの選択肢は無数にあるため，事前に絞り込むことが必要である．とはいえ，絞り込んだ数の現物であっても開発コストはかかってしまう．そこで，SCM を利用したシステムが実際にどの程度の性能を達成できるのかを予測する手法を考えたい．

これまで，SCM を利用したシステムについての研究はいくつかされている [1], [2] が，SCM の現物を用意できることを前提としていることが多い．また，SCM の現物が用意できない状況でも SCM の性能特性をエミュレートできるフレームワークの研究もされている [3] が，無数のバランスの選択肢がある SCM それぞれに対して，その SCM を利用したシステムの性能予測に対応することは非常にコストが大きく困難である．

そこで本研究では，SCM 等の入手困難な，あるいは入手できないデバイスを利用した様々なシステム全体の性能予測をコストを抑えてできるようにする手法を提案する．具体的には，いくつかの既存のストレージデバイスを用いたシステムに対して行ったベンチマークの結果を回帰分析することで，ストレージデバイスの性能を説明変数として，システムのベンチマーク結果を得る生成式を求める．さらに，その生成式を用いて SCM を用いた将来のストレージシステムに対するベンチマーク結果の推定を行う．

本論文の構成は以下の通りである．２章では本研究の背景を述べ，３章では我々の提案手法について述べる．４章では実験を行い，実際に求めた生成式からベンチマーク結果の推定を行う．５章では本研究についてまとめ，また今後の課題を述べる．

2. 背　　景

本章では，本研究の背景を述べる．特に，今回の提案手法で予測する対象である SCM についての説明と，SCM に関する既存研究の紹介を１章よりも少し詳しく行う．また，回帰分析の説明を行う．

2.1 Storage Class Memory（SCM）

現在のストレージシステムにおけるメモリ階層を図１に示す．この図に見られるように，アクセス遅延を考えると，CPU に用いられる SRAM が１桁 ns，メインメモリに用いられる DRAM が２桁 ns であるのに対して，SSD に用いられる NAND フラッシュメモリは２〜４桁 μs となっていて，その間には３〜５桁程度の性能差が存在する．この性能差がデータ処理のボトルネックとなってしまうことがあるため，このような状況は好ましくないと言える．

このギャップを埋められるメモリが Storage Class Memory（SCM）と呼ばれている．SCM の役割を担いうるデバイスとして現在注目されているのが，１章でも述べた MRAM・ReRAM・

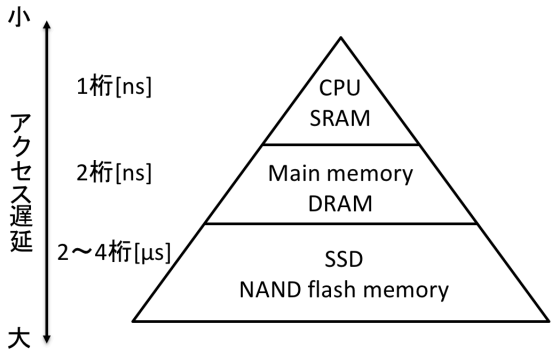

図 1　メモリ階層

PRAM といった不揮発性メモリである．また，SCM の中でも役割によって区別があり，MRAM のような DRAM に近い性能をもつ SCM を Memory-type SCM（M-SCM）と呼び，ReRAM や PRAM のような NAND フラッシュメモリに近い性能をもつ SCM を Storage-type SCM（S-SCM）と呼ぶ．M-SCM はアクセス性能に優れるが大容量化が難しく高コストであり，S-SCM は大容量化は可能であるがアクセス性能が M-SCM と比べて劣っていて低コストであるという特徴がある [4]．

ここまで述べたように，一口に SCM といってもその性能や可能性は様々であり，SCM を用いた新しいシステムの設計を考える場合，コストとアクセス遅延と容量のバランスをあらかじめ考える必要がある．しかし，どのバランスをシステムに採用するかの選択肢は無数にあるため，実際に SCM を用いて性能を測りながら新しいシステムの設計を逐一考えることは非常に高コストであるといえる．

2.2　SCM に関する既存研究

本節では SCM に関する既存研究を整理し，本研究の貢献を明確にする．

既存のデータベース管理システムが NAND フラッシュメモリを前提としているシステムであることを考えると，SCM が使えるならばデータベース管理システムも変えた方が良いという観点から，新しいデータベース管理システムを設計する研究が存在する [1], [2]．また，SCM と NAND フラッシュメモリを組み合わせた SSD を考えるとき，どのようなメモリ構成で最大限のパフォーマンスを発揮できるかを調べた研究 [5] であったり，SCM の書き込み遅延やエネルギー消費を改善する手法の提案 [6] なども存在する．これらは全て，実際に SCM の現物が用意できることを前提とした研究となっている．

実際に SCM の現物が用意できなくとも，SCM の性能特性をエミュレートできるフレームワークの研究 [3] もされている．この研究では，SCM は入手困難であるから，SCM のエミュレーションプラットフォームが必要であるということを主張しており，商用ハードウェアを用いた手法を示している．しかしこの手法では，無数のバランスの選択肢がある SCM それぞれに対して，その SCM を利用したシステムの性能予測に対応することは非常にコストが大きく困難である．

本研究は，SCM 等の評価に用いたいデバイスが直接用意できない状況でも性能予測を行うことができる点や，想定できる様々なシステム全体の性能予測を回帰分析を用いることで低コストで行うことができる点が既存研究と異なっている．

また，本研究のように内部の解析なしにシステム性能を統計的に分析する試みが又川らによってなされている [7], [8]．

2.3　回帰分析

回帰分析 [9] は統計学の用語である．予測や要因分析を行う対象となる変数である目的変数 Y と，その目的変数 Y に影響を与えると考えられる変数である X の間の関係を調べて，$Y = f(X)$ という形のモデルに当てはめる統計的手法を回帰分析という．特に，説明変数 X が 1 種類であるとき，その回帰分析は単回帰分析といい，複数種類であるときは重回帰分析という．式 $Y = f(X)$ は回帰式といい，一次関数・二次関数・対数関数など様々な可能性があるが，

$$Y = aX + b$$

の形で回帰式が表されていて，この a, b を推定する単回帰分析が最も基本的な回帰分析であるといえる．

この a と b を推定するための代表的な手法として最小二乗法が知られている．最小二乗法とは，回帰式 $Y = f(X)$ が n 個の測定値 $(x_i, y_i)(i = 1, 2, ..., n)$ に対してなるべく適切な近似となるように，残差 $y_i - f(x_i)$ の二乗の和が最小となるように回帰式の係数を決定する手法である．数学的に計算することで，先の式の a, b の推定値は次のように求めることが出来る．

$$a = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} {x_i}^2 - \left(\sum_{i=1}^{n} x_i\right)^2}$$

$$b = \frac{\sum_{i=1}^{n} {x_i}^2 \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} x_i y_i \sum_{i=1}^{n} x_i}{n\sum_{i=1}^{n} {x_i}^2 - \left(\sum_{i=1}^{n} x_i\right)^2}$$

また，本研究では回帰分析の当てはまりの良さを表す指標である決定係数 R^2 を用いる．決定係数は $0 \leqq R^2 \leqq 1$ の値をとり．R^2 が 1 に近ければ近いほど，当てはまりのよい回帰式が得られていると言える．観測された n 個のデータを $(x_i, y_i)(i = 1, 2, ..., n)$ と表すとき，観測された全 y_i の平均値を $\bar{y}$，回帰式によって得られる各 y_i の推定値を $\hat{y}_i$ とすると，決定係数 R^2 は以下の式で表される．

$$R^2 = \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

3.　提案手法

本章では，SCM を用いた将来のストレージシステムに対するベンチマーク結果の推定を実際にどのように行うかを具体的に述べる．

3.1 回帰分析の適用方法

回帰分析を利用するにあたって，説明変数 X と目的変数 Y をどのように設定するかが重要である．本研究で行いたいことは「将来のストレージシステムに対するベンチマーク結果の推定」であるため，このベンチマーク結果を Y とおくのが適切である．また，MRAM・ReRAM・PRAM 等様々な可能性のある SCM それぞれに対してどのようにベンチマークが異なってくるのかを知りたいため，それを考慮して X を設定する必要がある．今回はストレージデバイス自体の性能を X とおくことにしている．X は，ストレージデバイスの IO 速度計測のためによく用いられる fio [10]・Iometer [11] などのマイクロベンチマークから得られた値か，あるいはカタログスペックから得られた値を採用する．まとめると，以下のとおりである．

- 説明変数 X：ストレージデバイス自体の性能．すなわち，マイクロベンチマークかカタログスペックから得られた値．
- 目的変数 Y：ベンチマーク結果，すなわち，システム全体の性能を測るアプリケーションベンチマークから得られる値．

3.2 具体的な手順

提案手法の具体的な手順は以下の通りである．

（1） マイクロベンチマークかカタログスペックから，説明変数 X を得る．

（2） いくつかの既存のストレージデバイスを用いたシステムに対して様々なバリエーションのワークロードを用いてベンチマークを行い，ベンチマーク結果，すなわち目的変数 Y を得る．

（3） ある特定のワークロード A を走らせた時の既存のストレージデバイスに対するベンチマーク結果を回帰分析することで，ストレージデバイスの性能からワークロード A を走らせた時のベンチマーク結果を生成する式を求める．

（4） 上で求めたワークロード A のベンチマーク結果生成式を用いて，X に想定するデバイス（例えば MRAM）に対応する値を入れてそのときの Y を求めることで，ワークロード A に対するベンチマーク結果を推定する．

（5） 以上を他のワークロード B・C・D・... についても行う．

3.3 ここまでの手順の問題点と解決策

ここまでの手順を用いることで，想定するデバイスにおける各ワークロード A・B・C・D・... に対するベンチマーク結果を推定することはできるが，実際に走らせていないワークロードに対するベンチマーク結果を推定することはできない．そこで，改めて以下の設定で回帰分析を行うことで実際に走らせていないワークロードに対するベンチマーク結果の推定も可能となる．

- 説明変数 X：ワークロードを示すパラメータ．例えば，扱う総データ量など．
- 目的変数 Y：想定するデバイスにおける，各ワークロードに対するベンチマーク結果．

3.4 提案手法の妥当性の確認

本節では，提案手法によって得られたベンチマーク結果の推定値の妥当性を確認する方法について述べる．

ベンチマーク結果の推定値の妥当性とはすなわち，ベンチマーク結果の生成式の妥当性でもある．ベンチマーク結果の生成式を求めるために今回は回帰分析を用いているため，つまり回帰分析によるアプローチが適切であるかを確認する必要がある．

いくつかの既存のストレージデバイスを用いたシステムに対して走らせたワークロードによるベンチマーク結果から得られている説明変数 X と目的変数 Y の組み合わせをグラフ上に表し，これが一つの回帰式で可能な限り説明されていることが望ましい．そこで，今回は先に述べた決定係数 R^2 をの値を見て回帰分析によるアプローチの妥当性を確認する．説明変数 X として採用する値は，カタログスペック・fio から得られた値・Iometer から得られた値のいずれかであり，さらに各マイクロベンチマークから得られる値にも様々なものがある．この方法を用いることで，説明変数 X として採用しうる様々な値のうちどの値を X として採用した回帰分析が最も妥当性が高いかを判断することが可能である．

表 1 使用したデバイス

通常 SSD	Intel(R) SSD DC S3520 Series (1.2TB, 2.5in SATA 6Gb/s, 3D1, MLC)
高速 SSD	Intel(R) Optane SSD 900P Series (480GB, 1/2 Height PCIe x4, 20nm, 3D XPoint)
DRAM	Samsung M386A4K40BB0-CRC 32GB DDR4-2400 LP ECC LRDIMM

表 2 実験マシンの構成

OS	Ubuntu 16.04.3 LTS
CPU	Intel(R) Xeon(R) CPU E5-2698 v4 @ 2.20GHz × 2
Java	Java SE 8, Update 151

4. 実験と考察

本章では，3 章で述べた提案手法によって SCM を用いた将来のストレージシステムに対するベンチマーク結果の推定を行い，その妥当性を確認する実験を行う．

4.1 実験環境

4.1.1 ストレージデバイス及びマシン構成

今回は，SCM を用いた将来のストレージシステムに対するベンチマーク結果の推定を行うために使用するストレージデバイスを 3 種類用意した．一つ目は通常の NAND フラッシュメモリを搭載した SSD である．二つ目は新しい不揮発性メモリ技術である「3D XPoint」を利用した高速な SSD である．これは不揮発性メモリの技術を用いてはいるものの，メモリ階層内では高速な SSD として扱うことができる．三つ目は DRAM である．DRAM は通常メインメモリとして使用されるが，今回は Linux の tmpfs を利用することで DRAM をストレージデバイスとして扱って実験を行う．ストレージデバイスに関する詳細は表 1 に示す．また，マシン構成を表 2 に示す．

4.1.2 使用するソフトウェア

今回は測定を行うシステムは，MyCassandra [12] を用いたデータベースシステムとする．また，そのシステム全体の性能を測るアプリケーションベンチマークとしては Yahoo! Cloud

表 3　YCSB のワークロード

Workload	書き込み割合	読み出し割合	アクセス分布
Write-Only	100%	0%	Zipfian 分布
Write-Heavy	50%	50%	Zipfian 分布
Read-Heavy	5%	95%	Zipfian 分布
Read-Only	0%	100%	Zipfian 分布

Serving Benchmark（YCSB）[13] を用いる．以下，この二つのソフトウェアについて述べる．

4.1.2.1 MyCassandra

MyCassandra はストレージエンジン部分を差し替え可能にしたデータベース管理システムである．MyCassandra は Apache Cassandra [14] というデータベース管理システムを拡張したものとなっており，本来の Apache Cassandra 由来のストレージエンジンだけでなく，MySQL [15] のストレージエンジンを用いることも可能になっている．MyCassandra においては，Apache Cassandra 由来のストレージエンジンは Google Bigtable [16] を元に作られているため Bigtable エンジンと呼ばれており，MySQL のストレージエンジンとしては InnoDB がデフォルトとして用いられているが，これを単に MySQL エンジンと呼んでいる．本研究では，同一のシステム内でも可能な限り実験のバリエーションを増やしたいという動機から，このストレージエンジンが差し替え可能となっている MyCassandra を実験対象として採用している．ストレージエンジンによってその特性は当然異なっており，Bigtable エンジンは書き込み性能特化，MySQL エンジンは読み出し性能特化となっている．

4.1.2.2 Yahoo! Cloud Serving Benchmark（YCSB）

YCSB は NoSQL 型データベースに対して利用できるベンチマークツールであり，読み出し及び書き込みの比率や，サーバへのアクセス分布，1 秒あたりに発行する処理要求数の目標値（スループット）などの項目を指定できる．その設定に基いて YCSB が対象とするデータベースに対してワークロードを実行し，書き込みや読み出しに要した時間，すなわちアクセス遅延を集計する．本実験では，データの書き込み比率と読み出し比率に応じて Write-Only，Write-Heavy，Read-Heavy，Read-Only の 4 種類のワークロードを利用し，それぞれのワークロードで扱うデータの総量を 2G・4G・8G・16G・32G・64G・128G と変化させてアクセス遅延を集計する．各ワークロードの読み書き処理の比率は表 3 に示す．アクセス対象のデータ分布としては，YCSB のデフォルトで設定されている Zipfian 分布を用いる．Zipfian 分布とは，データの新しさとは無関係に，人気によってアクセス頻度が決まるようなアプリケーションのデータアクセス分布を確率としてモデル化したものであり，ごく一部のデータがヘッドになり，大部分がテールになるという特徴を持っている．

4.2　実験結果と考察

4.2.1　fio によるマイクロベンチマーク結果

fio によるマイクロベンチマーク結果を示す．今回は，blocksize パラメータを 1K・4K・16K・64K・256K と 5 種類使用し，readwrite パラメータを randread・randwrite の 2 種類使用し，計 10 種類の条件でベンチマークを行った．同一条件で 5 回ベンチマークを走らせ，最大値と最小値を除いた 3 回の平均を取って最終的な結果としている．表 4 に読み出しアクセス遅延，表 5 に書き込みアクセス遅延，表 6 に読み出しスループット，表 7 に書き込みスループットを示す．

概ね，いずれのデバイスでもブロックサイズが大きくなればなるほどアクセス遅延が大きくなり，スループットも大きくなっていることが観察できる．また，全体的に同一条件ならば NAND Flash が最も性能が悪く，3D XPoint が真ん中の性能で，DRAM が最も良い性能が出ていることも観察できる．一部，ブロックサイズ 1K と 4K で性能が逆転している箇所が存在するが，ext4 のファイルシステムで用いているブロックサイズが 4096byte であることが関係している可能性がある．しかし，原因を正確に特定することはまだ出来ていないため，今後の課題としたい．

4.2.2　MySQL エンジン＋ YCSB のベンチマーク結果と回帰分析及び考察

MySQL エンジンの MyCassandra を用いたデータベースシステムに対して，YCSB でベンチマークを取った結果を示す．今回のベンチマークでは，表 3 で示した 4 種類のワークロードを NAND Flash・3D XPoint・DRAM の 3 種類のデバイスに対して走らせ，それぞれの読み出し遅延，書き込み遅延を測定した．これを，使用するデータ量 2G・4G・8G・16G・32G・64G・128G[byte] と変化させ，それぞれに対して結果を集計した．fio マイクロベンチマークのときど同様に，同一条件で 100 秒のベンチマークを 5 回走らせ，最大値と最小値を除いた 3 回の平均を取って最終的な結果としている．その結果を図 2 に示す．この図は非常にデータ量が多く煩雑であるため，一部分だけを用いた回帰分析の具体例を以下に示す．

例えば，データ量 16G のときの Write-Only ワークロードに対する YCSB 結果を回帰分析することでベンチマーク結果を生成する式を求めることを考える．このとき用いる目的変数 Y は，図 2 の該当箇所を読み取ると，NAND Flash は 4590.5499，3D XPoint は 2164.09408，DRAM は 1684.78087 となる．説明変数 X の候補として今回考えるのは，表 5 に示した各ブロックサイズにおける書き込みアクセス遅延と，表 7 に示した各ブロックサイズにおける書き込みスループットの計 10 種類である．fio の読み出し遅延やスループットを説明変数の候補として今回考えないのは，Write-Only ワークロードで計測しているのは書き込みに関する性能のみだからである．10 種類それぞれにおいて単回帰分析を行い，決定係数 R^2 が最も大きくなる説明変数 X がどのマイクロベンチマーク結果であるかを調べる．実際に回帰分析を行って決定係数を求めた結果を表 8 に示す．表 8 の結果から，fio でブロックサイズ 64K のときに測定した書き込みアクセス遅延を説明変数 X とする回帰分析が一番高い決定係数 $R^2 = 0.999852$ を得られることを確認できた．このときの回帰直線をグラフに表したのが図 3 である．3 つのデバイスから得られたベンチマーク結果がグラフ内の一直線上に並んでいることが確認できる．

以上の具体例から確認した手順と同様にして，YCSB の各

MySQL Engine

workload	data [GB]	NAND Flash read latency [us]	NAND Flash write latency [us]	3D XPoint read latency [us]	3D XPoint write latency [us]	DRAM read latency [us]	DRAM write latency [us]
Write-Only	2		4616.02527		1827.35622		1566.22883
	4		4252.80592		1911.93034		1416.51313
	8		4546.99747		1935.08688		1375.28897
	16		4590.5499		2164.09408		1684.78087
	32		4619.23023		2156.5838		1348.40848
	64		5080.41588		2281.78561		1587.08323
	128		5279.26575		2130.56413		1784.671
Write-Heavy	2	1155.33263	2805.29355	951.208558	1741.05167	900.508833	1437.99658
	4	1151.46028	2788.10088	958.13071	1819.68761	910.561842	1506.31857
	8	1130.27222	2712.42675	930.276642	1682.80994	892.430341	1411.90745
	16	1171.94158	2871.21676	938.056035	1699.77425	907.150283	1447.2986
	32	1165.61274	2770.2356	1019.76507	1948.52751	924.707217	1567.58883
	64	1150.59265	2770.00707	970.071249	1773.36206	947.173369	1610.71554
	128	1212.89575	3056.34015	971.909572	1762.45119	973.370668	1649.09997
Read-Heavy	2	944.004196	2064.56038	930.034755	1734.14267	930.506902	1542.27229
	4	933.407788	2047.69763	929.338036	1806.46599	933.879975	1506.74293
	8	948.727	2091.26897	938.153463	1803.05717	932.055682	1532.46573
	16	960.493515	2087.64127	935.196281	1787.93134	947.345982	1599.63181
	32	968.327175	2093.57862	952.300401	1822.10598	947.62424	1575.86884
	64	962.708709	2153.43275	962.890216	1836.82943	946.221426	1535.37623
	128	983.844977	2161.76618	959.128493	1855.33681	960.578351	1573.16239
Read-Only	2	955.486717		953.14715		950.065357	
	4	951.73811		939.342793		950.621037	
	8	943.88272		951.66939		956.657993	
	16	962.23773		953.45061		955.806457	
	32	961.44467		953.945893		961.957607	
	64	953.76053		964.40833		970.183463	
	128	970.062073		966.21263		971.18902	

図 2　YCSB のベンチマーク結果（MySQL エンジン）

表 4　fio マイクロベンチマーク結果：読み出しアクセス遅延 [μs]

blocksize	NAND Flash	3D XPoint	DRAM
1K	124.49	17.877	0.91333
4K	151.73	18.287	1.0333
16K	250.82	25.803	3.5567
64K	583.91	49.79	15.247
256K	1651.9	135.28	53.26

表 5　fio マイクロベンチマーク結果：書き込みアクセス遅延 [μs]

blocksize	NAND Flash	3D XPoint	DRAM
1K	57.207	80.303	0.93
4K	53.217	27.403	1.3967
16K	87.313	30.01	5.7967
64K	222.34	55.427	19.527
256K	867.55	161.94	68.673

表 6　fio マイクロベンチマーク結果：読み出しスループット [MB/s]

blocksize	NAND Flash	3D XPoint	DRAM
1K	7.7445	48.589	793.18
4K	25.466	194.52	2938.6
16K	61.641	560.69	4103.5
64K	106.44	1197.2	4000.8
256K	150.75	1825.9	4724.5

表 7　fio マイクロベンチマーク結果：書き込みスループット [MB/s]

blocksize	NAND Flash	3D XPoint	DRAM
1K	16.524	11.877	776.73
4K	70.728	135.69	2265.4
16K	174.07	491.27	2547.0
64K	276.00	1067.1	3168.6
256K	284.67	1466.4	3475.7

表 8　データ量 16G，Write-Only ワークロードの書き込みアクセス遅延の回帰分析の決定係数

マイクロベンチマークの種類	決定係数 R^2
1K latency	0.145438
4K latency	0.868356
16K latency	0.980820
64K latency	0.999852
256K latency	0.997747
1K throughput	0.388325
4K throughput	0.419025
16K throughput	0.516074
64K throughput	0.657708
256K throughput	0.755037

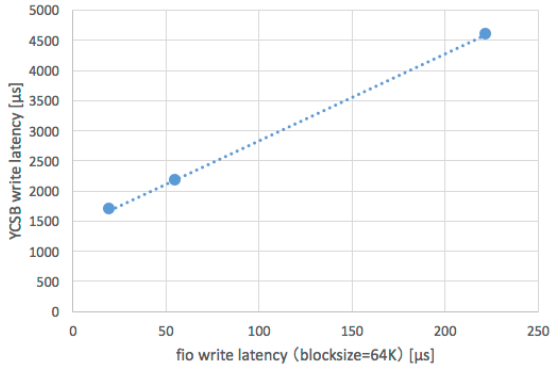

図 3　データ量 16G，Write-Only ワークロードの書き込みアクセス遅延に対し，64K latency の fio ベンチマーク結果を説明変数とした時の回帰直線

ワークロードの読み出しアクセス遅延，書き込みアクセス遅延に対して fio ベンチマーク結果から回帰分析を行い，最大の決定係数 R^2 を得られるのは fio ベンチマークのどの結果を説明変数 X としたときであるのかの確認を図 2 に示した全てのケー

スで行う．これをまとめた結果を表 9（読み出し性能），表 10（書き込み性能）に示す．表の結果を観察すると，以下のような点が確認できる．

- 回帰分析が全体的に良く当てはまっているのは Write-Only ワークロードの書き込みアクセス遅延であることが確認できる．全体的に書き込みアクセス遅延の回帰分析よりも読み出しアクセス遅延の回帰分析の方が当てはまりが悪く，ワークロードの読み出しの割合が増えるほど決定係数が下がっていく傾向があることが読み取れる．
- ワークロードによって，当てはまりの良い回帰分析ができる fio ベンチマークの値に傾向があることが読み取れる．例えば，Write-Only ワークロードの書き込みアクセス遅延はブロックサイズ 64K のアクセス遅延を用いて回帰分析すると良い場合が多く，ワークロードの書き込みの割合が減っていくにつれて 64K → 16K → 4K と最も当てはまりの良い回帰分析ができる fio のブロックサイズが小さくなっていくおおよその傾向があることが確認できる．Write-Heavy の読み出しアクセス遅延は fio のブロックサイズ 1K か 4K のアクセス遅延を用いると当てはまりの良い回帰分析ができる場合が多いが，ワークロードの読み出しの割合が増えていくにつれて fio のブロックサイズ 1K や 256K のスループット等が混ざってきて不安定になってくる．

以上の 2 つを踏まえて考察すると，MyCassandra の MySQL エンジンでは，書き込みアクセス遅延よりも読み出しアクセス遅延の方が，ストレージデバイス自体の単純なアクセス遅延やスループット以外の影響を受けている可能性が高いと考えることが出来る．なぜなら，読み出しの多いワークロードになっていくにつれて，ストレージデバイス自体の性能を説明変数 X とした回帰分析の当てはまりの良さを表す決定係数 R^2 の値が全体的に下がっていき不安定になっていることが確認できるからである．それでは，ストレージデバイス自体の性能以外に具体的に何の影響を受けているかということが問題になるが，この点は正確には解明できていない．この点を解明し，ストレージデバイス自体の性能以外の数値も説明変数 X として取り入れた回帰分析を行って傾向を観察することは今後の課題となる．

4.2.3 Bigtable エンジンと MySQL エンジンの結果の比較と考察

Bigtable エンジンの MyCassandra を用いたデータベースシステムに対しても YCSB でベンチマークを取った．諸々の条件は MySQL エンジンのときと同様で，変更したのはストレージエンジン部分が MySQL から Bigtable になっている点のみである．MySQL で行った手順と同様にして，YCSB の各ワークロードの読み出しアクセス遅延，書き込みアクセス遅延に対して fio ベンチマーク結果から回帰分析を行い，最大の決定係数 R^2 を得られるのは fio ベンチマークのどの結果を説明変数 X としたときであるのかの確認を全てのケースで行う．

MySQL エンジンでの最大決定係数の平均値は，読み出しで 0.837477（表 9），書き込みで 0.998331（表 10）であった，Bigtable エンジンでも同様にして得た最大決定係数の平均値を求め，結果を表 11 にまとめる．

表 9　MySQL エンジン＋ YCSB の全てのワークロードにおける最大の決定係数とその fio ワークロード（読み出し性能）

YCSB ベンチマークの種類	データ量 [GB]	最大の決定係数を得られた fio ワークロード	最大の決定係数
Write-Heavy	2	1K latency	0.996149
Write-Heavy	4	1K latency	0.996337
Write-Heavy	8	1K latency	0.999531
Write-Heavy	16	4K latency	0.999995
Write-Heavy	32	256K throughput	0.926602
Write-Heavy	64	4K latency	0.999997
Write-Heavy	128	256K latency	0.997418
Read-Heavy	2	256K latency	0.994335
Read-Heavy	4	1K throughput	0.293095
Read-Heavy	8	1K latency	0.942166
Read-Heavy	16	256K latency	0.730001
Read-Heavy	32	1K latency	0.991907
Read-Heavy	64	1K throughput	0.996902
Read-Heavy	128	256K latency	0.990454
Read-Only	2	256K throughput	0.994483
Read-Only	4	256K latency	0.281778
Read-Only	8	1K latency	0.928411
Read-Only	16	256K latency	0.908367
Read-Only	32	1K throughput	0.259597
Read-Only	64	1K latency	0.949353
Read-Only	128	1K throughput	0.410145
		最大の決定係数の平均値	0.837477

表 10　MySQL エンジン＋ YCSB の全てのワークロードにおける最大の決定係数とその fio ワークロード（書き込み性能）

YCSB ベンチマークの種類	データ量 [GB]	最大の決定係数を得られた fio ワークロード	最大の決定係数
Write-Only	2	256K latency	0.999121
Write-Only	4	64K latency	0.999994
Write-Only	8	64K latency	1.000000
Write-Only	16	64K latency	0.999852
Write-Only	32	16K latency	0.997095
Write-Only	64	64K latency	0.999501
Write-Only	128	256K latency	0.999710
Write-Heavy	2	64K latency	0.997886
Write-Heavy	4	16K latency	0.996789
Write-Heavy	8	64K latency	0.998971
Write-Heavy	16	64K latency	1.000000
Write-Heavy	32	16K latency	0.999530
Write-Heavy	64	256K latency	0.999473
Write-Heavy	128	256K latency	0.998817
Read-Heavy	2	16K latency	0.993892
Read-Heavy	4	4K latency	0.996381
Read-Heavy	8	4K latency	0.999586
Read-Heavy	16	16K latency	0.9902
Read-Heavy	32	4K latency	0.999083
Read-Heavy	64	4K latency	0.999734
Read-Hevay	128	4K latency	0.999327
		最大の決定係数の平均値	0.998331

表 11 より，Bigtable エンジンにおける読み出しと書き込みの性能は，ストレージデバイス自体の単純なアクセス遅延やスループット以外の影響を MySQL エンジンよりも大きく受

表 11　各エンジンにおける読み書き性能に関する最大決定係数の平均値

	MySQL エンジン	Bigtable エンジン
読み出し	0.837477	0.752747
書き込み	0.998331	0.755262

表 12　各 YCSB のデータ量における回帰式の a, b の値及び SCM の性能予測

データ量 [GB]	a の推定値	b の推定値	SCM①性能予測値	SCM②性能予測値
2	15.55701	1128.205	1906.055	1517.130
4	13.99711	1140.000	1839.856	1489.928
8	15.64159	1069.077	1851.157	1460.117
16	14.39262	1386.865	2106.496	1746.680
32	15.69999	1152.240	1937.240	1544.740
64	17.08295	1290.301	2144.404	1717.353
128	17.73891	1306.947	2193.893	1750.420

表 13　走らせていない YCSB のデータ量における SCM の性能予測

YCSB のデータ量 [GB]	SCM①の性能予測値	SCM②の性能予測値
48	2027.159	1626.645
96	2150.680	1720.387
192	2397.722	1907.872

表 14　各ストレージデバイスにおける，回帰分析の決定係数

ストレージデバイス	決定係数 R^2
NAND Flash	0.839929
3D XPoint	0.332145
DRAM	0.419822
SCM①（予測値）	0.641802
SCM②（予測値）	0.484443

けていると考えられる．その要因の 1 つとして考えられるのは，Bigtable エンジンが書き込み特化の性能を実現するために行っている内部処理のコンパクションの存在である．コンパクションに関する既存研究で述べられているように [17]，コンパクションの処理が行われている最中はストレージシステムに対する読み出しアクセス遅延や書き込みアクセス遅延が悪化する．したがって，今回の実験で得られた結果はストレージデバイス自体の単純なアクセス遅延やスループット以外の影響を大きく受けていてしまい，当てはまりの良い回帰分析ができないものとなった．他の要因として，Bigtable エンジンではデータの書き込みにメモリ部分（DRAM）を多用すること，DRAM 自体のキャッシュを切らず製品出荷時の状態のままの設定で行っていたこと等も考えられる．今後の課題として Bigtable エンジンを用いた適切な条件による実験が挙げられるが，コンパクション処理の適切な制御，ワークロードを走らせる時間を長くすること等を念頭に置かなければならない．

4.2.4　SCM の性能予測と他ワークロードのベンチマーク結果推定

本項では，ここまでで得られた実験結果をもとに SCM をストレージデバイスとした一つのストレージシステムの性能予測を行い，実際に走らせていないワークロードにおける性能予測も行う．

今回は，比較的安定した実験結果を得られた MySQL エンジンにおける Write-Only ワークロードの書き込みアクセス性能の予測を行う．表 10 より，Write-Only ワークロードにおいては fio のブロックサイズ 64K の書き込みアクセス遅延を用いた回帰分析が当てはまりが比較的良いため，これを基準に考える．fio のブロックサイズ 64K の書き込みアクセス遅延は表 5 より，NAND Flash で 222.34，3D XPoint で 55.427，DRAM で 19.527 であるため，これを基準に SCM のマイクロベンチマーク結果を仮定する．SSD 寄りの性能である S-SCM はアクセス遅延 50 [μs] と仮定し，これを SCM①とする．DRAM 寄りの性能である M-SCM はアクセス遅延 25 [μs] と仮定し，これを SCM②とする．

まず，Write-Only ワークロードにおいて，fio のブロックサイズ 64K の書き込みアクセス遅延を用いた回帰分析によって得られた回帰式 $Y = aX + b$ の a, b の推定値が YCSB の各データ量においてどうなったかと，その a, b を用いて SCM①と SCM②の YCSB 書き込みアクセス遅延性能を予測した結果を表 12 に示す．概ね，SCM の性能は図 2 に見られる 3D XPoint と DRAM の性能の間に収まっていることを確認できる．

次に，表 12 で得られたデータを用いて，実際に走らせていないワークロードにおける SCM の性能予測を行う．具体的には，YCSB のデータ量を説明変数 X とし，SCM の性能予測値を目的変数 Y として回帰分析を行い，実際に走らせていないデータ量（例えば 48GB・96GB・192GB など）における SCM の性能を予測する．この手順による性能予測値を表 13 に示す．概ね，データ量が大きくなればなるほどアクセス遅延が大きくなる傾向が読み取れる．

最後に，この SCM の性能予測の妥当性を確認する．そもそも，データ量を説明変数 X としたアクセス遅延の回帰分析が適切である保証はない．ここでは，判断材料の 1 つとして先程も用いた回帰分析の決定係数 R^2 を確認する．YCSB の Write-Only ワークロードにおいて，データ量を説明変数 X として書き込みアクセス遅延を目的変数 Y とした回帰分析の決定係数 R^2 が各ストレージデバイス（予測した SCM①と SCM②も含む）においてどのような値をとるかを表 14 にまとめる．また各ストレージデバイスの回帰分析のグラフを図 4 に示す．図 4 は 3D XPoint と DRAM のグラフを薄くし，SCM の予測値のグラフを強調したグラフとなっている．表とグラフから，大まかにはデータ量が増えるほどアクセス遅延が大きくなっていることが確認できるが，決定係数は全体的に低い．したがって，大まかな予測は可能であるが，前項までで行っていた同一ワークロードにおける他デバイスの回帰分析による性能予測と比べると推定精度は落ちると言うことができる．前項までの考察と同様，アクセス性能に影響を与えている要因を特定し，それを新たに説明変数 X として取り入れることで，より当てはまりの良い回帰分析を行える可能性はあるため，その部分は今後の課題としたい．

5.　まとめと今後の課題

本研究では，既存のストレージデバイスを用いたシステムに

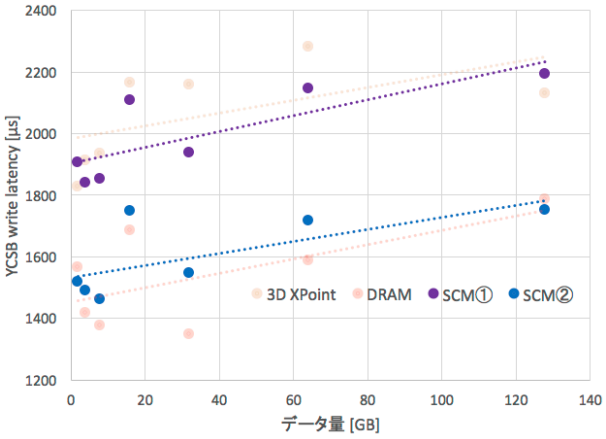

図 4　各ストレージデバイスの回帰分析のグラフ（SCM を強調）

対してベンチマークを行い，その結果を回帰分析することで，ストレージデバイス自体の性能からベンチマーク結果を生成する式を求め，その生成式を用いて SCM を用いた将来のストレージシステムに対するベンチマーク結果の推定を行う手法を提案した．また，提案手法をもとに実際に実験を行ってベンチマーク結果の推定を行い，提案手法の妥当性の確認を行った．

実験により，ストレージデバイス自体の性能以外の影響が小さいと考えられるケースにおける回帰分析は高い精度で行えることを確認できた．また，ストレージデバイス自体の性能以外の影響が大きいケースについては，その原因を特定し，ストレージデバイス自体の性能以外の情報も説明変数 X として合わせて利用することによって回帰分析の妥当性を上げられる可能性があるという指摘を行った．

また，この提案手法は SCM に限らず入手できないデバイスの性能を説明変数 X として入れることができるならば，それだけでベンチマーク結果の推定を行える．したがって，現在の技術では到達し得ない性能のデバイスを用いた場合のベンチマーク結果の推定も行えるため，将来のストレージシステムの設計を行う上で非常に有用な手法であると考えられる．

今後の課題としては，以下のような点が挙げられる．

- 今回の実験では，既存のデバイスとして用意できているものが 3 種類のみであった．この種類をより充実させることで，回帰分析によって得られる式の妥当性はさらに上がると考えられるため，可能な限りストレージデバイスを充実させた上で実験を行うべきである．
- ストレージデバイス自体の性能以外の影響が大きく出ている部分について原因の特定を行い，それを考慮した上で実験と分析を行うこと．原因の特定を行えたら，予想外の挙動をしないようにストレージシステムに対して適切な制御を行い，さらにストレージデバイス自体の性能以外の適切な説明変数 X を設定して重回帰分析を行うことが必要である．

謝 辞

本研究の一部は，国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の委託業務として行われた．また，本研究は JSPS 科研費 25700008 および 16K12406 の助成を受けたものである．

文　献

[1] Hideaki Kimura. FOEDUS: OLTP engine for a thousand cores and NVRAM. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pp. 691–706. ACM, 2015.

[2] Katelin A. Bailey, Peter Hornyack, Luis Ceze, Steven D. Gribble, and Henry M. Levy. Exploring storage class memory with key value stores. In *Proceedings of the 1st Workshop on Interactions of NVM/FLASH with Operating Systems and Workloads*, INFLOW '13, pp. 4:1–4:8. ACM, 2013.

[3] Dipanjan Sengupta, Qi Wang, Haris Volos, Ludmila Cherkasova, Jun Li, Guilherme Magalhaes, and Karsten Schwan. A framework for emulating non-volatile memory systemswith different performance characteristics. In *Proceedings of the 6th ACM/SPEC International Conference on Performance Engineering*, pp. 317–320. ACM, 2015.

[4] Takahiro Onagi, Chao Sun, and Ken Takeuchi. Design guidelines of storage class memory based solid-state drives to balance performance, power, endurance, and cost. *Japanese Journal of Applied Physics*, Vol. 54, No. 4S, 2015.

[5] Chihiro Matsui, Tomoaki Yamada, Yusuke Sugiyama, Yusuke Yamaga, and Ken Takeuchi. Optimal memory configuration analysis in tri-hybrid solid-state drives with storage class memory and multi-level cell/triple-level cell NAND flash memory. *Japanese Journal of Applied Physics*, Vol. 56, No. 4S, 2017.

[6] Sheyang Ning, Tomoko Ogura Iwasaki, and Ken Takeuchi. Write stress reduction in 50nm Al x O y ReRAM improves endurance 1.4× and write time, energy by 17%. In *Memory Workshop (IMW), 2013 5th IEEE International*, pp. 56–59. IEEE, 2013.

[7] Naoki Matagawa and Kazuyuki Shudo. Breakdown of a benchmark score without internal analysis of benchmarking program. *CoRR*, arXiv:1610.06307, 2016/10/20.

[8] 又川尚樹, 首藤一幸. ベンチマークの内部解析を要さないスコア内訳分析手法. 電子情報通信学会 技術研究報告, Vol. 115, No. 518, pp. 229–234, 2016/3/24-25.

[9] 千風彦襄谷. 回帰分析のはなし. 東京図書, 東京, Japan, 1985.

[10] J. Axboe. fio HOWTO. `https://github.com/axboe/fio/blob/master/HOWTO`.

[11] Open Source Development Lab. Iometer. `http://iometer.org/`.

[12] Shunsuke Nakamura and Kazuyuki Shudo. MyCassandra: A cloud storage supporting both read heavy and write heavy workloads. Proceedings of the 5th Annual International Systems and Storage Conference (SYSTOR '12) , p. 14. ACM, 2012.

[13] Brian F Cooper, Adam Silberstein, Erwin Tam, Raghu Ramakrishnan, and Russell Sears. Benchmarking cloud serving systems with YCSB. In *Proceedings of the 1st ACM symposium on Cloud computing*, pp. 143–154. ACM, 2010.

[14] The Apache Software Foundation. Apache Cassandra. `http://cassandra.apache.org/`.

[15] Oracle Corporation. MySQL. `http://www.mysql.com/`.

[16] Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C Hsieh, Deborah A Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes, and Robert E Gruber. Bigtable: A distributed storage system for structured data. *ACM Transactions on Computer Systems (TOCS)*, Vol. 26, No. 2, pp. 4:1–4:26, 2008.

[17] 曽我 樹大, 華井 雅俊, 高塚 康成, 首藤 一幸. Key sorting buffer を用いた時系列データのデータベース処理高速化. 第 8 回データ工学と情報マネジメントに関するフォーラム, 2016.