修　　士　　論　　文

入出力トレースを用いた瓦書き磁気記録型ディスクドライブの性能特性試験に関する研究

指導教員　　豊田正史 准教授

東京大学 大学院情報理工学系研究科
電子情報学専攻

氏　名　　48-166456 佐藤 佑紀

提　出　日　　平成29年2月1日

概 要

磁気ディスクドライブは電磁的記録を保存する手段として中心的役割を担い, 多くのコンピュータで情報記憶装置として用いられてきた. その役割の重要性から磁気ディスクドライブに対する記録容量増加の需要は現在に至るまでとどまることなく, 継続的に記録容量は増加してきた [1].

現在記録密度の上昇を牽引している技術の一つが瓦書き磁気記録型ディスクドライブであり注目を集めている. 瓦書き磁気記録方式は記録方式を変更することによって記録密度の向上を図る技術であるが, その記録方式がもたらす影響の観測はいまだ十分に行われていない.

本論文では, 実際のドライブ管理 (drive managed) 方式の磁気ディスクドライブに対してマイクロベンチマークを用いて基礎性能観測を行うとともに, ホスト管理（host managed）方式の瓦書き磁気記録型ディスクドライブについてエミュレータを用いて著者が構築した入出力トレースベースの性能特性試験環境を示し, トランザクション処理負荷に対する当該磁気ディスクドライブの性能特性の試験結果を報告し, 考察する.

目 次

図 目 次

表 目 次

第1章　序論

1.1　はじめに

磁気ディスクドライブの歴史は大容量化の歴史でもある. 世界で初めての磁気ディスクドライブは 1956 年に誕生した. 磁気ディスクドライブは誕生した当時から大容量化が熱望されており, IBM によって開発された IBM RAMAC 305 に搭載された世界最初の磁気ディスクドライブである IBM 350 は 24 インチのプラッタ 50 枚で構成されていたが, その記憶容量は 5MB に満たなかった [1]. 1961 年に導入された IBM 1301 はヘッドを読み取り用と書き込み用の２つ用いる技術, ヘッドを空気抵抗でディスクから自力で浮かせることで書き込み, 読み取りを行う技術が初めて用いられ, IBM 350 と比べて同面積で 13 倍の記憶容量を実現した. その後も技術革新は進み, 1980 年に現在の Seagate が記憶容量 5MB でパソコン用の 5.25 インチ HDD を開発して以来, パソコン用磁気ディスクの開発が進み, 1980 年代後半頃には 3.5 インチ HDD の開発が主流となっていった. 大きさが 3.5 インチで統一されたことでますます大容量化に求められる技術は高度化していく. まずは, 読み込み, 書き込みに用いられ, トラック幅に直接関係してくるヘッド技術が 1990 年代に MR ヘッド (Magneto Resistance Head) へと変わり,2000 年に入ると GMR ヘッド (Giant Magnetic Resistance Head) そして TMR ヘッド (Tunnel Magneto Resistance Head) へと移行していった. またヘッド技術のみならず垂直磁気記録方式といった記録方式においても技術が向上し現在に至る.

最近では, エネルギーアシスト磁気記録, ビットパターンメディア, 瓦書き磁気記録などの技術が注目を集めている. エネルギーアシスト磁気記録技術は大きく分けて熱アシスト磁気記録 (Thermal Assisted Magnetic Recording) とマイクロ波アシ

スト磁気記録 (MAMR : Microwave Assisted Magnetic Recording) の 2 種類が存在し, これらは記録メディアの磁性粒を小さくして記録密度を高めることにより失われてしまう熱安定性を高保磁力材料で補った際に, 現在の磁気ヘッドの磁気では書き込めなくなってしまう問題が生じるので熱やマイクロ波を照射することで, 高保磁力材料に磁気を通りやすくする方式である. ビットパターンメディア (BPM : Bit Pattened Media) は磁気メディアの表面に加工をすることでノイズの発生を抑制する方式である. これらの技術は HDD を構成する部品の性能を向上させることで高密度化, 大容量化を図る方式であるが, 瓦書き磁気記録 (SMR : Shingled Magnetic Recording) は部品レベルではなく記録方式の変更によって高密度化, 大容量化を図る方式であるため他の方式に比べて実現しやすく, すでに製品化もなされており, 現在の磁気ディスクドライブの高密度化を牽引している.

SMR の技術としては 2009 年に R. Wood らによってその技術概要が提案されており, 2010 年には A. Amer らによって SMR 型磁気ディスクドライブを実際に設計する上で問題となってくるランダムな書き込みへの対処方法などが提案されてきた [2] [3]. その後 2014 年には, Storage Networking Industry Association (SNIA) などによってハードウェアレベルの標準化や, International Committee for Information Technology Standards (INCITS) の T10 committee や T13 committee などによって SMR 型磁気ディスクドライブを扱うためのコマンドセットの標準化が行われている [4] [5] [6]. 現在, SMR については物理空間におけるデータの利用効率を高めるためのデータマネジメントアルゴリズムの研究や, SMR 型磁気ディスクドライブに適したファイルシステムの研究等が行われている [7] [8] [9] [10] [11]. SMR 型磁気ディスクドライブに書き込みと読み込みを行い, 性能特性を見ている研究は Abutalib らや Fenggang らによって行われているが, Abutalib らによる研究はディスク空間のごく 1 部のみの評価にとどまっており, Fenggang らによる研究は Host Aware 型 SMR 型磁気ディスクドライブの評価のみとなっており, 性能特性の評価は十分に行われていない [12] [13]. そこで本研究では, Drive Managed 型 SMR ディスクドライブにおけるシーケンシャルおよびランダムな IO に対する基本性能特性の評価および Host Managed 型 SMR ディスクドライブのエミュレーション環境と IO トレースを用いて

トランザクション処理負荷に対する当該磁気ディスクドライブの性能特性の試験結果を報告し, 考察する.

1.2 本論文の構成

本論文は以下のように構成される. 第 2 章で瓦書き型磁気ディスクドライブの技術概要について解説し, 第 3 章ではドライブ管理型 SMR 磁気ディスクに対して行ったマイクロベンチマークによる性能評価の概要と結果を述べる. 第 4 章で Zoned Block Commands (zbc) およびそれを用いたホスト管理型 SMR 磁気ディスクドライブのエミュレーションについて解説し, 第 5 章ではそのエミュレーション環境を用いて従来型の磁気ディスクドライブに対して行った IO トレース実験の概要と結果を述べる. 最後に第 6 章でまとめと今後の展望について述べる.

第2章　SMR型磁気ディスクドライブ

2.1　SMR型磁気ディスクドライブの物理構造

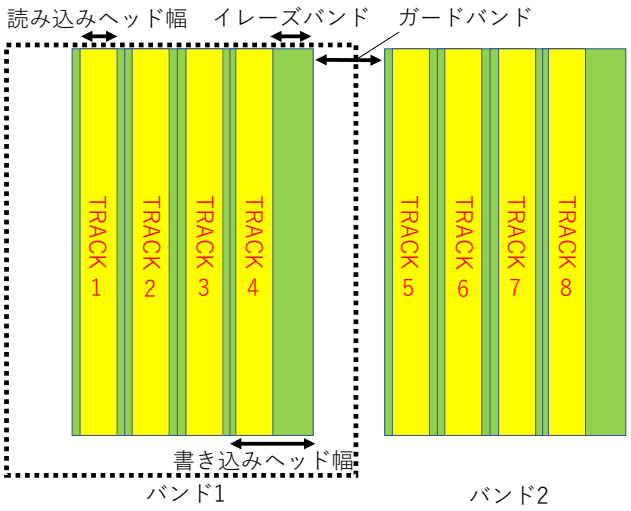

図 2.1: SMR の記録方式

瓦書き磁気ディスクドライブ (SMR ディスク) の技術は 2009 年に R.Wood らによって提唱され, 2014 年に一般向け HDD として初めて製品として販売された [2]. 図 2.1 に SMR の記録方式を示す. SMR 技術とは, 書き込み用の記録磁気ヘッドの技術的限界によりトラック幅を狭めることができないことにより記録磁気ヘッドより相対的に幅が狭い読み込み用の磁気ヘッドにトラックの幅を合わせるために, 以前に書いたトラックを少しずつずらしながら重ねて書いていく方式である. この記録方式により高密度化が可能な理由は大きく分けて 2 つあり, 強い記録磁界が得られる

ことと, 従来ディスクの記録面上に必要だった余分なスペースが減らすことができたことである. これは, 従来の方式でトラック密度を高めるためには記録磁気ヘッドの幅を狭めなくてはならなかったが, SMR 方式ではトラック幅よりも記録磁気ヘッドの幅を大きく設定できるため記録磁界を強く保つことができることにある. 後者はより複雑で, 磁気ディスクの記録密度を高めるためにはトラック間の幅 (トラックピッチ) を狭めなくてはならないが, その際には記録素子から生じる記録磁界が隣接する記録トラックに干渉して記録された情報を消去してしまう書き滲みと呼ばれる問題を考慮しなくてはならないことによる. 書き滲みによってデータが消されてしまう領域をイレーズバンドと呼ぶが, トラックピッチは通常このイレーズバンドを考慮して広めに取られている. 記録磁気ヘッドはディスクの内周部にアクセスする際と, 外周部にアクセスする際ではディスクに対しての傾きが異なり, これにより磁気ディスクの内周部では外周側の, 外周部では内周側のイレーズバンドが大きくなる. そのため従来は外周部や内周部はトラックピッチを大きく取っていた. しかし, SMR 方式ではイレーズバンドの狭い側のみを使用することが可能である. つまり, ディスクの外周では内周側に, 内周では外周側に向かって重ねて記録していけば今まで必要だったイレーズバンド間に必要な余分なスペースを減らすことができ, トラックピッチを狭めることが可能になるのだが, 余分なスペースが全く必要なくなるわけではなく, その代わりに 1 度に書き込む最小単位であるバンドと呼ばれるトラックのまとまった領域の間に存在するバンド間干渉を防ぐための仕組みであるガードバンドが必要になる.

2.2 SMR 型磁気ディスクドライブのファームウェア

次に SMR のファームウェアについて説明する. SMR ディスクを既存のディスクと同様に互換的に扱うのか, それとも新たなコマンドセットを用意し SMR ディスクに対してのアクセスを shingled なものに限定するのかといった調整を行うのがファームウェアである. ファームウェアは 3 種類が提案されており, SMR を従来の磁気ディスクとの互換性を保ちながら扱う方式を Drive Managed 方式, SMR ディスクへのワー

クロードをホストでshingledなアクセスに最適化して扱う方式をHost Managed方式, 従来のディスクと互換的に扱うかSMRディスクへのワークロードを調整するかホスト側で選択することができる方式をHost Aware方式と呼ぶ. Host Managed方式を採用すれば, SMRへのワークロードを適切に管理することができるので, SMRのshingled構造に対して精緻な制御が可能になり, 安定したパフォーマンスが実現できる. しかし, Drive Managed方式は既存のホストに対して一切の変更を加えずにSMRディスクを扱う方式であるのに対して, Host Managed方式ではまったく新しいソフトウェアが必要になり, さらにはファイルシステム, OS, ハードウェア等にも変更が必要となるため, 導入コストが非常に高い方式である. Host Managed方式の基本は, 物理的な単位であるバンドに対して, 論理的な単位であるゾーンを割り振り, ゾーン単位でのアクセスを行うことなのだが, SMRへの最適化のためHost Managed方式ではゾーンアクセスに対して様々な制約が存在する. そこでその制約を緩和することで, Drive Managed方式のように互換性を保ちながら, Host Managedのようなゾーン単位でのアクセスを可能にした方式がHost Aware方式である. Host Aware方式はその特徴からDrive Managed方式とHost Managed方式の長所と短所を併せ持っている. 現在はDrive Managed方式が主流であるので以降はDrive Managed方式を前提として説明する.

2.3 Shingled Translation Layer (STL)

Drive Managed方式のSMRではデータの記録方式も従来と異なっている. 従来のデータ記録方式は, OSから受け取ったデータはディスクに内蔵されたメモリ上に蓄えられ, ディスクコントローラーがアクセス時間を最小にするように, 記録するデータのスケジューリングを行った後に実際に記録が行われていた. 一方, SMR方式のデータ記録方式は, OSから受け取ったデータをメモリ上に蓄えるところまでは変わらないが, 新たにメディアキャッシュと呼ばれるキャッシュをディスクに内蔵し, バッファメモリ上のデータをメディアキャッシュへと書き移す. SMRはその特性上ランダム書き込みはできないので, メディアキャッシュ上のデータはシーケンシャル書き

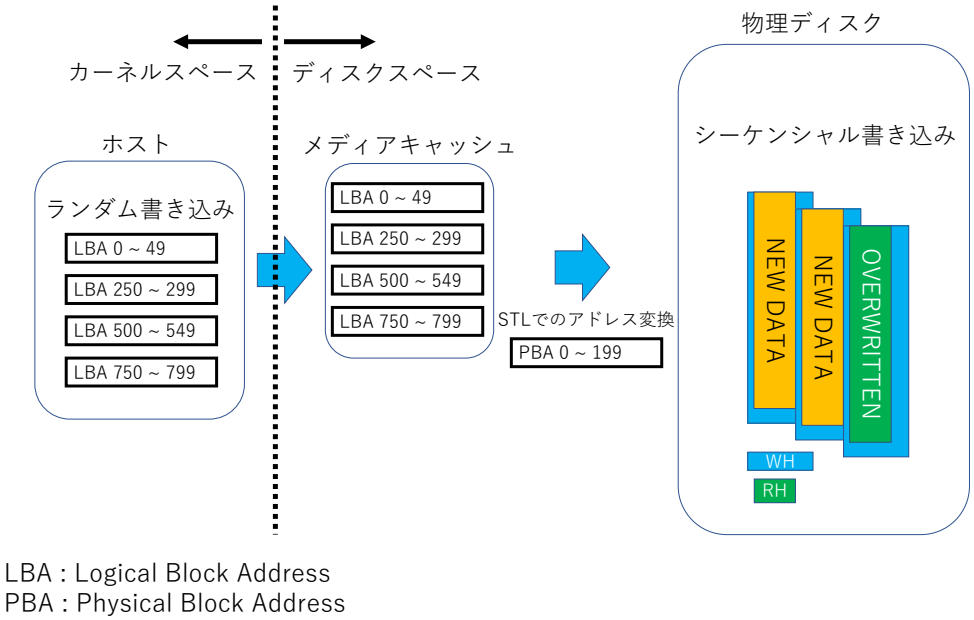

図 2.2: Shingled Translation Layer での動作

込み用に再構成され, バンド単位で書き込みを行う. 記録済みのデータを書き換える場合は, そのデータが存在するバンドのデータを 1 度メディアキャッシュ上に読み出し, メディアキャッシュ上で書き換える部分を書き換えた後, バンドに書き込みを行うという方式になっている. この際に, 元のバンドに書き込む場合に比べ, 新しいバンドに書き込む場合の方が高速であるが, 不要になったデータを削除するためのガベージコレクションの仕組みが必要になってくるため, ディスクコントローラに多少複雑な機構が必要になってくる. ガベージコレクションなどのこれらの SMR 独自の操作が行われるのが Shingled Translation Layer (STL) と呼ばれるレイヤで行われ, 図 2.2 のように最終的に物理ブロックアドレスへと変換がなされてディスクへ書き込まれる.

第3章　SMR型磁気ディスクドライブの基本性能の測定

3.1　性能試験手法

表 3.1: マイクロベンチマーク性能試験環境

CPU	Intel(R) Xeon(R) CPU E3-1240 v5 @ 3.50GHz
Memory	DDR4 8192MB × 2
OS	CentOS release 6.8 (Final)
Kernel	2.6.32-642.4.2.el6
HDD	Seagate Archive Disk 8TB × 2

本節で述べる性能試験はSMRに書き込みを行った時, STLがどのような影響を及ぼすのかを確認することを目的とするものである. 性能試験は表3.1のような環境で行った. SMRディスクは同じ型番のものを2種類用いて計測を行い, まず初期状態においての2つのディスクのアクセスパターンをシーケンシャル読み込みとランダム読み込みによって計測した. 計測の際はO_DIRECTによってOSのキャッシュは介さないようにし, 磁気ディスクのバッファキャッシュについてはある場合とない場合の計4通り計測した. 計測に於いては, バッファキャッシュがある場合とない場合の2通りについて計測を行ったが, 同様の傾向が見られたので以下ではバッファキャッシュがない場合の結果についてのみ触れる.

図3.2と図3.3は初期状態の瓦書き磁気ディスクに対して, 先頭セクタ10GB分を1MB単位でシーケンシャル読み込みを行った際の結果及び, ランダム読み込みをディ

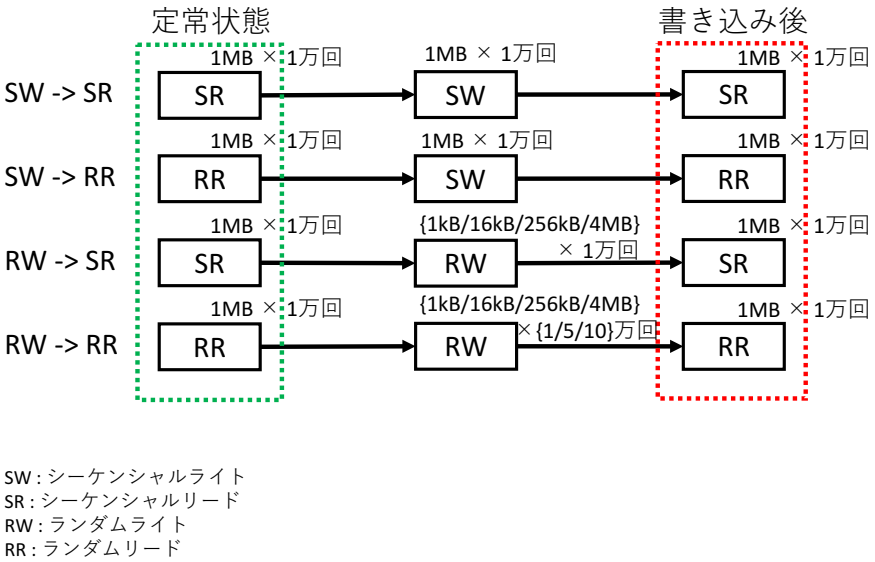

図 3.1: マイクロベンチマーク負荷

スク全体に対して 1MB 単位で 10GB 分行った結果であり, 縦軸は 1MB を読むのにかかった時間 (ms) を, 横軸は読み込んだセクタ番号をそれぞれ表している. 結果については両方のディスクで同様の結果が得られたので, 片方のディスクでの結果のみを示している.

初期状態において両方のディスクで同様の結果が得られたので双方のディスクにそれぞれ違う書き込み負荷を与えて, 読み込み性能の変化を観測した. 図 3.1 に実験に用いた負荷を示す. 読み込み性能の変化は, 書き込みを行う直前の読み込みによるレイテンシと書き込み直後の読み込みによるレイテンシを比較することで評価し, 書き込みと読み込みはそれぞれランダムとシーケンシャルの 2 通りずつ行い, 計 4 通り行った. シーケンシャル書き込みは先頭セクタから 1MB 単位で 10GB, ランダム書き込みはディスク全体に対して 1kB, 16kB, 256kB, 4MB の 4 種類の単位でそれぞれがほぼ同じ回数書き込まれるようにそれぞれ 10GB ずつ書き込み, シーケンシャ

ル読み込みは先頭セクタから 1MB 単位で 10GB, ランダム読み込みはディスク全体に対して 1MB 単位で 10GB 読み込んだ．さらにランダム書き込み後のランダム読み込みについては定常状態に落ち着くまでの時間を見るために, 書き込み量を 1GB, 5GB, 10GB の 3 通りに変化させて読み込みレイテンシの時間変化の観測を行った．ここでいう定常状態とはメディアキャッシュに書き込みデータが残っていない状態のことを指す．

3.2 実験結果

図 3.4 から図 3.7 は書き込み負荷を与える前後の読み込み性能を示している．図中の緑の系列が初期状態を赤の系列が書き込み直後のレイテンシをそれぞれ表しており, 横軸はセクタ番号を縦軸はレイテンシを表している．また, 図のキャプションの SW はシーケンシャル書き込み, RW はランダム書き込み, SR はシーケンシャル読み込み, RR はランダム読み込みをそれぞれ表しており, SW → SR はシーケンシャル書き込みの前後にシーケンシャル読み込みを行っていることを表している．図 3.8 から図 3.11 は図 3.4 から図 3.7 の実験におけるそれぞれのレイテンシの割合を cumulative curve で示しており, 横軸はレイテンシを縦軸はパーセンテージを表している．図 3.12 から図 3.14 は RW → RR において書き込み量を変化させた場合の時間経過に伴う読み込み性能の変化を示しており, 横軸は書き込みが終了した時刻を 0 秒とした経過時間を縦軸はレイテンシを表している．

図 3.4 から図 3.11 によると SW → SR, RW → SR では有意な性能低下は見られないのに対して, SW → RR では若干のレイテンシの増大が見られ, RW → RR では著しいレイテンシの増大が見られる．またランダム読み込みにおいてはいずれも高遅延の増大が見られ, SW → RR では 100ms 以上の高遅延が 3%に, RW → RR では 15%に増大していた．図 3.12 から図 3.14 によると書き込み終了後は読み込みレイテンシのバースト的な増大が観測された．このバーストが開始する時間は書き込み量によって再現性があるのである程度予測が可能であり, バーストの継続時間は書き込み量が増えるに従って増大することが分かった．SMR は従来型の磁気ディスクド

ライブとは異なるレイテンシの増大現象が見られるため, データベース処理において大量の書込みの後に読込みレイテンシが増大する可能性があり, 性能管理上考慮が必要である.

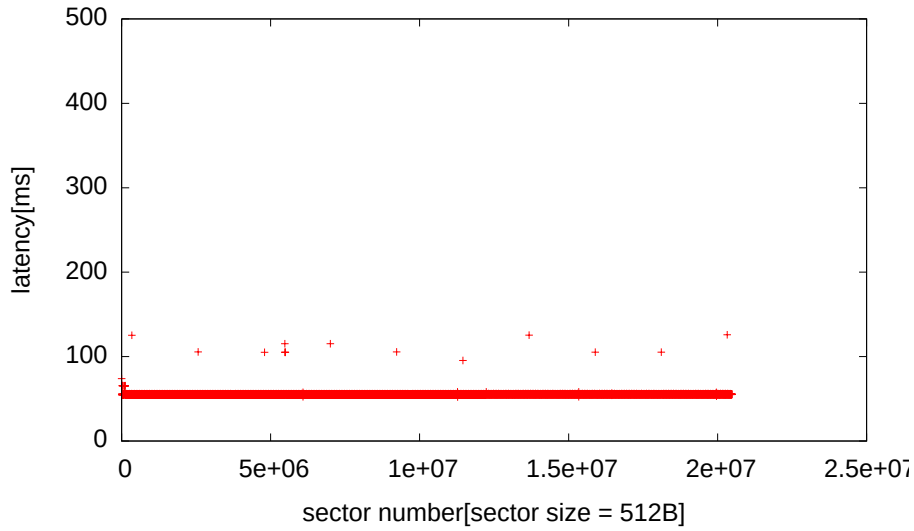

図 3.2: 初期状態に於けるシーケンシャル読み込み

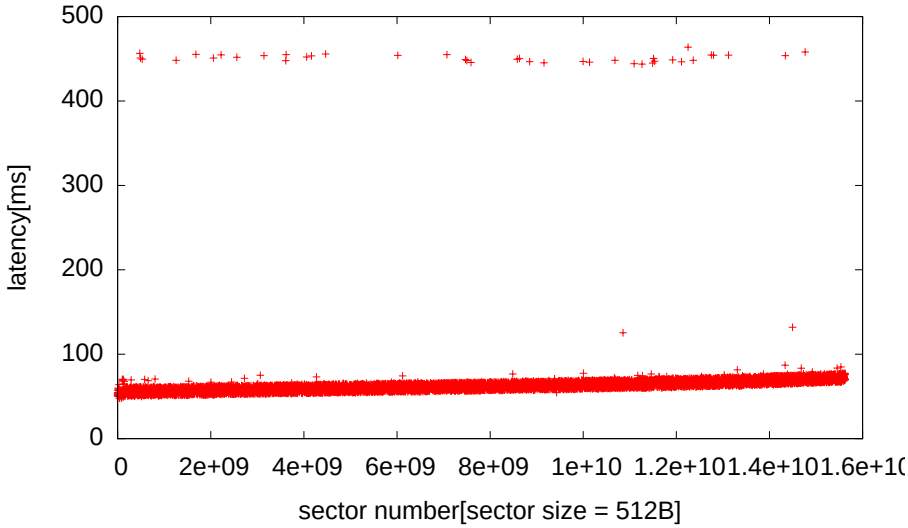

図 3.3: 初期状態に於けるランダム読み込み

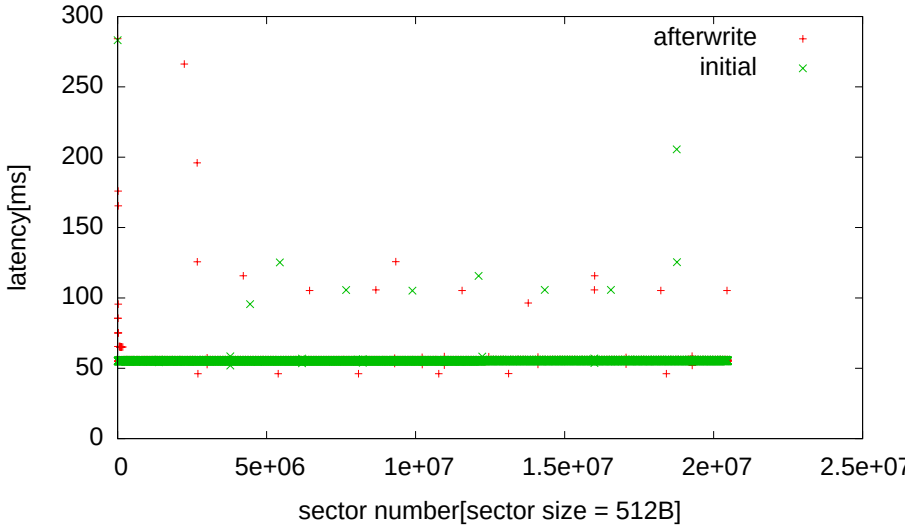

図 3.4: SW → SR 実行時の読み込み性能の変化

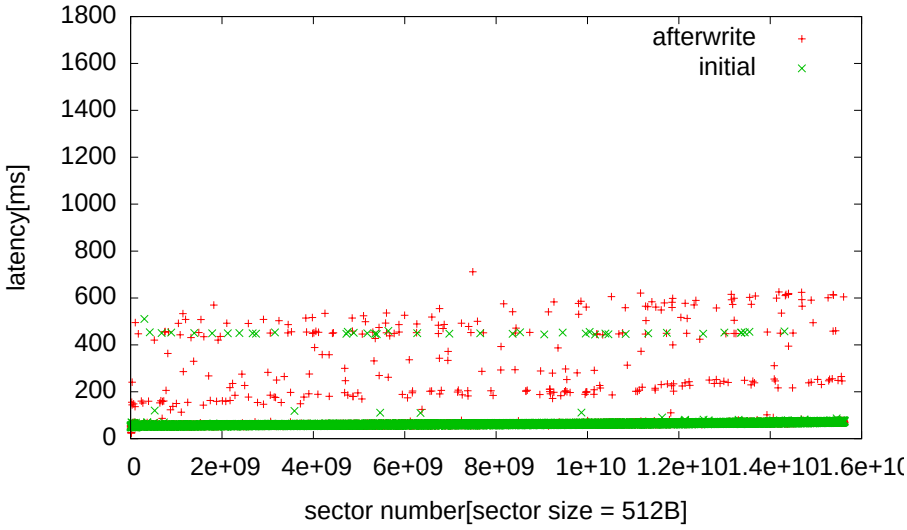

図 3.5: SW → RR 実行時の読み込み性能の変化

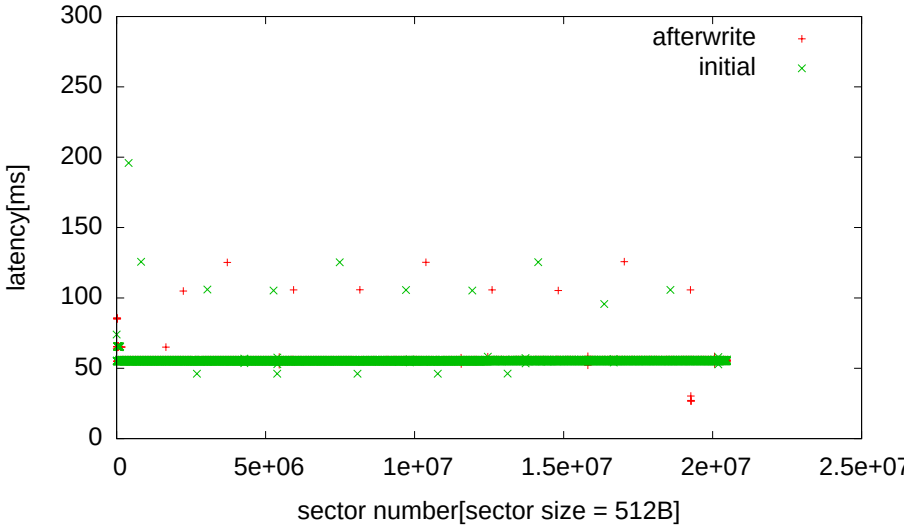

図 3.6: RW → SR 実行時の読み込み性能の変化

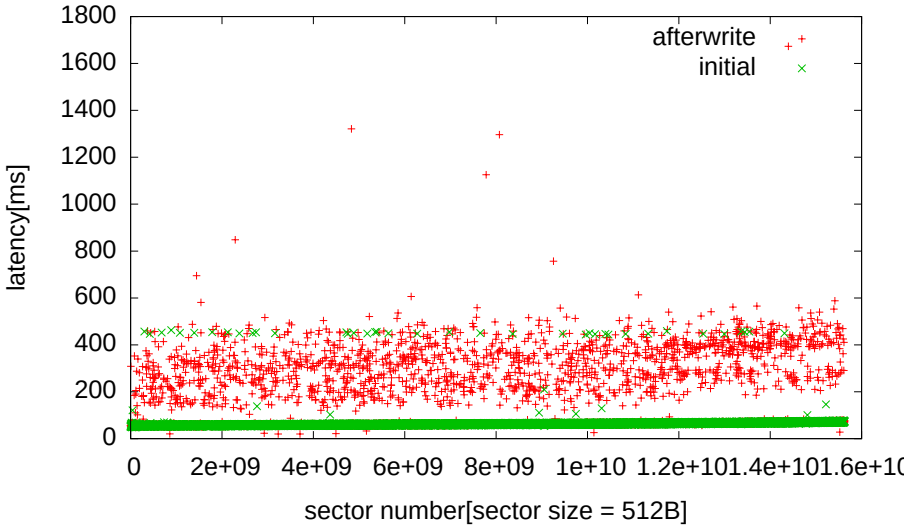

図 3.7: RW → RR 実行時の読み込み性能の変化

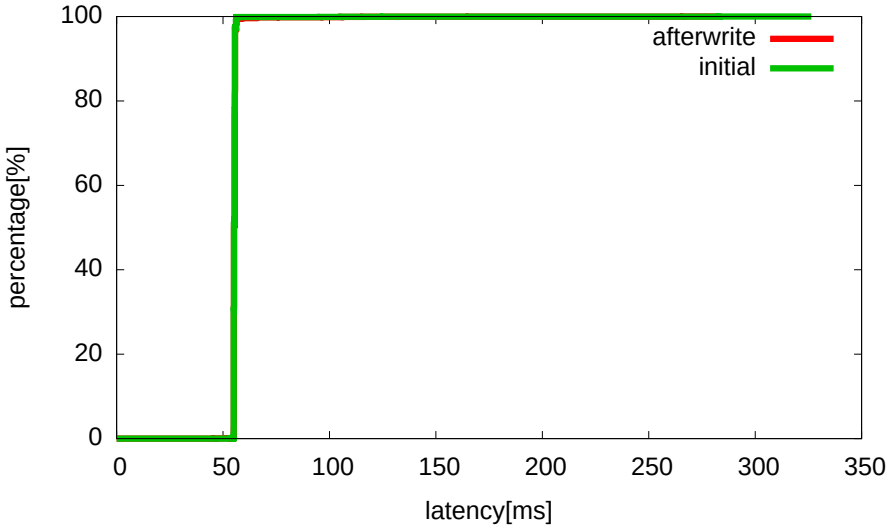

図 3.8: SW → SR 実行時の読み込み性能の変化 (cumulative curve)

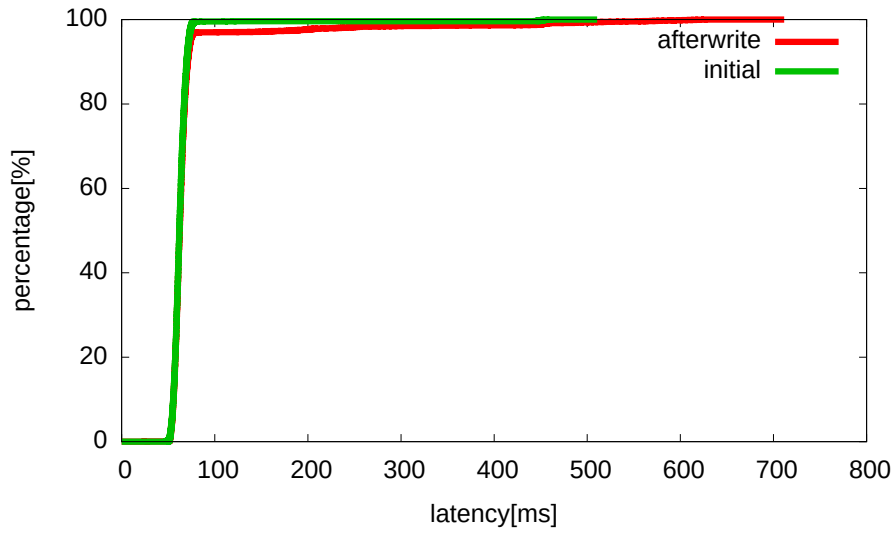

図 3.9: SW → RR 実行時の読み込み性能の変化 (cumulative curve)

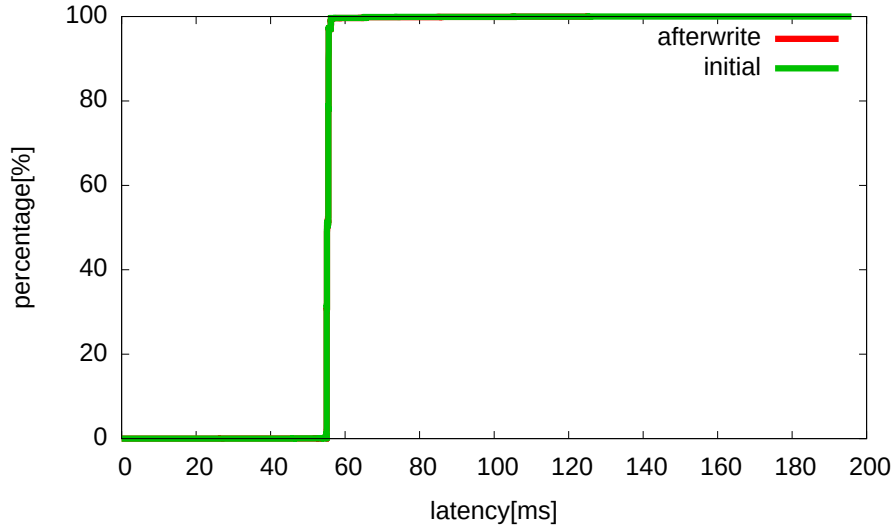

図 3.10: RW → SR 実行時の読み込み性能の変化 (cumulative curve)

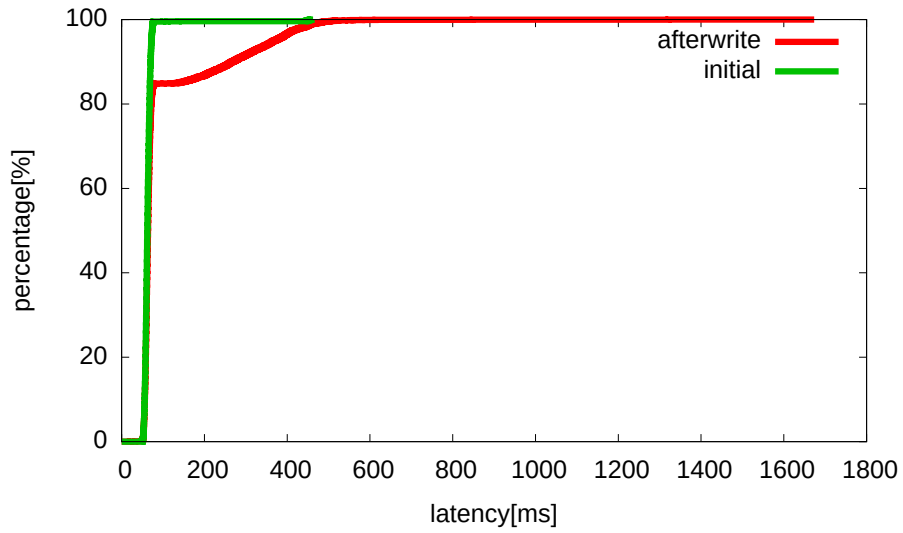

図 3.11: RW → RR 実行時の読み込み性能の変化 (cumulative curve)

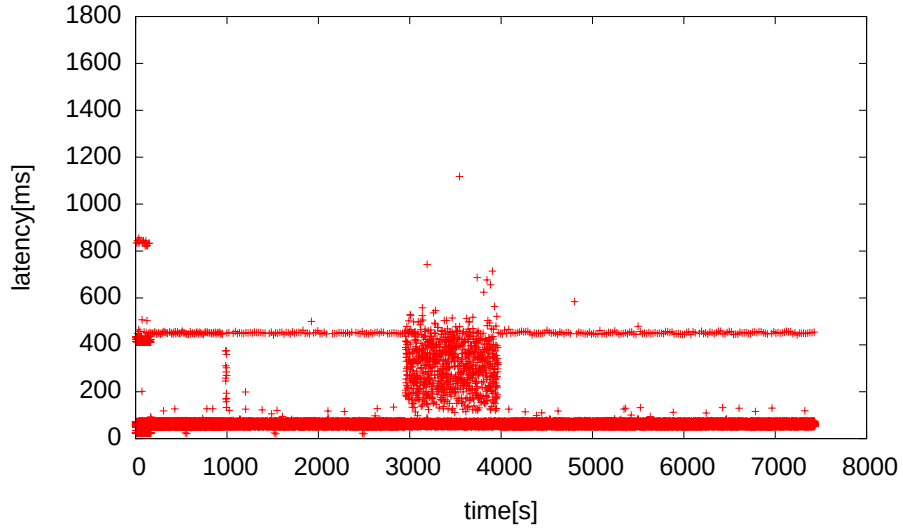

図 3.12: RW → RR の書き込み量に対する読み込み性能の時間変化 (書き込み量 1G)

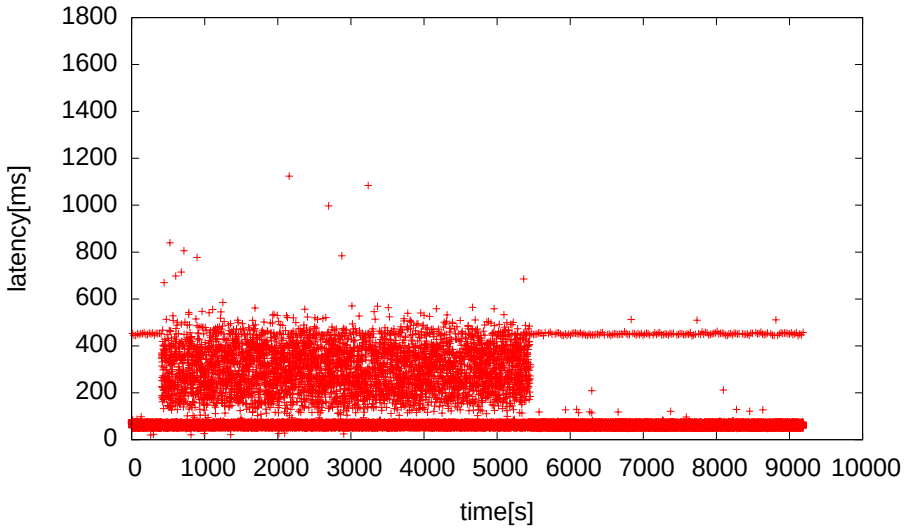

図 3.13: RW → RR の書き込み量に対する読み込み性能の時間変化 (書き込み量 5G)

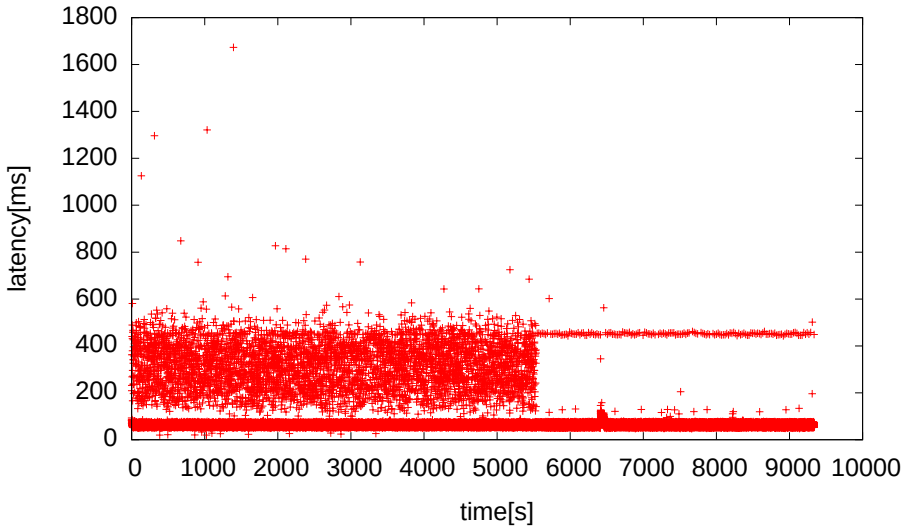

図 3.14: RW → RR の書き込み量に対する読み込み性能の時間変化 (書き込み量 10G)

第4章　Host-Managed SMR ディスクドライブの性能エミュレータ

4.1　SMR 型磁気ディスクドライブの論理構造

表 4.1: ゾーンの種類と特徴

ゾーンタイプ	ファームウェア	特徴
Conventional Zone	Host Aware & Host Managed	従来のディスクと同様に扱える
Sequential Write Required Zone	Host Managed	書き込みは必ずシーケンシャル
Sequential Write Preferred Zone	Host Aware	書き込みはシーケンシャルを推奨

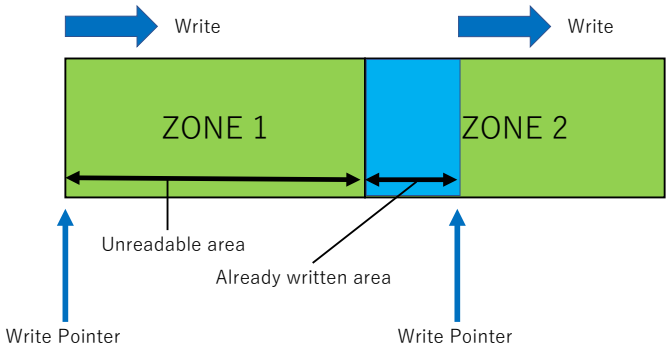

図 4.1: ゾーンの構造

SMR 型磁気ディスクドライブの物理構造については第 2 章で解説した．その際 Host Aware 方式と Host Managed 方式では物理的な単位であるバンドに対して論理的な単位であるゾーンを割り振りディスクへのアクセスを実現していると述べたが, この節ではゾーンについてより仔細に述べたいと思う．まず, 表 4.1 に示したようにゾーンは 3 つの種類に分類されそれぞれ Conventional Zone, Sequential Write Preferrd Zone, Sequential Write Required Zone に分かれている．Conventional Zone は従来のディスクと同様に扱うことができ, ランダムな書き込みも自由に行える．これは, Host Aware と Host Managed のどちらのファームウェアにおいても使用することができるとされているが, Host Managed 方式のディスクドライブにおいてはハードウェアレベルでバンドと隔離されたトラックを用意して使用するため, このゾーンタイプを使用できたとしても小さな容量しか用いることができず, あくまでメタデータ等の管理など用途を限定して使用する必要がある．Sequential Write Required Zone は Host Managed 方式のディスクで用いられるゾーンである．このゾーンではシーケンシャルな書き込みのみしか受け付けず, ランダムな書き込みはエラーになる．Sequential Write Preferred Zone は Host Aware 方式のみで使用することができるゾーンタイプで, 基本的には Sequential Write Required Zone と同様にシーケンシャルな書き込みのみしか受け付けないが, ランダムな書き込みが行われた場合にはそれ以降 Conventional Zone として扱うというゾーンである．

Sequential Write Required Zone における書き込みの様子を図 4.1 に示した．これらのゾーンについて語る上で重要な概念が Write Pointer(WP) である．この Write Pointer(WP) によって書き込みと読み込みを制限している．つまり書き込みは WP が指す位置からのみ行うことができ, それ以外の場所からの書き込みは Sequential Write Required Zone においては全くできず, また読み込みについてもゾーンの開始位置から WP が指す位置までのデータしか読めなくなっている．また書き込みと読み込みはゾーン単位で行わなくてはならず, ゾーンを超える書き込みや読み込みはできない．WP はゾーンの状態の管理にも用いられており, WP がゾーンの初めに位置しているときは Empty, WP がゾーンの中ほどに位置しているときは Open, WP がゾーンの終端のに位置しているときは Full と大きく分けて 3 つの状態に分か

れる. ゾーンが Open と Full の状態のときは WP をゾーンの初期位置に戻すために WP の位置を Reset することができる. WP の位置の Reset の際に, Sequential Write Preferred Zone においては古いデータをすべて 0 で上書きするのに対し, Sequential Write Required Zone では古いデータは読めなくするのみでありその処理方法は異なる.

4.2 Zoned Block Command (zbc) 概要

表 4.2: zbc の主要関数と概要

関数名	概要
zbc_open	デバイスを open し zbc デバイスディスクリプタの取得を行う
zbc_close	デバイスを close し zbc デバイスディスクリプタの開放を行う
zbc_pwrite	セクタ単位で書き込みを行う
zbc_pread	セクタ単位で読み込みを行う
zbc_zone_operation	WP の位置のリセット等のゾーンの状態の管理を行う

Zoned Block Command (zbc) は T10 委員会によって標準化がなされている Host Aware 方式, Host Managed 方式 SMR ディスクに対してゾーン構造に合わせた書き込み及び読み込みを行うための SCSI コマンドに代わるコマンドセットであり, Western Digital 社の Damien Le Moal らによってその仕様に従ったものが実装されている [5] [14]. 表 4.2 に zbc コマンドセットにおける主要な関数とその機能について示した. zbc_open と zbc_close は従来の open, close コマンドと同等の機能を持っている関数であり, それぞれデバイスディスクリプタの取得と開放を行う. zbc_pwrite と zbc_pread は従来の pwrite, pread コマンドと同等の機能を持っている関数であるが, 節で述べたように論理空間では複数のゾーンに分かれており, それぞれが Write Pointer を有しているため書き込みと読み込みに制限を加える必要があるため, これらの関数を用いる際にはそれらを検証するための機構を別途設ける必要があり注意が必要である. zbc_zone_operation は zbc コマンドセット特有のものであり, ゾーン

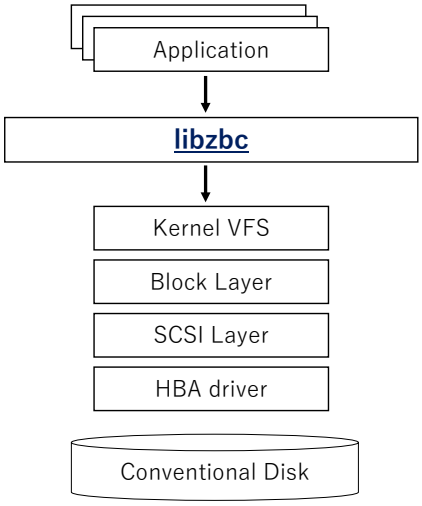

図 4.2: Host Managed 方式ディスクのエミュレーション

の状態の管理を行う機能を提供する．Write Pointer の位置の reset もこの関数を用いて行う．

また, 実装された zbc コマンドセットには図 4.2 のように従来型のディスクを Host Managed 方式 SMR ディスクとしてエミュレーションして zbc コマンドを用いてアクセスする機能が備わっている．第 5 章で行う実験はこのエミュレーション機能を用いて行う．

第5章　入出力トレースを用いた測定

5.1　性能特性測定試験手法

表 5.1: SMR ディスクエミュレータ測定試験環境

CPU	Intel(R) Xeon(R) CPU E3-1240 v5 @ 3.50GHz
Memory	DDR4 8192MB × 2
OS	CentOS Linux release 7.4.1708 (Core)
Kernel	4.14.10-1.el7.elrepo.x86_64
Emulation target HDD	HGST 4TB (HUS726040ALA610)
Libzbc	5.4.1

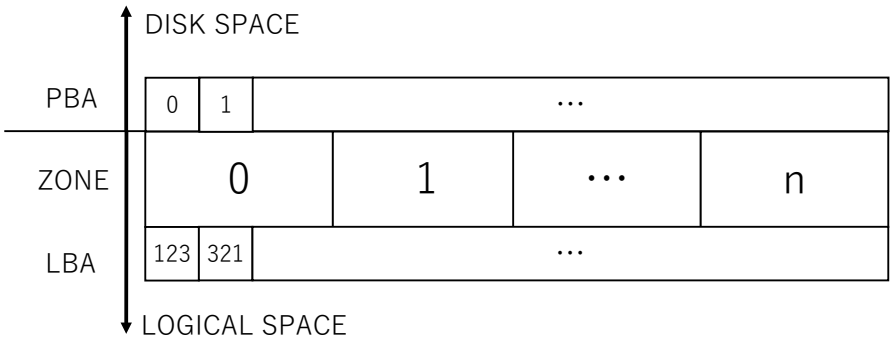

図 5.1: LBA と PBA の対応

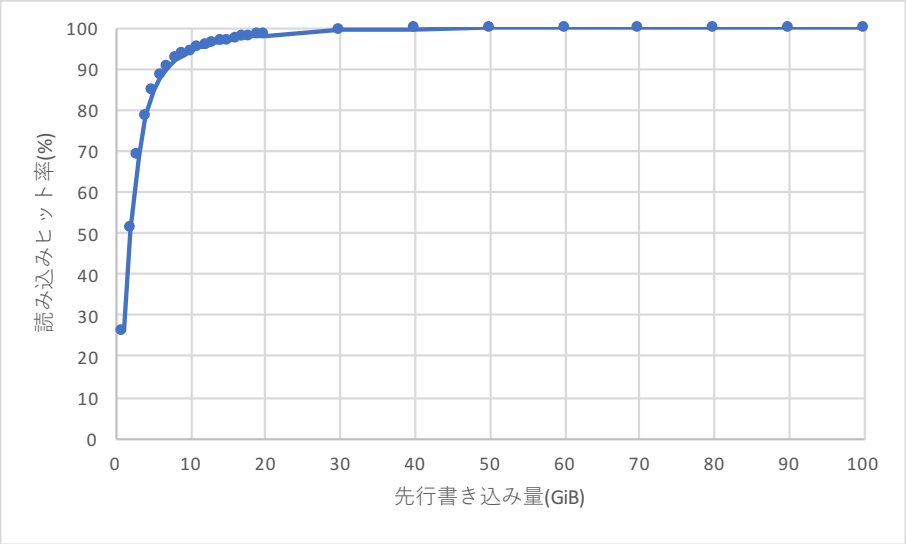

図 5.2: 先行書き込み量の決定

表 5.2: 性能試験の条件と結果の対応

先行書き込み数	畳み込み	バッファサイズ	計測項目	図番号
100 万	なし	0	all	図 5.3 - 図 5.7
1000 万	なし	0	all	図 5.8 - 図 5.12
100 万	あり	10	all	図 5.14 - 図 5.18
1000 万	あり	10	all	図 5.19 - 図 5.23
100 万	あり	100	all	図 5.24 - 図 5.28
1000 万	あり	100	all	図 5.29 - 図 5.33
100 万	あり	1000	all	図 5.34 - 図 5.38
1000 万	あり	1000	all	図 5.39 - 図 5.43
100 万	あり	10000	all	図 5.44 - 図 5.48
1000 万	あり	10000	all	図 5.49 - 図 5.53

表 5.3: IO 数と容量の対応

先行書き込み数 / 計測 IO 数	先行書き込み容量	計測書き込み容量	計測読み込み容量
100 万 / 10 万	3.8GiB	123MiB	268MiB
100 万 / 30 万	3.8GiB	369MiB	802MiB
100 万 / 100 万	3.8GiB	1.2GiB	2.6GiB
100 万 / 300 万	3.8GiB	3.6GiB	7.8GiB
1000 万 / 10 万	38GiB	123MiB	268MiB
1000 万 / 30 万	38GiB	369MiB	802MiB
1000 万 / 100 万	38GiB	1.2GiB	2.6GiB
1000 万 / 300 万	38GiB	3.6GiB	7.8GiB

本節で述べる性能試験は Host Managed 方式 SMR ディスクの入出力性能特性を確認することを目的としたものである. 測定試験は表 5.1 のような環境で行った. また, その他の条件として HDD のディスクキャッシュおよびリードアヘッドを ON にし, linux のページキャッシュも使用した. さらにエミュレータの設定としてゾーンのサイズを 256MiB, Conventional Zone の数は 0 としすべてのゾーンが Sequential Required Zone となるようにした. Host Managed 方式 SMR ディスクでは図 5.1 のように実際の物理ディスクに割り当てられた PBA は 0 から順番に割り当てられているのに対して, ゾーンに割り当てられた LBA はホスト側で割り当てるため, これらの対応を記憶しておく必要があるが, 本性能試験では LBA と PBA のアドレステーブルをメモリ上に展開し試験を行った. その際のアドレス空間は 1TiB とした. またここでいう PBA とは物理ディスク上にあらかじめ割り当てられた物理ブロックアドレスのことではなく, LBA を連続した LBA へと変換したものを PBA と定義する. 性能試験は TPC-C の IO トレースを用いて行った [15]. この際アドレス空間を 1TiB としたため, トレースに含まれるアドレスは 1TiB で丸め, SMR の物理アドレスサイズに合わせるため 4kiB aligned とした. また, TPC-C はオンライントランザクション処理ベンチマークの 1 種でありディスクの広範囲に対して読み込みと書き込みが発生するが, 性能試験開始時にはディスクの中身が空であり Sequential Required Zone

はデータが何も書かれていない時には読み込みができないのでまずトレースのデータの一部を使用して先行的に書き込みだけを行うようにし, その後読み込みを含むトレースの IO で計測を行った. 先行書き込みの数は 100 万回, 1000 万回の 2 通り, 計測 IO の数は 10 万回, 30 万回, 100 万回, 300 万回の 4 通りに変化させ, IO の内訳 (読み込み成功数, 読み込みエラー数, 書き込み数) およびその割合, 実行時間, IOPS, スループットを計測しグラフにした. 先行書き込みの数は先行的に 1GiB から 20GiB まで 1GiB 刻みで, 30GiB から 100GiB まで 10GiB 刻みで書き込みを行った後 15 万回の IO トレースを再生して読み込み成功数を調査する予備実験を事前に行うことにより決定した. 予備実験の結果を図 5.2 に示す. 横軸が先行書き込み量で縦軸は読み込み成功割合である. この結果において読み込み成功率が約 80%と約 99.5%になるように先行書き込み量を決定した. また本章で行うすべての実験の条件と結果の対応を表 5.2 にまとめた. 表 5.2 の計測項目 all については 5.2 節において後述する 5 種類の指標すべてを計測したことを表している. また, 先行書き込み数と計測 IO 数とそれらをバイト換算したものを表 5.3 にまとめた.

5.2 性能試験結果

性能試験の結果を図 5.3 から図 5.12 に示す. 図 5.3 から図 5.7 までが先行書き込み数 100 万回の結果を, 図 5.8 から図 5.12 までが先行書き込み数 1000 万回の結果を示している. 横軸は計測 IO 数で縦軸は図 5.3 と図 5.8 が読み込み成功数, 読み込みエラー数, 書き込み数を, 図 5.4 と図 5.9 が読み込み成功数, 読み込みエラー数, 書き込み数のそれぞれの計測 IO 数に占める割合を, 図 5.5 と図 5.10 は書き込みと読み込みの実行時間を, 図 5.6 と図 5.11 は書き込みと読み込み (読み込みエラーは除く) の IOPS を, 図 5.7 と図 5.12 は書き込みと読み込みのスループットを示している. これらの結果から, 先行書き込み数 100 万回では計測 IO 数の上昇に伴い IOPS が減少するが, 先行書き込み数 1000 万回では計測 IO 数の上昇にともない IOPS が上昇することが分かった. スループットについても同様のことが言える. 先行書き込み数 100 万回の結果においては非常に高い IOPS, スループットが観測されたがこれはページ

キャッシュによる効果だと思われる.

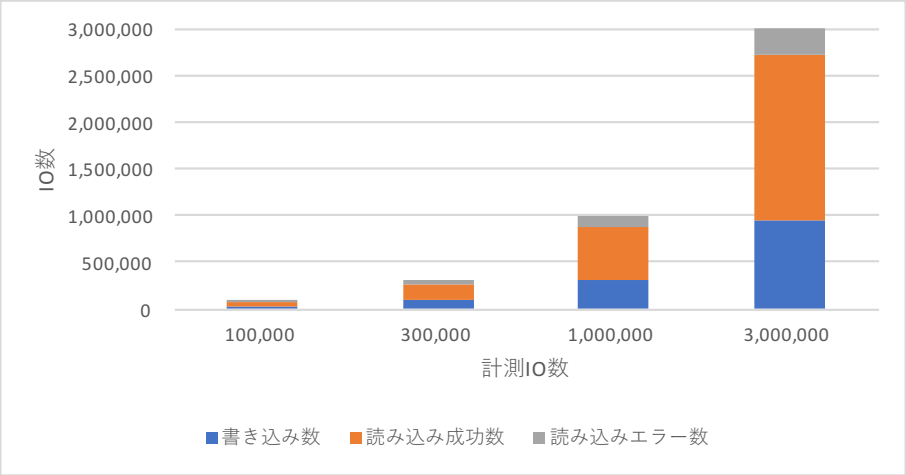

図 5.3: 計測 IO 数に対する各種 IO の内訳 (先行書き込み 100 万回)

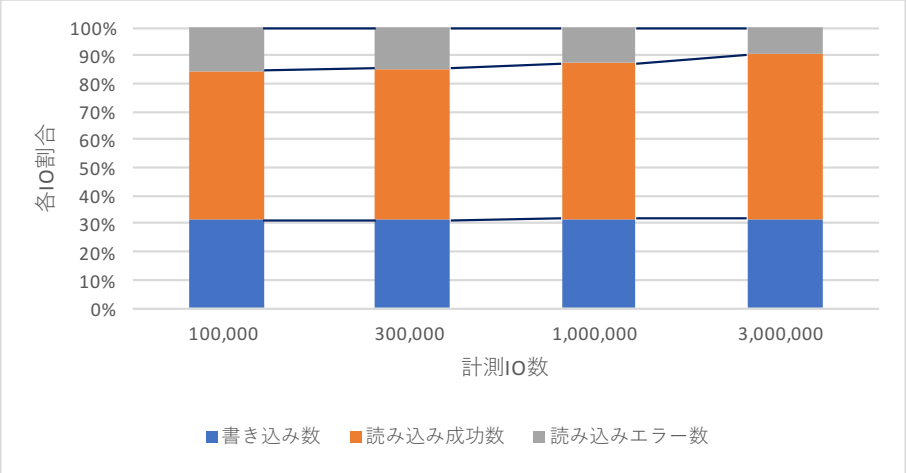

図 5.4: 計測 IO 数に対する各種 IO の割合 (先行書き込み 100 万回)

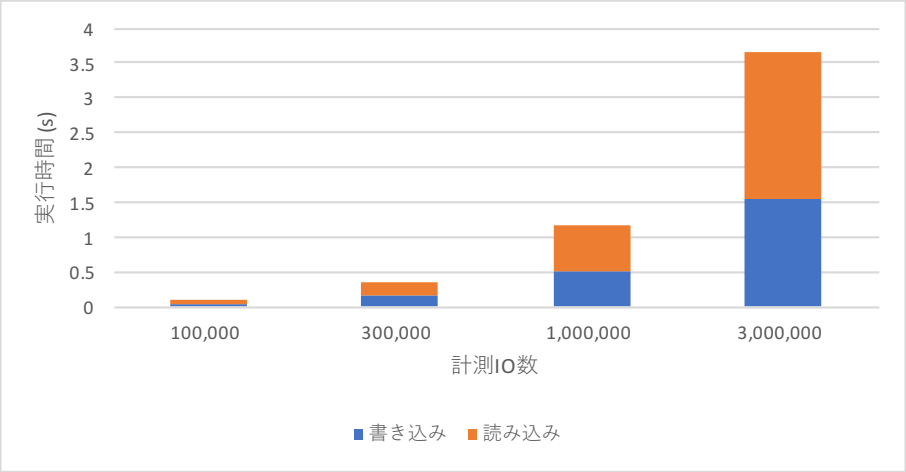

図 5.5: 各 IO 数の計測に要した実行時間 (先行書き込み 100 万回)

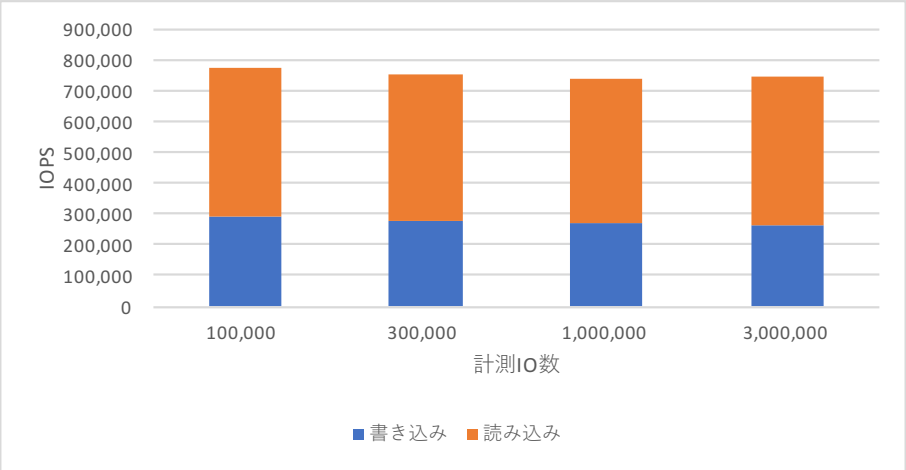

図 5.6: 各 IO 数における計測時の IOPS (先行書き込み 100 万回)

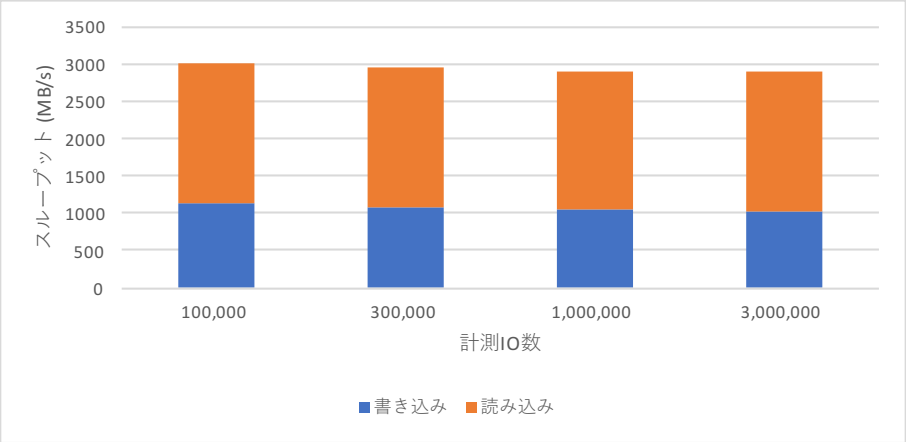

図 5.7: 各 IO 数における計測時のスループット (先行書き込み 100 万回)

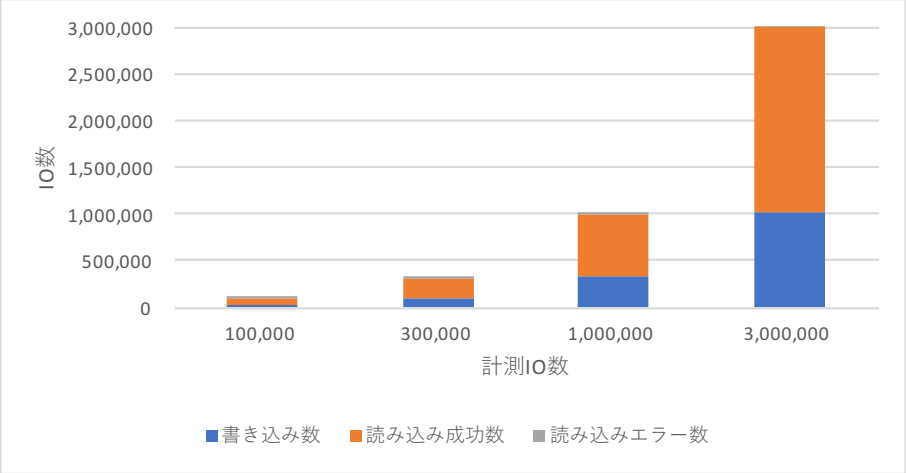

図 5.8: 計測 IO 数に対する各種 IO の内訳 (先行書き込み 1000 万回)

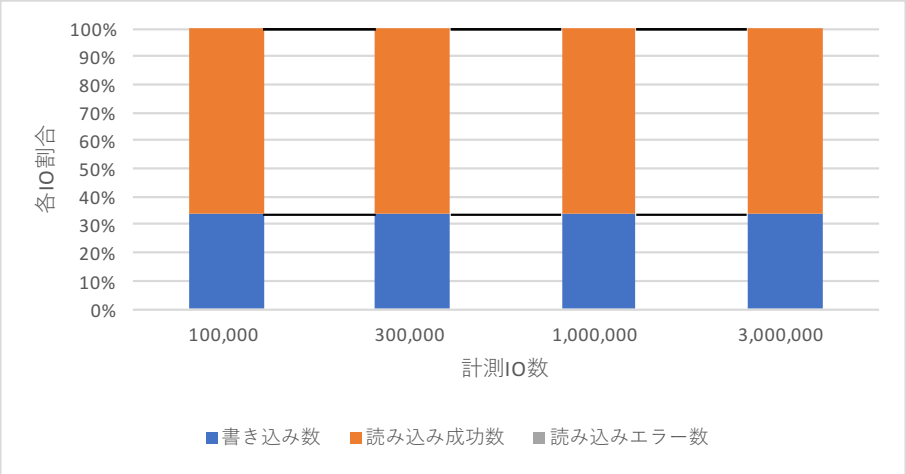

図 5.9: 計測 IO 数に対する各種 IO の割合 (先行書き込み 1000 万回)

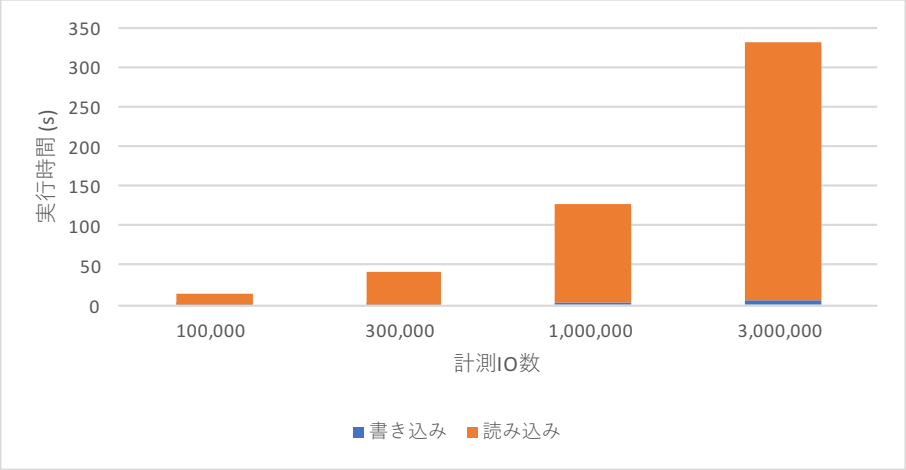

図 5.10: 各 IO 数の計測に要した実行時間 (先行書き込み 1000 万回)

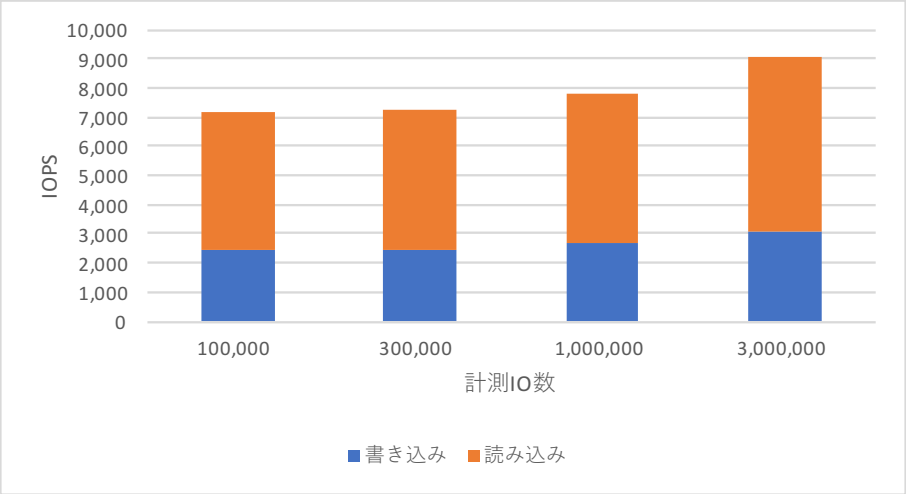

図 5.11: 各 IO 数における計測時の IOPS (先行書き込み 1000 万回)

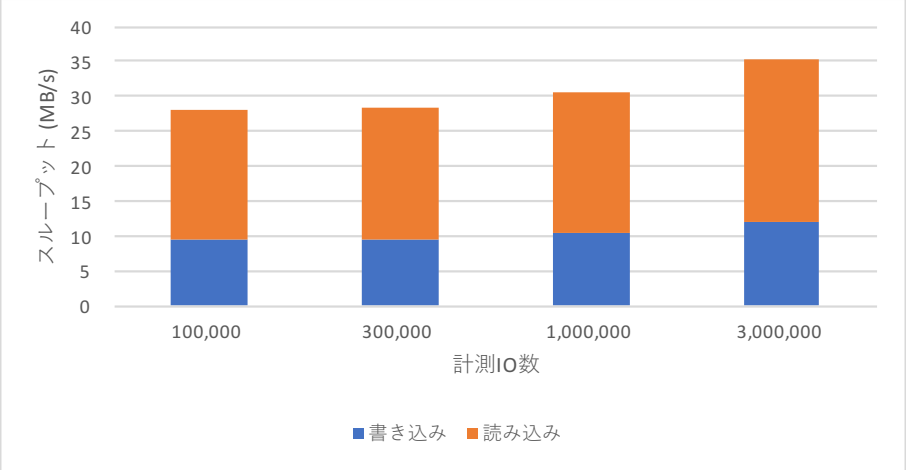

図 5.12: 各 IO 数における計測時のスループット (先行書き込み 1000 万回)

5.3 書き込みの畳み込み試験手法

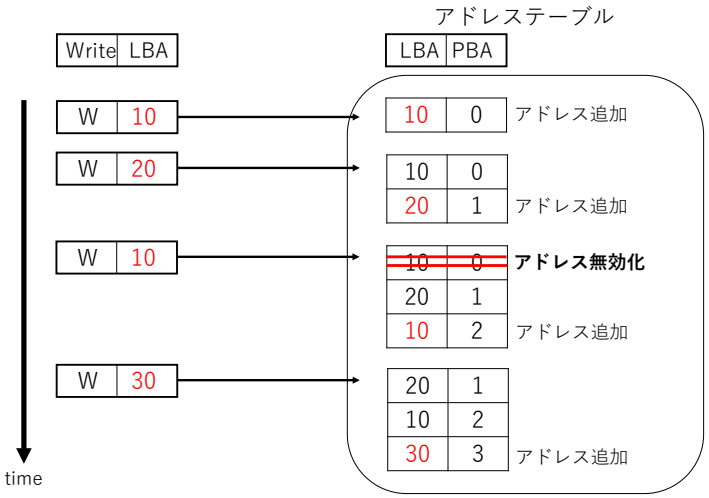

図 5.13: Copy On Write アドレス変換

5.1 と同様の条件で, 今度は書き込み要求が発行されても即座にディスクに書き込みを行うのではなく, メモリ上にバッファを用意してバッファに一時的に格納しバッファが一杯になったらしてバッファをフラッシュしてディスクに書き込みを行うという方法で計測を行った. 手法について図 5.13 に示す. 書き込みについては Copy On Write で行った. 読み込みについてもバッファ上に格納されている LBA についてはディスクへの読み込みを避けるようにした. バッファサイズは 10 個, 100 個, 1000 個, 10000 個の書き込みを保持するように変化させて行った.

5.4 書き込みの畳み込みを行った測定結果

性能試験の結果を図 5.14 から図 5.53 に示す. 図 5.14 から図 5.23 までがバッファサイズが 10 の時の結果を, 図 5.24 から図 5.33 までがバッファサイズが 100 の時の結果を, 図 5.34 から図 5.43 までがバッファサイズが 1000 の時の結果を, 図 5.44 から図

5.53 までがバッファサイズが 10000 の時の結果をそれぞれ示している. 横軸はすべて計測 IO 数とし, 縦軸は図 5.14, 図 5.19, 図 5.24, 図 5.29, 図 5.34, 図 5.39, 図 5.44, 図 5.49 が読み込み成功数, 読み込みエラー数, 書き込み数を図 5.15, 図 5.20, 図 5.25, 図 5.30, 図 5.35, 図 5.40, 図 5.45, 図 5.50 は読み込み成功数, 読み込みエラー数, 書き込み数のそれぞれの計測 IO 数に占める割合を, 図 5.16, 図 5.21, 図 5.26, 図 5.31, 図 5.36, 図 5.41, 図 5.46, 図 5.51 は書き込みと読み込みの実行時間を, 図 5.17, 図 5.22, 図 5.27, 図 5.32, 図 5.37, 図 5.42, 図 5.47, 図 5.52 は書き込みと読み込み (読み込みエラーは除く) の IOPS を, 図 5.18, 図 5.23, 図 5.28, 図 5.33, 図 5.38, 図 5.43, 図 5.48, 図 5.53 は書き込みと読み込みのスループットを示している. 先行書き込み 1000 万回においては書き込みの畳み込みを行った結果, IOPS とスループットは上昇傾向にあるが, 先行書き込み 100 万回においては IOPS とスループットは増減をしたり減少傾向にあったりと一定の挙動を示していない. これはページキャッシュが原因であると考えられるため, 次に O_DIRECT を用いて実験を行う.

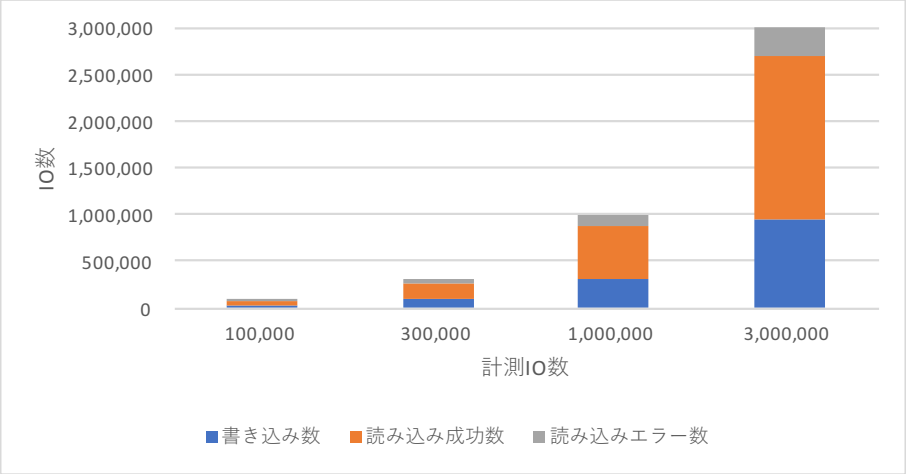

図 5.14: 書き込み畳み込みを用いた際の各種 IO の内訳 (バッファサイズ 10, 先行書き込み 100 万回)

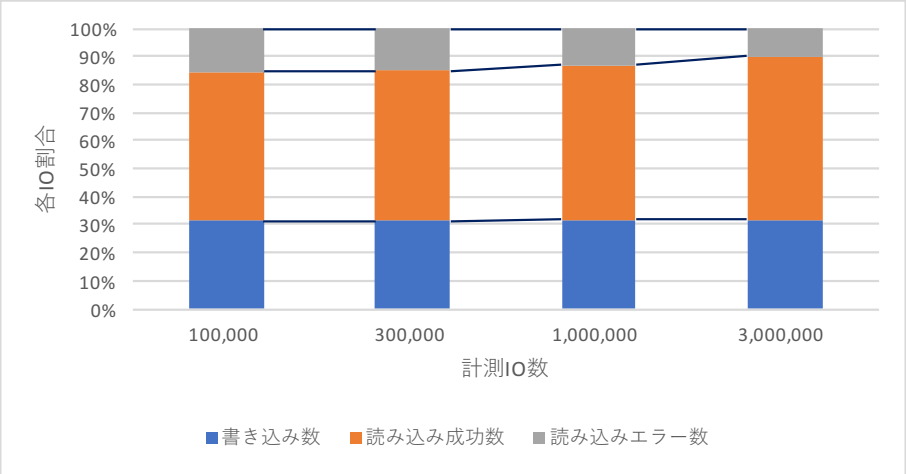

図 5.15: 書き込み畳み込みを用いた際の各種 IO の割合 (バッファサイズ 10, 先行書き込み 100 万回)

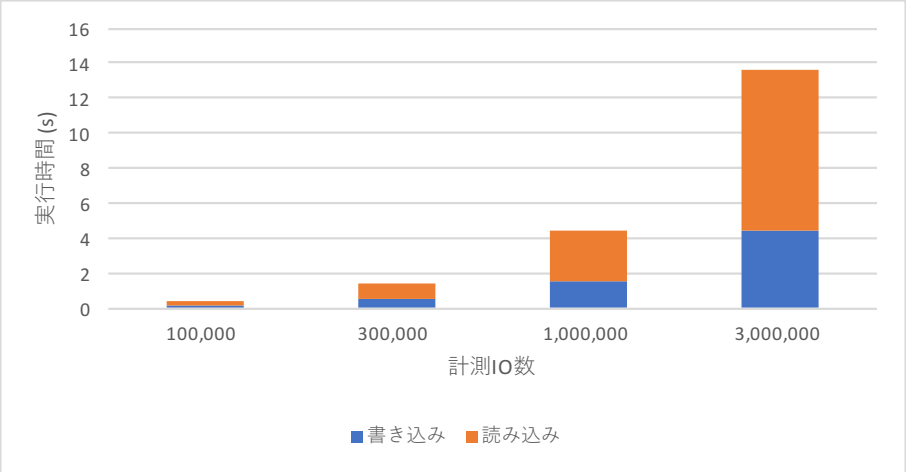

図 5.16: 書き込み畳み込みを用いた際の計測に要した実行時間 (バッファサイズ 10, 先行書き込み 100 万回)

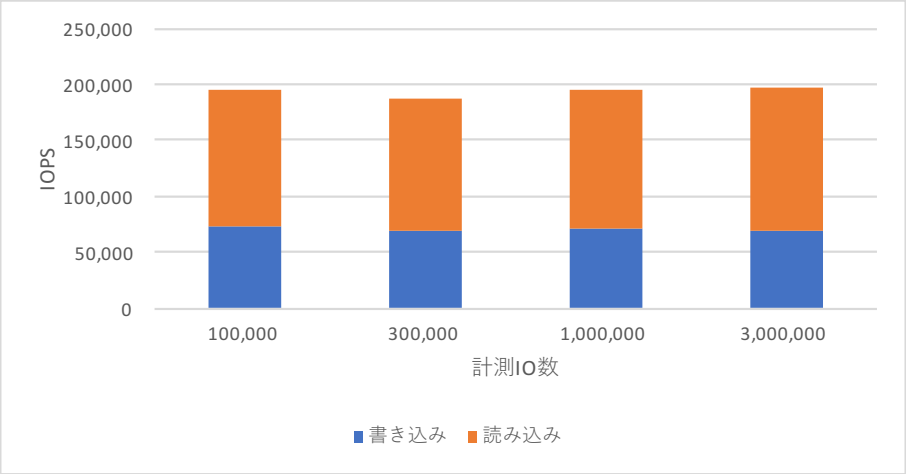

図 5.17: 書き込み畳み込みを用いた際の計測時の IOPS (バッファサイズ 10, 先行書き込み 100 万回)

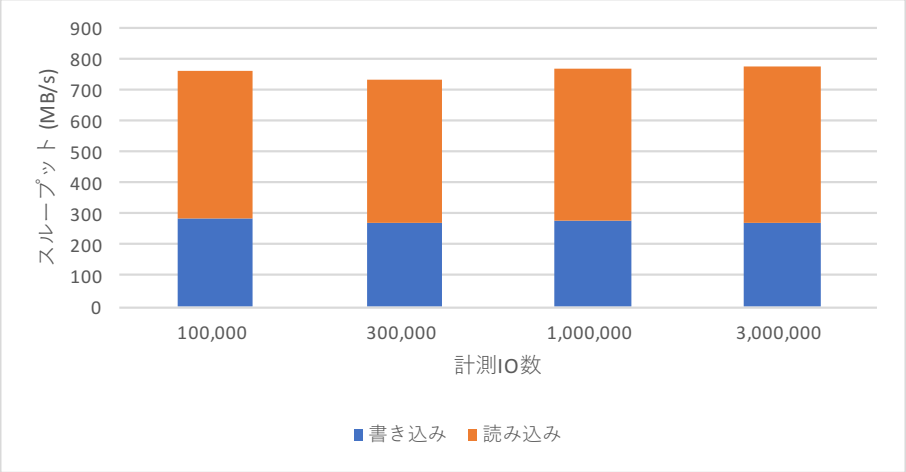

図 5.18: 書き込み畳み込みを用いた際の計測時のスループット (バッファサイズ 10, 先行書き込み 100 万回)

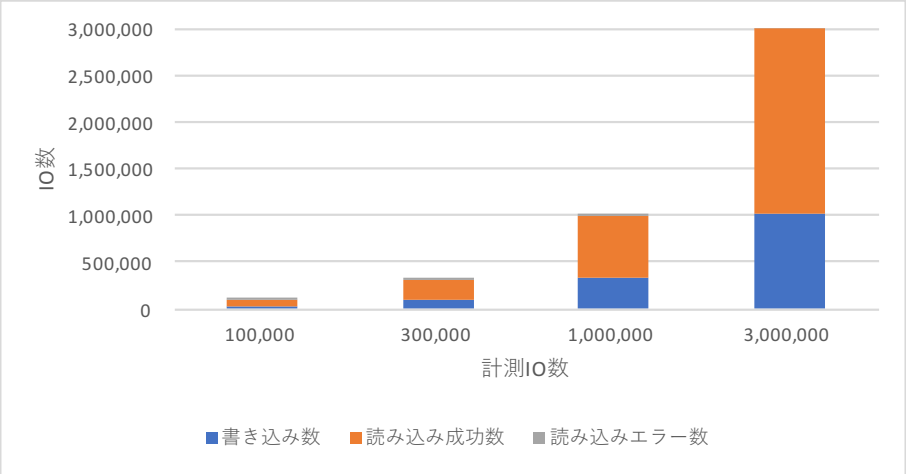

図 5.19: 書き込み畳み込みを用いた際の各種 IO の内訳 (バッファサイズ 10, 先行書き込み 1000 万回)

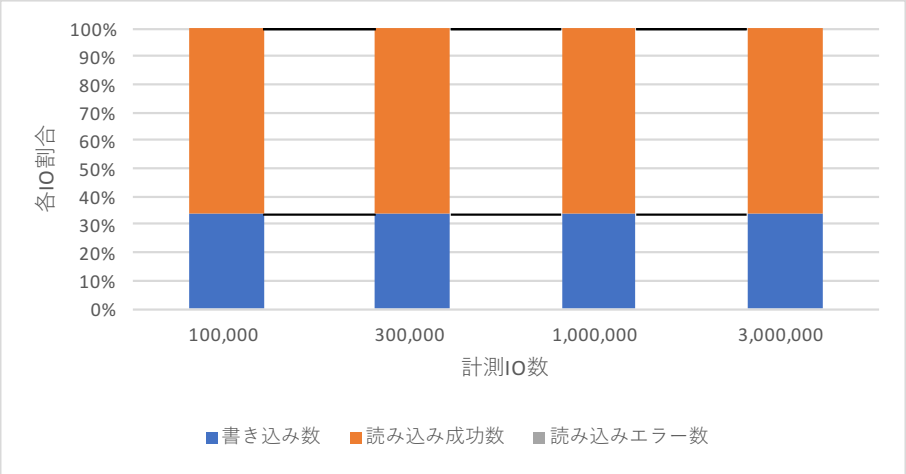

図 5.20: 計測 IO 数に対する各種 IO の割合 (バッファサイズ 10, 先行書き込み 1000 万回)

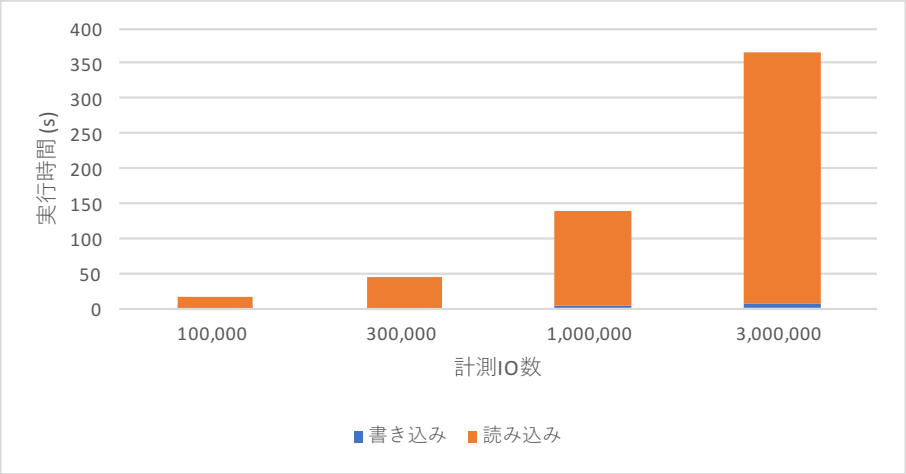

図 5.21: 各 IO 数の計測に要した実行時間 (バッファサイズ 10, 先行書き込み 1000 万回)

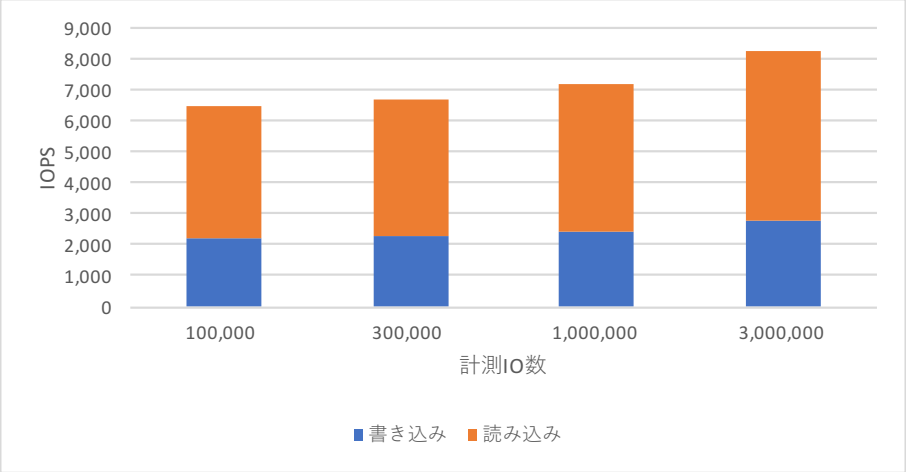

図 5.22: 各 IO 数における計測時の IOPS (バッファサイズ 10, 先行書き込み 1000 万回)

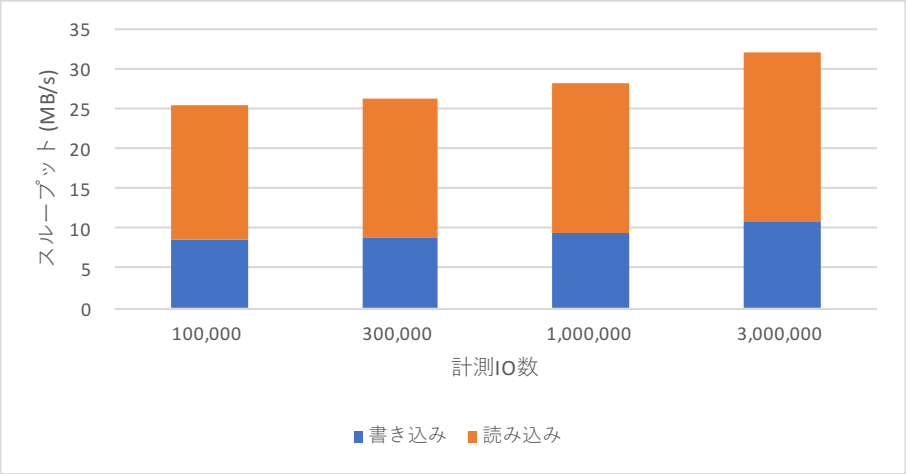

図 5.23: 各 IO 数における計測時のスループット (バッファサイズ 10, 先行書き込み 1000 万回)

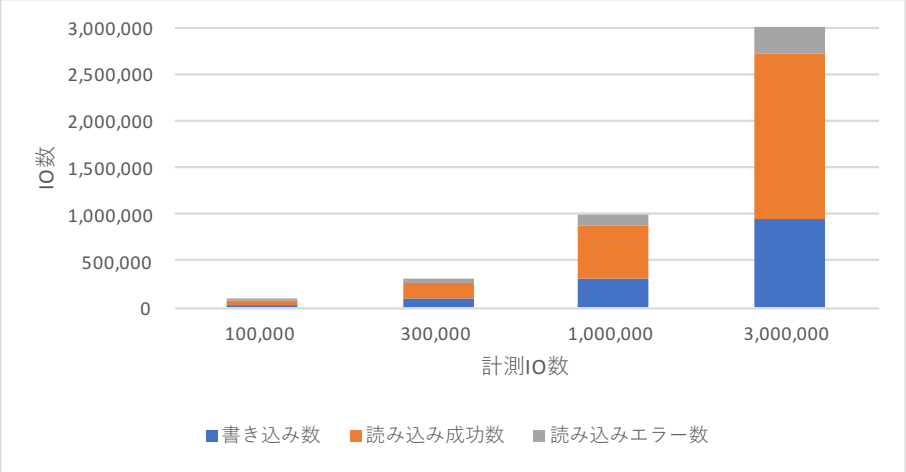

図 5.24: 書き込み畳み込みを用いた際の各種 IO の内訳 (バッファサイズ 100, 先行書き込み 100 万回)

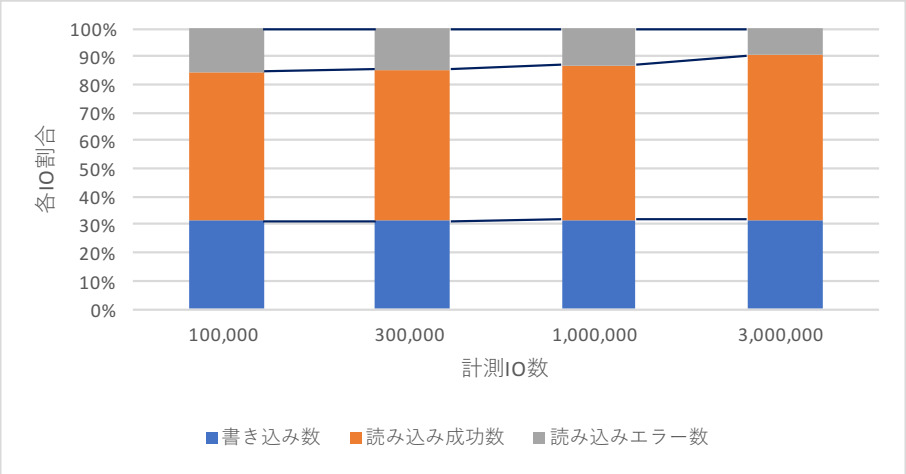

図 5.25: 書き込み畳み込みを用いた際の各種 IO の割合 (バッファサイズ 100, 先行書き込み 100 万回)

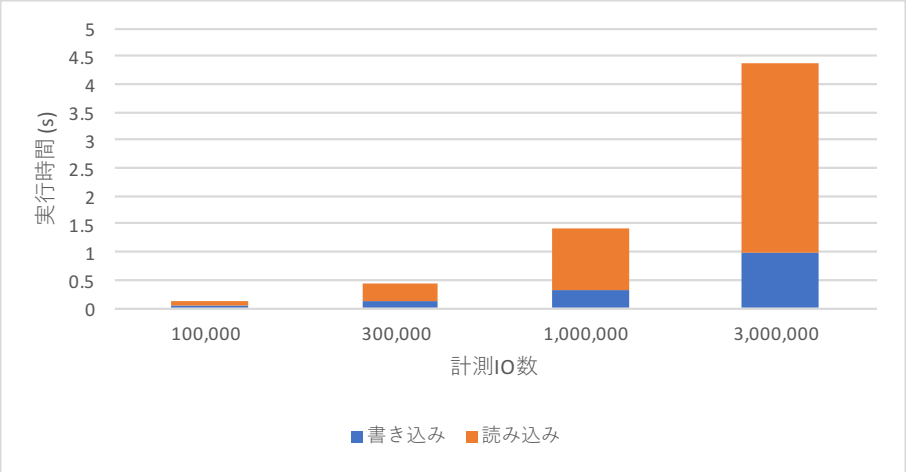

図 5.26: 書き込み畳み込みを用いた際の計測に要した実行時間 (バッファサイズ 100, 先行書き込み 100 万回)

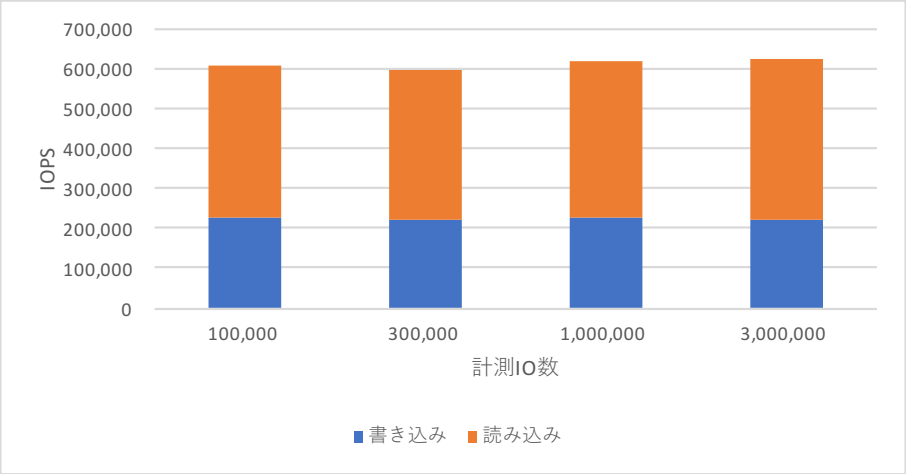

図 5.27: 書き込み畳み込みを用いた際の計測時の IOPS (バッファサイズ 100, 先行書き込み 100 万回)

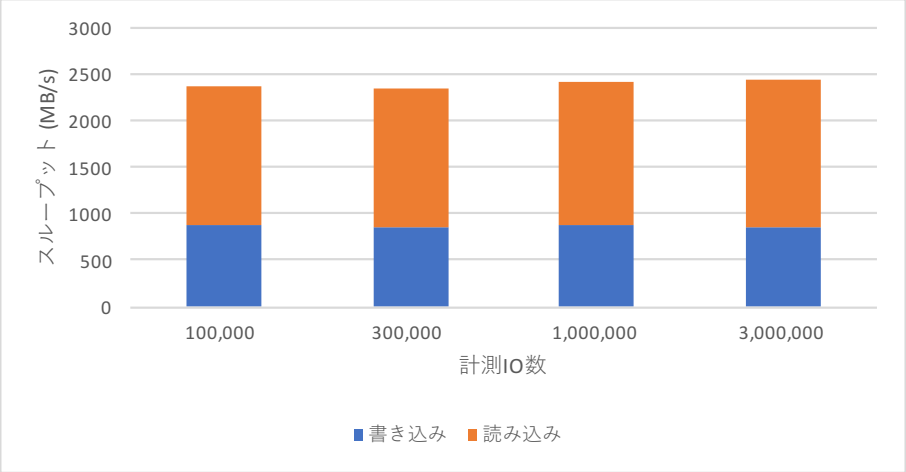

図 5.28: 書き込み畳み込みを用いた際の計測時のスループット (バッファサイズ 100, 先行書き込み 100 万回)

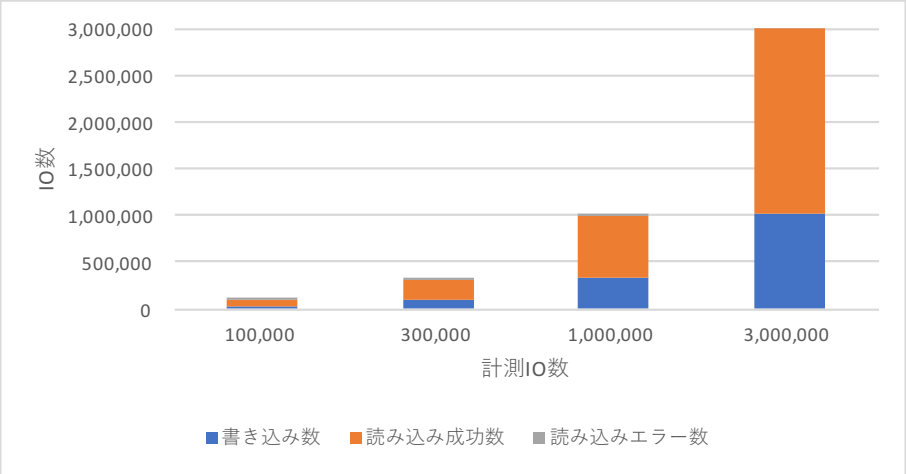

図 5.29: 書き込み畳み込みを用いた際の各種 IO の内訳 (バッファサイズ 100, 先行書き込み 1000 万回)

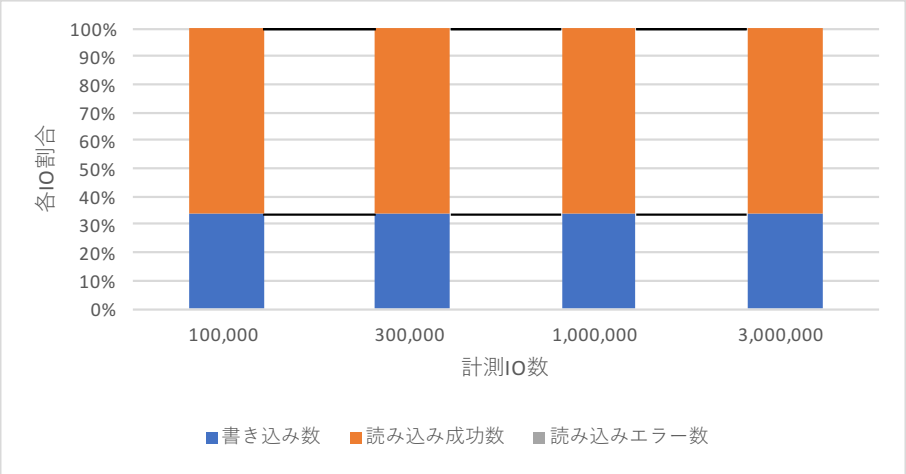

図 5.30: 計測 IO 数に対する各種 IO の割合 (バッファサイズ 100, 先行書き込み 1000 万回)

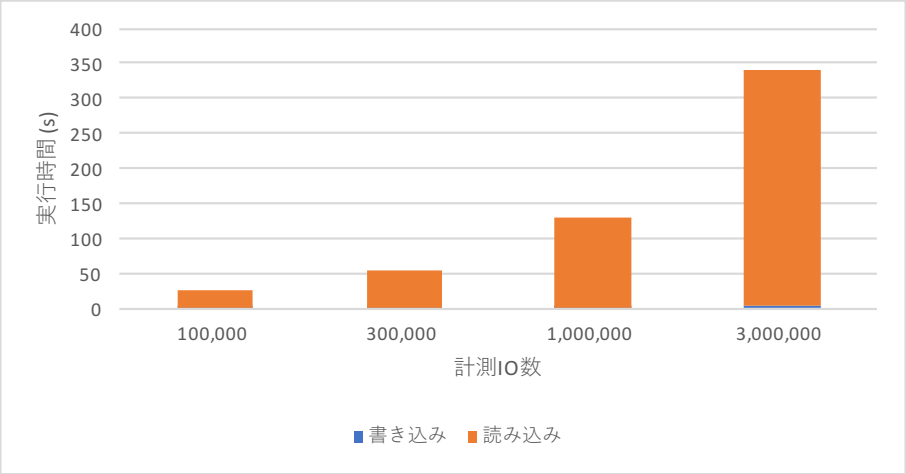

図 5.31: 各 IO 数の計測に要した実行時間 (バッファサイズ 100, 先行書き込み 1000 万回)

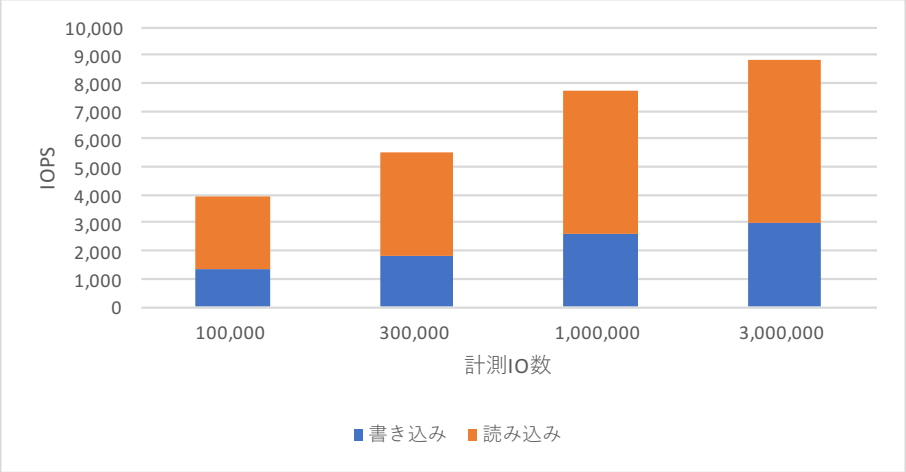

図 5.32: 各 IO 数における計測時の IOPS (バッファサイズ 100, 先行書き込み 1000 万回)

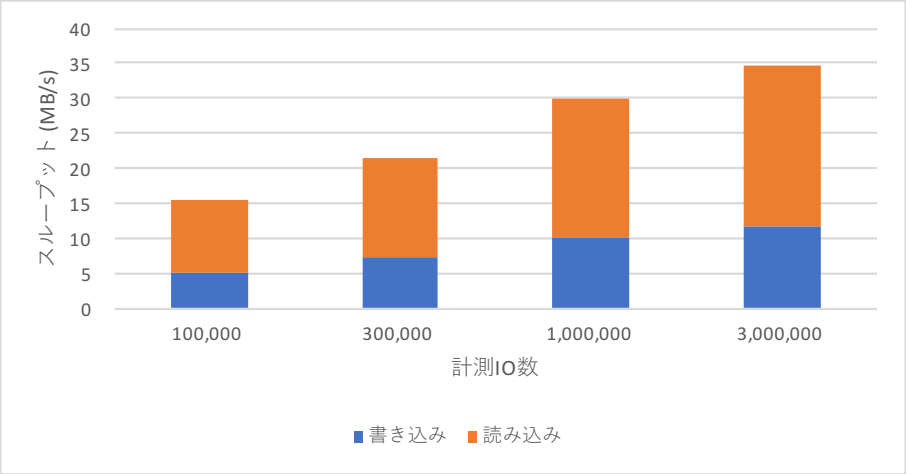

図 5.33: 各 IO 数における計測時のスループット (バッファサイズ 100, 先行書き込み 1000 万回)

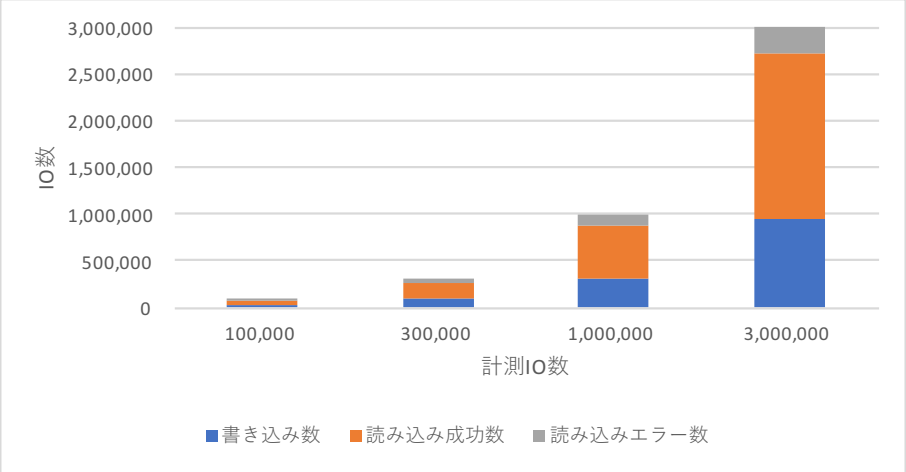

図 5.34: 書き込み畳み込みを用いた際の各種 IO の内訳 (バッファサイズ 1000, 先行書き込み 100 万回)

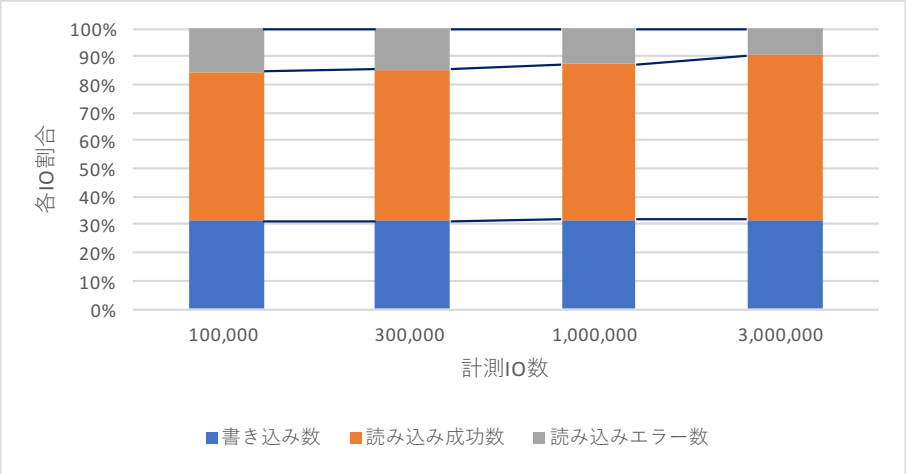

図 5.35: 書き込み畳み込みを用いた際の各種 IO の割合 (バッファサイズ 1000, 先行書き込み 100 万回)

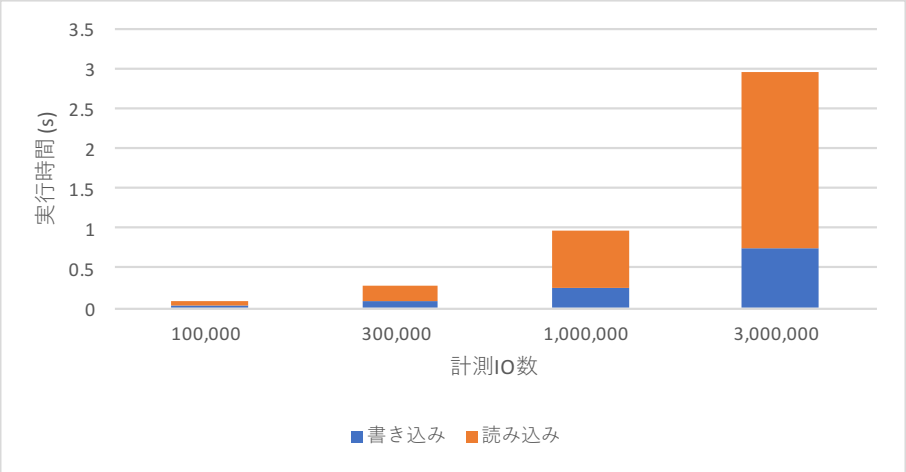

図 5.36: 書き込み畳み込みを用いた際の計測に要した実行時間 (バッファサイズ 1000, 先行書き込み 100 万回)

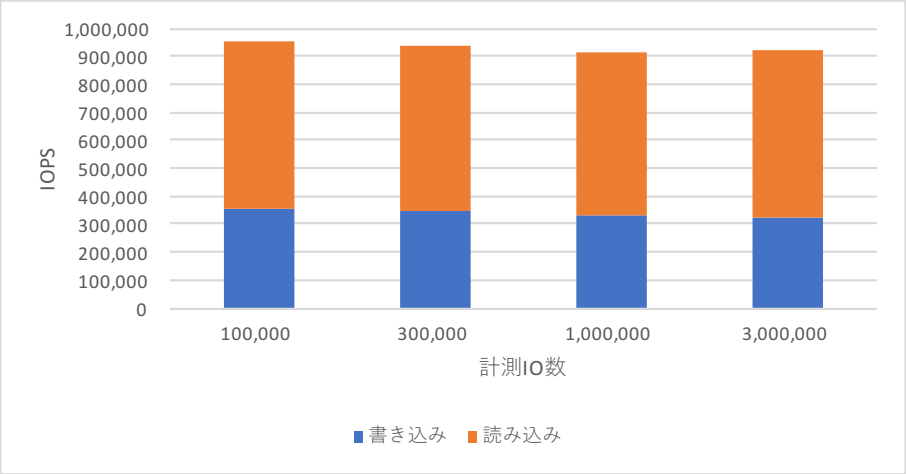

図 5.37: 書き込み畳み込みを用いた際の計測時の IOPS (バッファサイズ 1000, 先行書き込み 100 万回)

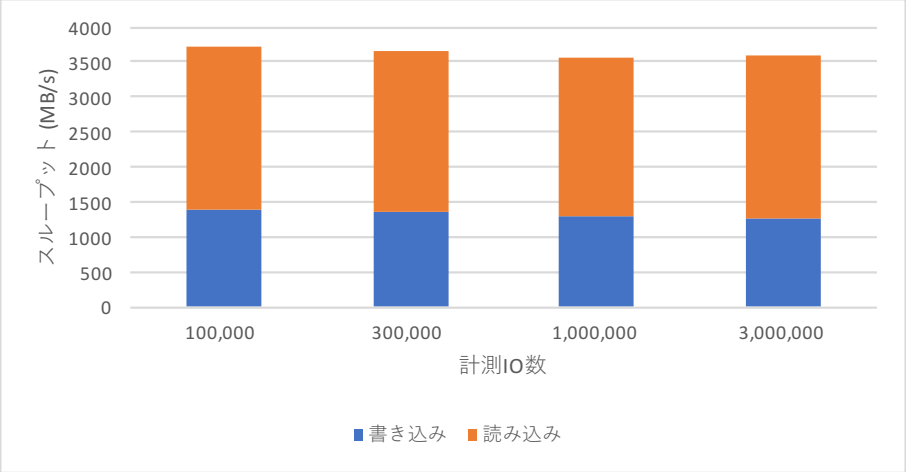

図 5.38: 書き込み畳み込みを用いた際の計測時のスループット (バッファサイズ 1000, 先行書き込み 100 万回)

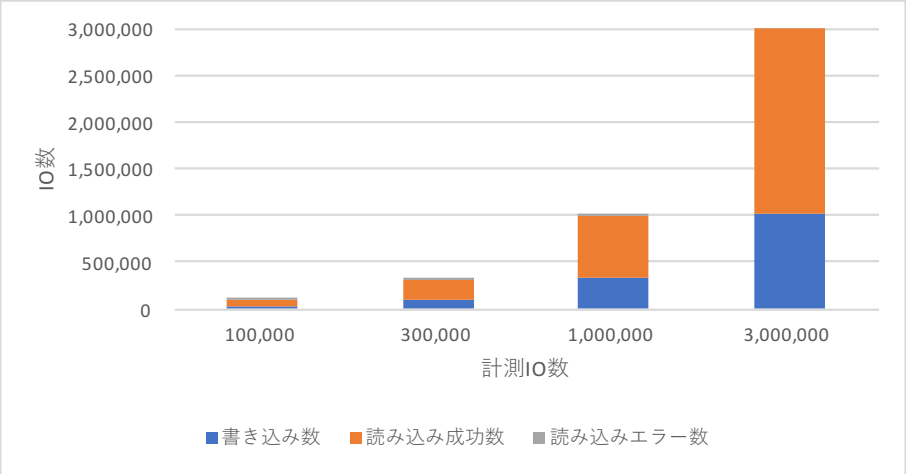

図 5.39: 書き込み畳み込みを用いた際の各種 IO の内訳 (バッファサイズ 1000, 先行書き込み 1000 万回)

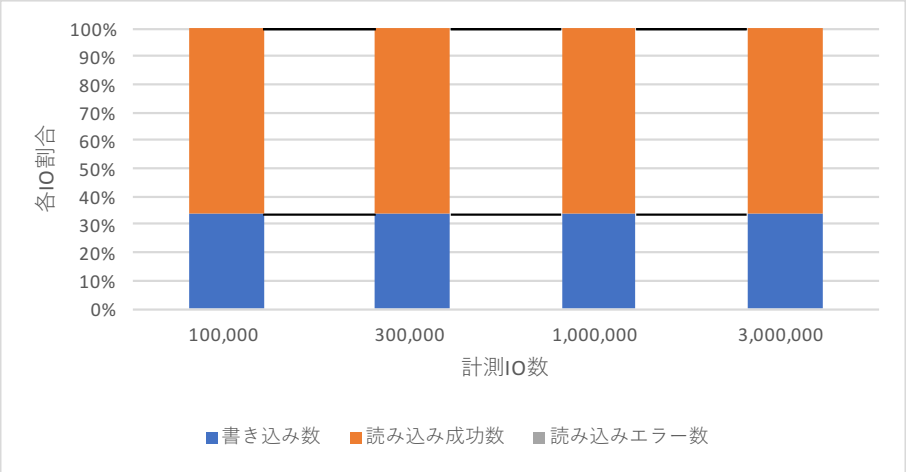

図 5.40: 計測 IO 数に対する各種 IO の割合 (バッファサイズ 1000, 先行書き込み 1000 万回)

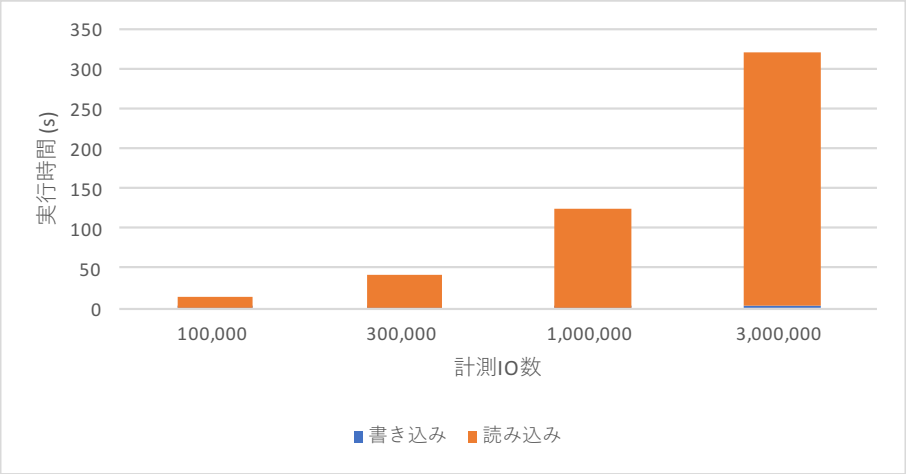

図 5.41: 各 IO 数の計測に要した実行時間 (バッファサイズ 1000, 先行書き込み 1000 万回)

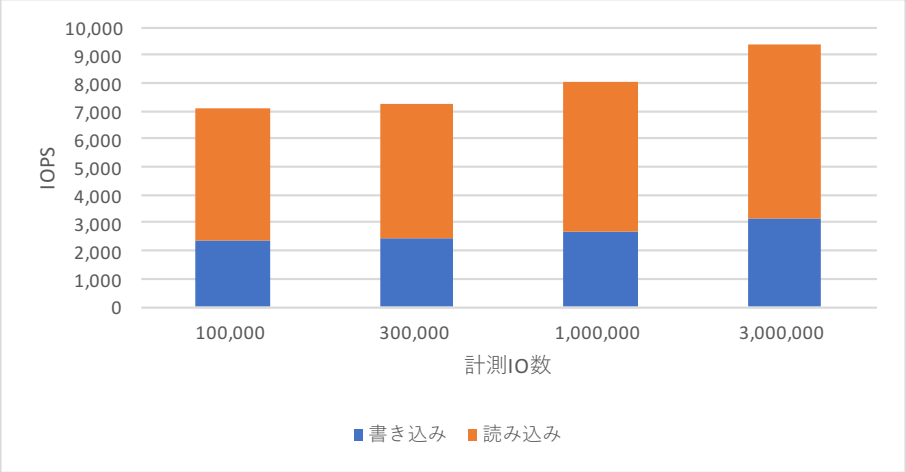

図 5.42: 各 IO 数における計測時の IOPS (バッファサイズ 1000, 先行書き込み 1000 万回)

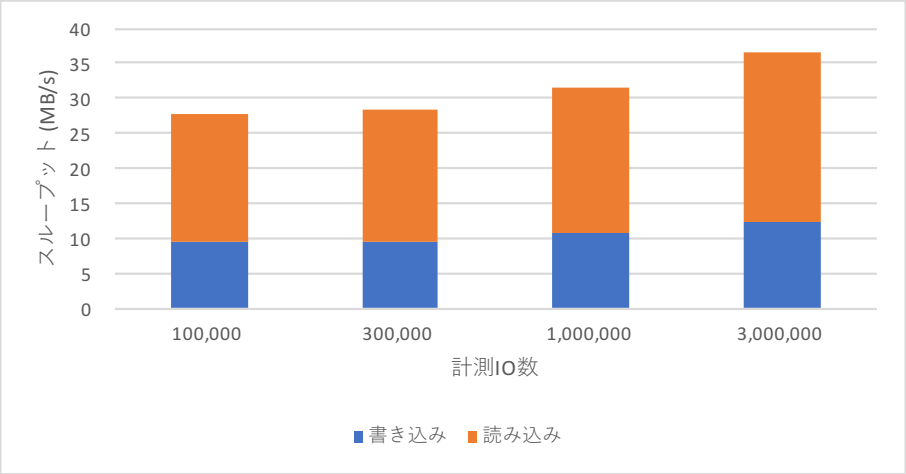

図 5.43: 各 IO 数における計測時のスループット (バッファサイズ 1000, 先行書き込み 1000 万回)

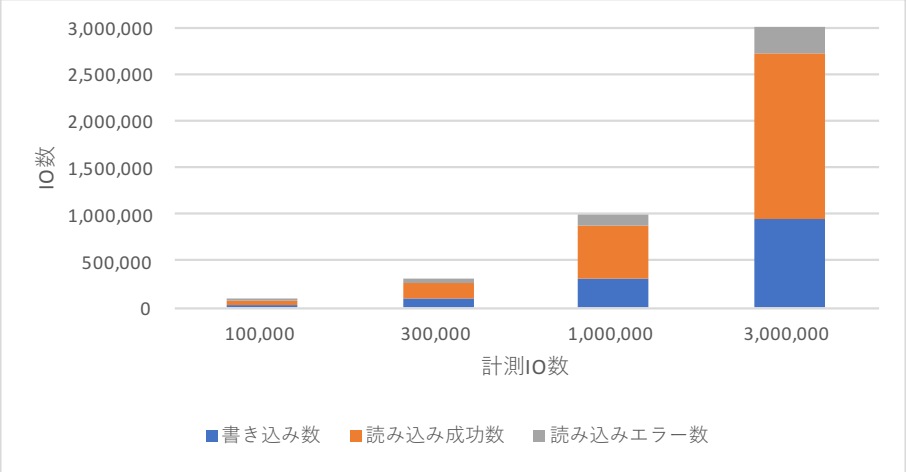

図 5.44: 書き込み畳み込みを用いた際の各種 IO の内訳 (バッファサイズ 10000, 先行書き込み 100 万回)

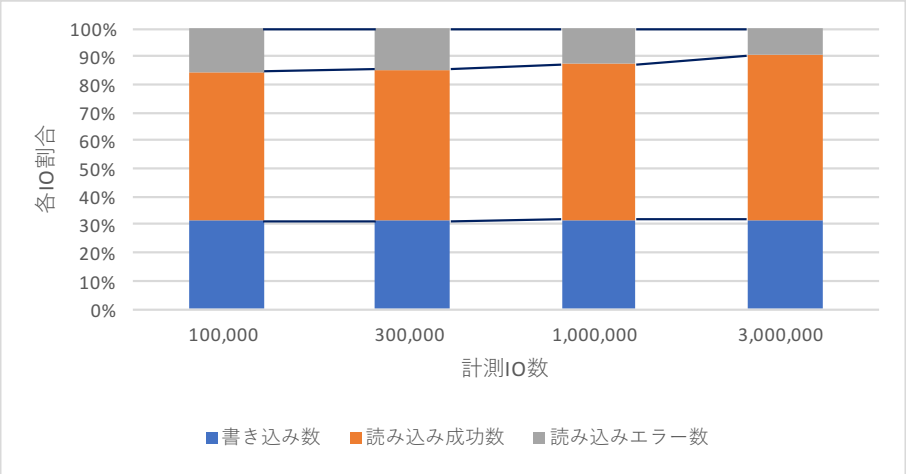

図 5.45: 書き込み畳み込みを用いた際の各種 IO の割合 (バッファサイズ 10000, 先行書き込み 100 万回)

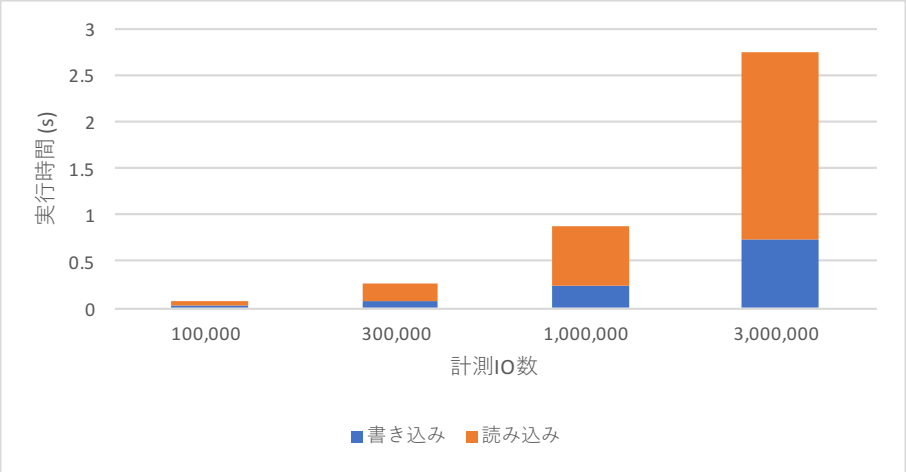

図 5.46: 書き込み畳み込みを用いた際の計測に要した実行時間 (バッファサイズ 10000, 先行書き込み 100 万回)

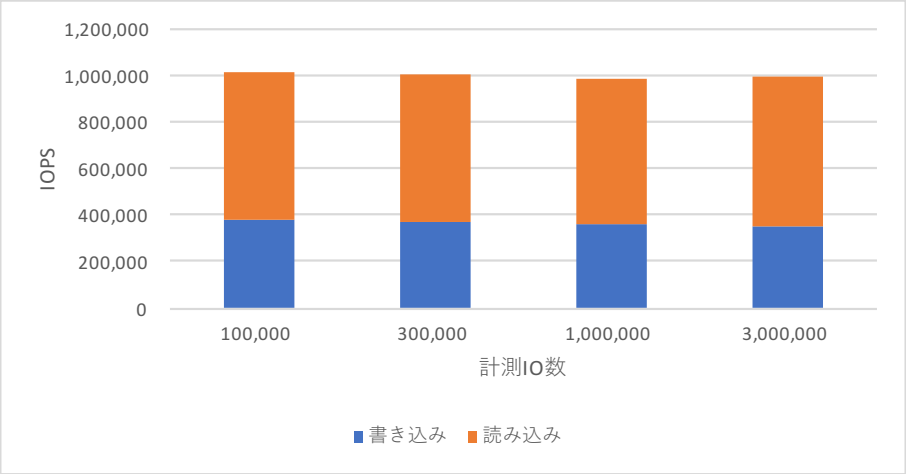

図 5.47: 書き込み畳み込みを用いた際の計測時の IOPS (バッファサイズ 10000, 先行書き込み 100 万回)

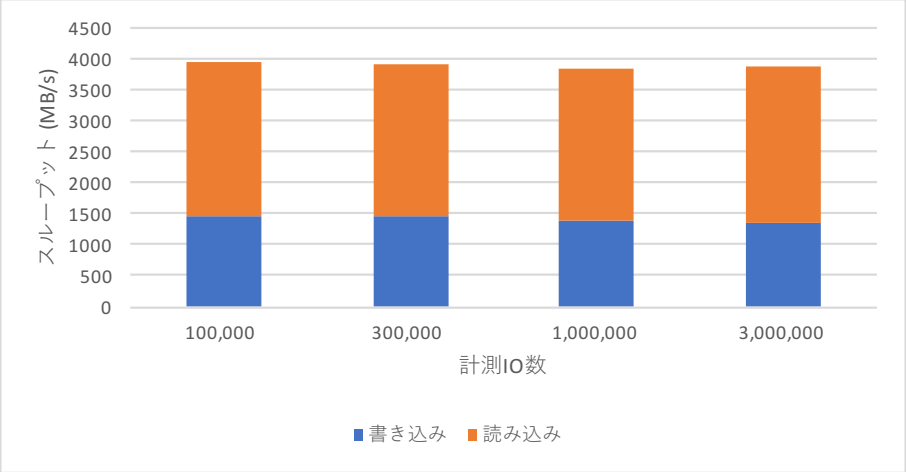

図 5.48: 書き込み畳み込みを用いた際の計測時のスループット (バッファサイズ 10000, 先行書き込み 100 万回)

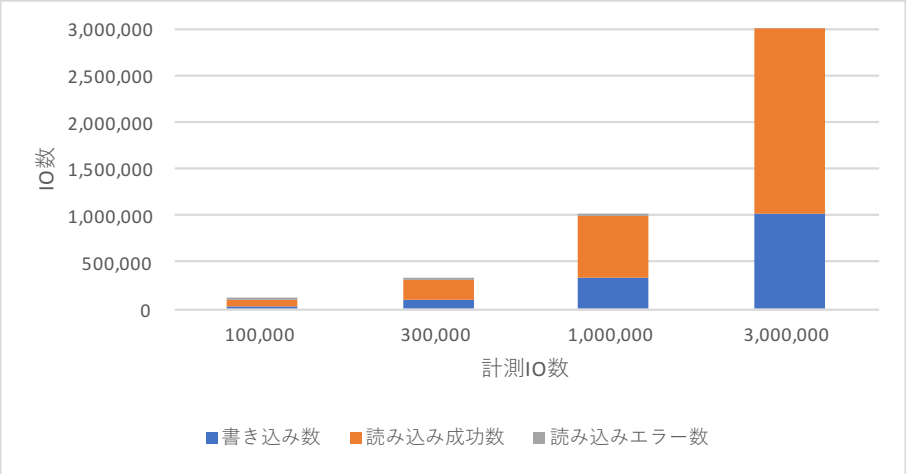

図 5.49: 書き込み畳み込みを用いた際の各種 IO の内訳 (バッファサイズ 10000, 先行書き込み 1000 万回)

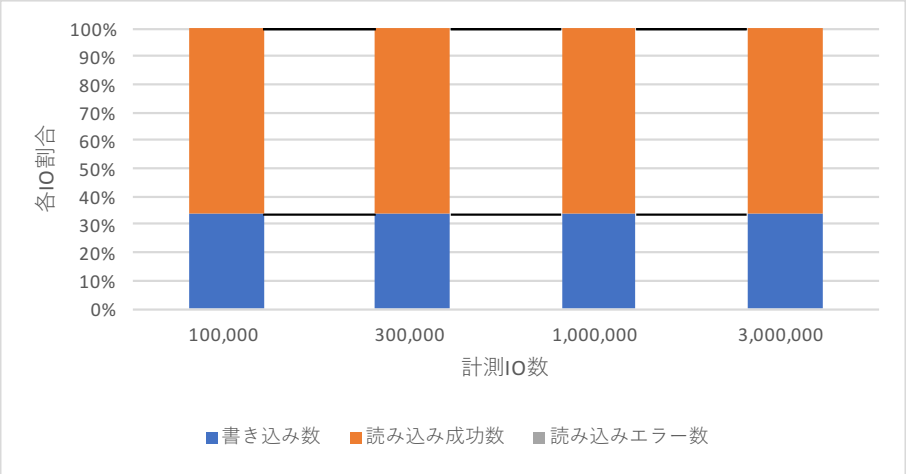

図 5.50: 計測 IO 数に対する各種 IO の割合 (バッファサイズ 10000, 先行書き込み 1000 万回)

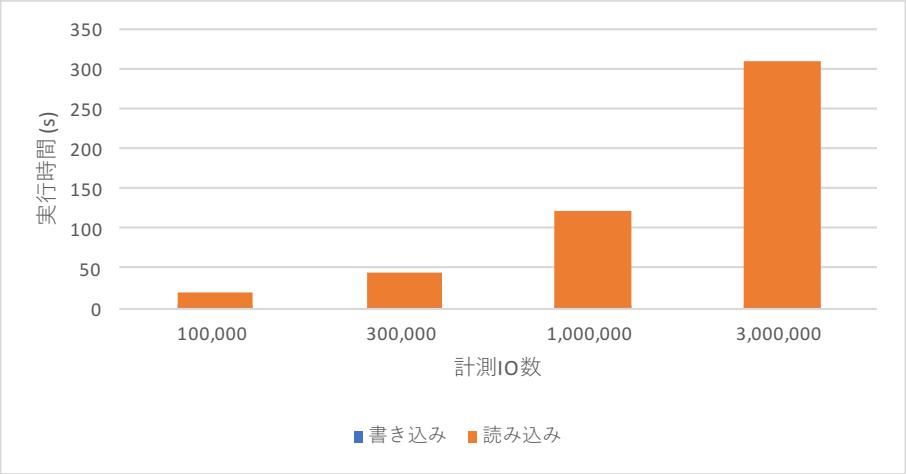

図 5.51: 各 IO 数の計測に要した実行時間 (バッファサイズ 10000, 先行書き込み 1000 万回)

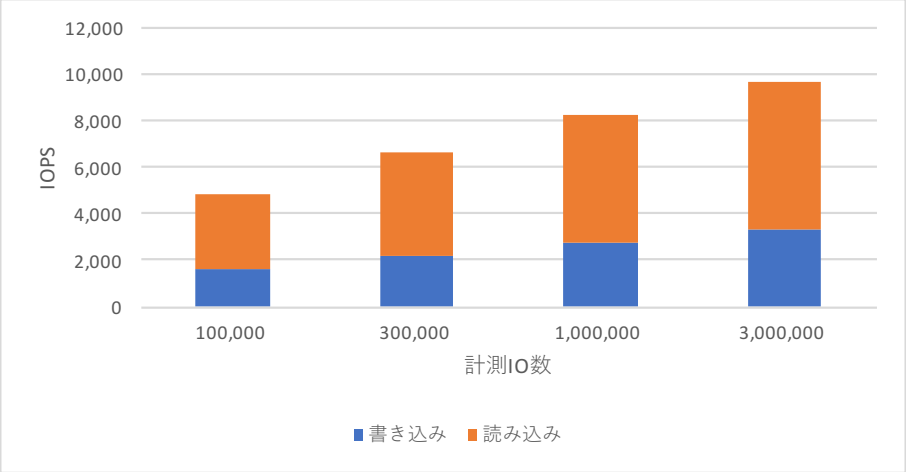

図 5.52: 各 IO 数における計測時の IOPS (バッファサイズ 10000, 先行書き込み 1000 万回)

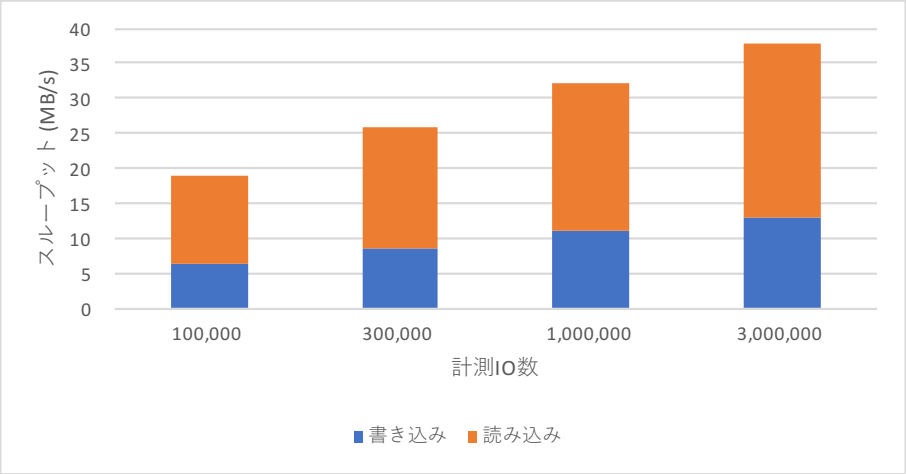

図 5.53: 各 IO 数における計測時のスループット (バッファサイズ 10000, 先行書き込み 1000 万回)

5.5 ページキャッシュを経由しない IO の書き込み畳み込み効果

5.3 節と同様の実験設定で今度はページキャッシュを経由させないため O_DIRECT を用いて比較を行った. 5.2 節と 5.3 節の結果より計測 IO 数による差異は確認できたので, 本実験では先行書き込み数を 100 万回と 1000 万回の 2 通り, 計測 IO 数を 10 万回と 100 万回の 2 通りで実行時間, IOPS, スループットについて計測を行い, 書き込み畳み込みの効果を確認した. 実験結果を図 5.54 から図 5.69 に示す. 横軸はバッファサイズであり, 縦軸は図 5.54, 図 5.58, 図 5.62, 図 5.66 は書き込みの実行時間を, 図 5.55, 図 5.59, 図 5.63, 図 5.67 は読み込みの実行時間を, 図 5.56, 図 5.60, 図 5.64, 図 5.68 は IOPS を, 図 5.57, 図 5.61, 図 5.65, 図 5.69 はスループットをそれぞれ表している. これらの結果によると, 書き込みの畳み込みによって IOPS, スループットが上昇し性能が向上していることがわかる. 読み込み実行時間についてもバッファサイズの上昇に伴い減少傾向が見られるが, 書き込み実行時間はバッファサイズ 10,000

のときのみ増加している．これは IO パスによる影響だと推察される．またバッファサイズ 0 について先行書き込み数によって IOPS が減少しているが，この減少の原因は読み込み成功率の違いによって全 IO 数に占める読み込み数と書き込み数の相対的な割合が変化し，読み込み時間は書き込み時間に比べて長いため読み込み成功率が上昇すると IOPS が減少するのだと思われる．しかし，この仮説のもと読み込み実行時間と書き込み実行時間の比は大まかに 1：20 で近似し，読み込みエラー率が 20%のときの IOPS を書き込みと読み込み合計で 250 とすると，計算上は読み込みエラー率が 0%の時の IOPS は 210 となるがこれは実験結果と一致しない．よってこれ以外にも原因があると思われる．

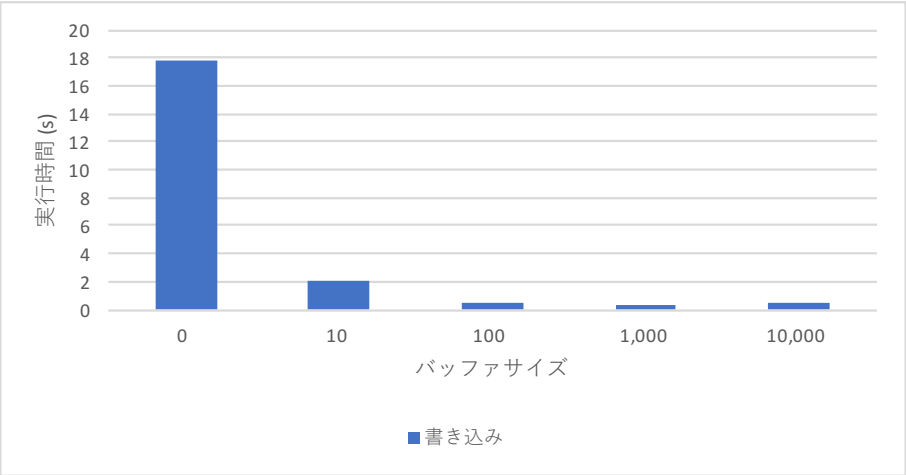

図 5.54: 各バッファサイズにおける計測時の書き込み実行時間 (先行書き込み 100 万回，計測 IO 数 10 万回)

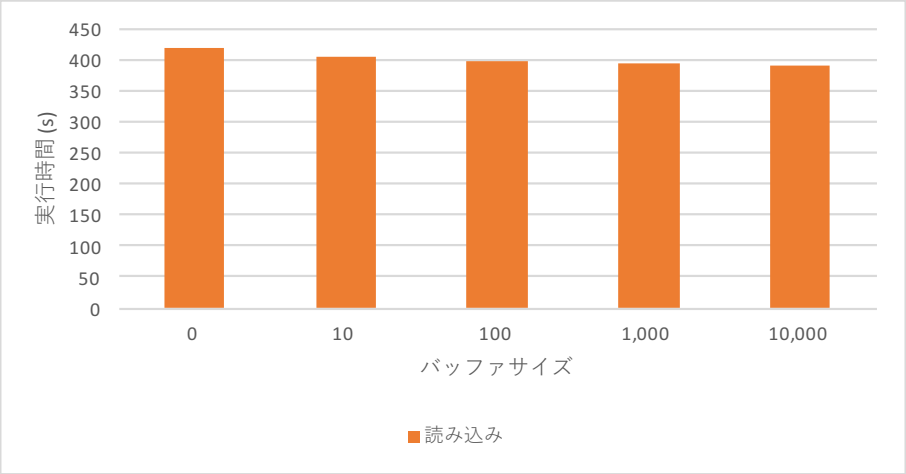

図 5.55: 各バッファサイズにおける計測時の読み込み実行時間 (先行書き込み 100 万回, 計測 IO 数 10 万回)

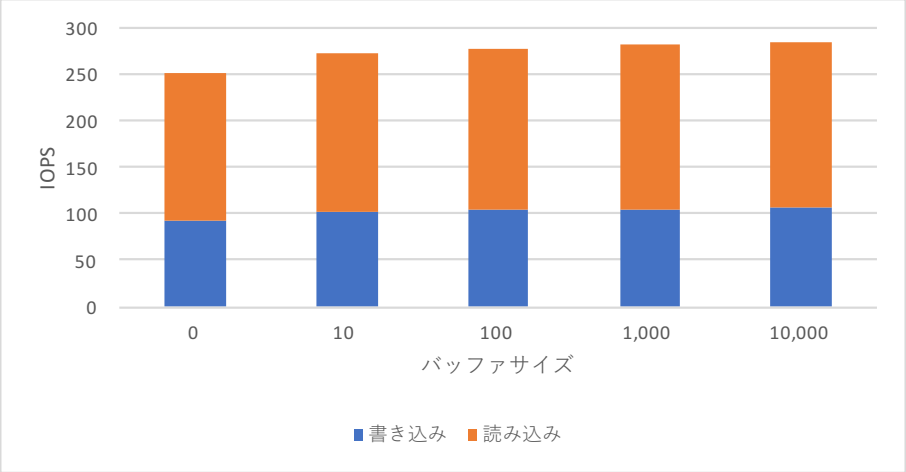

図 5.56: 各バッファサイズにおける計測時の IOPS (先行書き込み 100 万回, 計測 IO 数 10 万回)

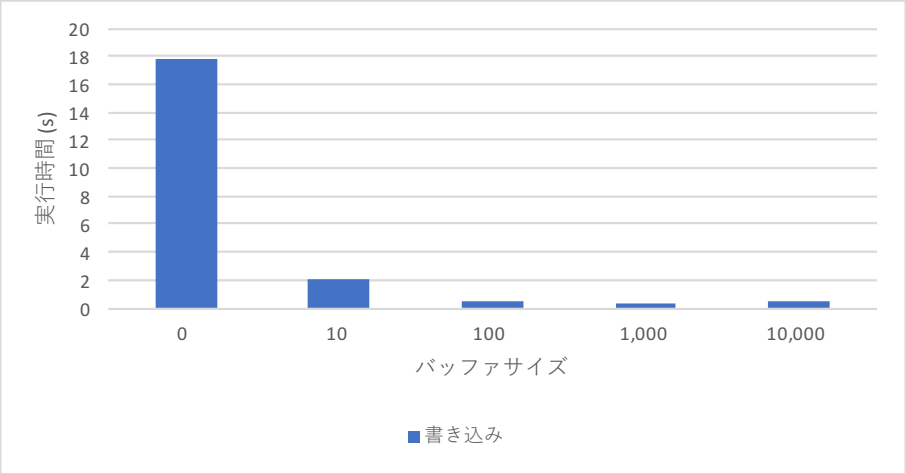

図 5.57: 各バッファサイズにおける計測時のスループット (先行書き込み 100 万回, 計測 IO 数 10 万回)

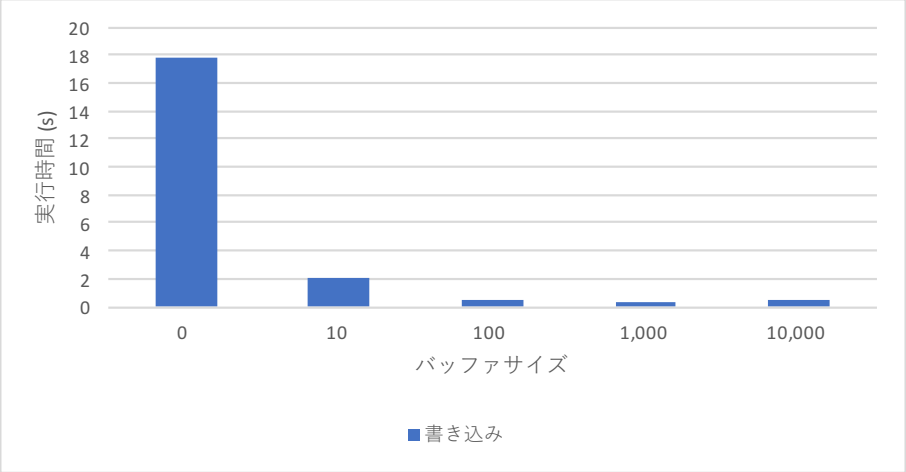

図 5.58: 各バッファサイズにおける計測時の書き込み実行時間 (先行書き込み 1000 万回, 計測 IO 数 10 万回)

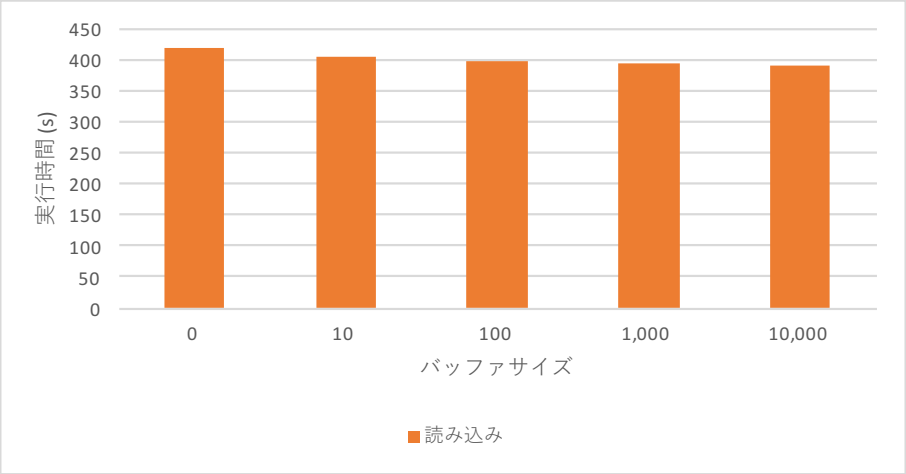

図 5.59: 各バッファサイズにおける計測時の読み込み実行時間 (先行書き込み 1000 万回, 計測 IO 数 10 万回)

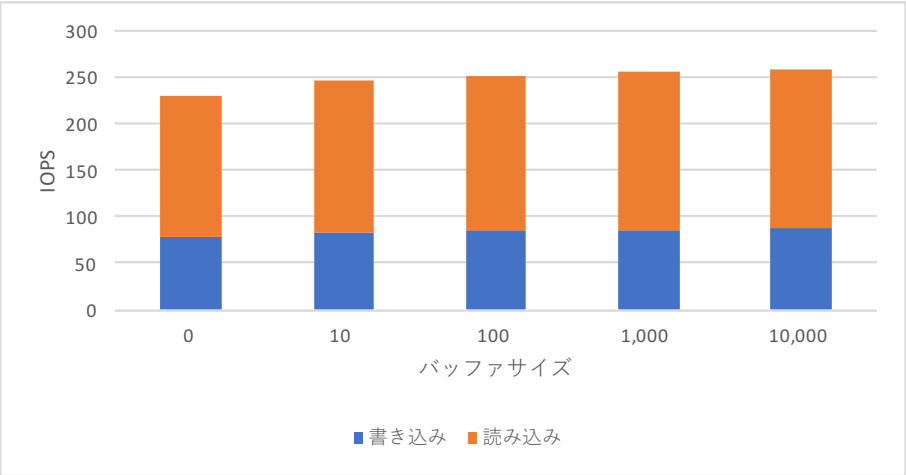

図 5.60: 各バッファサイズにおける計測時の IOPS (先行書き込み 1000 万回, 計測 IO 数 10 万回)

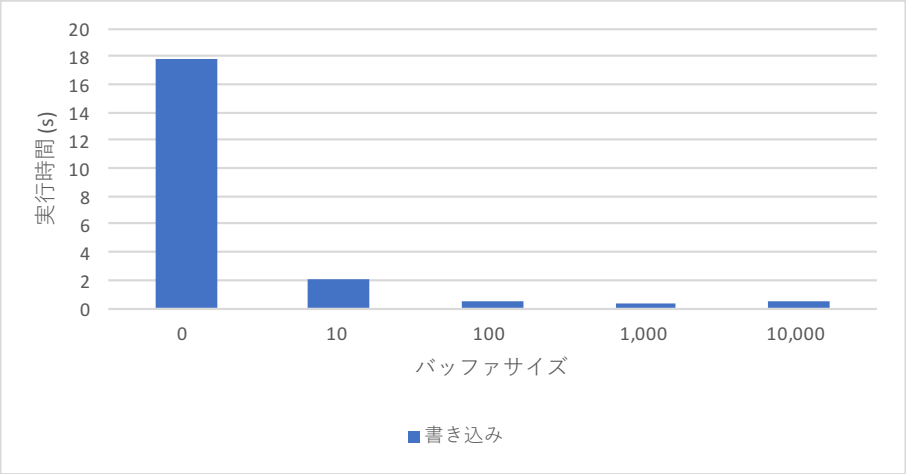

図 5.61: 各バッファサイズにおける計測時のスループット (先行書き込み 1000 万回, 計測 IO 数 10 万回)

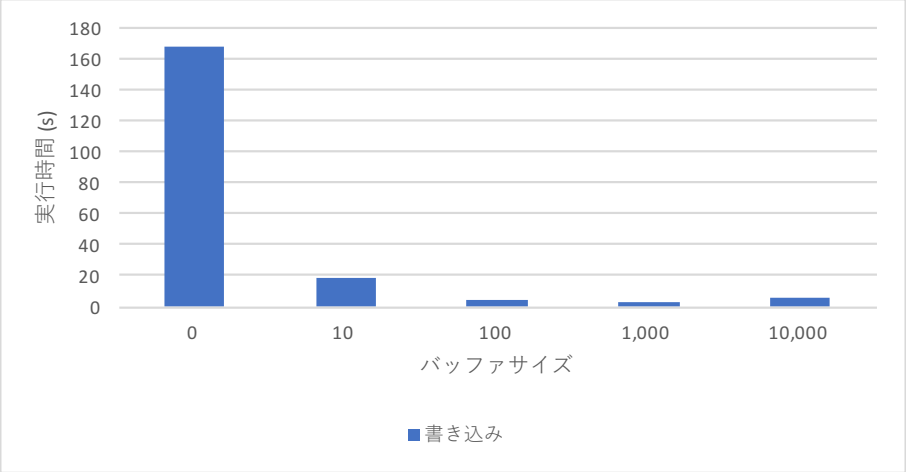

図 5.62: 各バッファサイズにおける計測時の書き込み実行時間 (先行書き込み 100 万回, 計測 IO 数 100 万回)

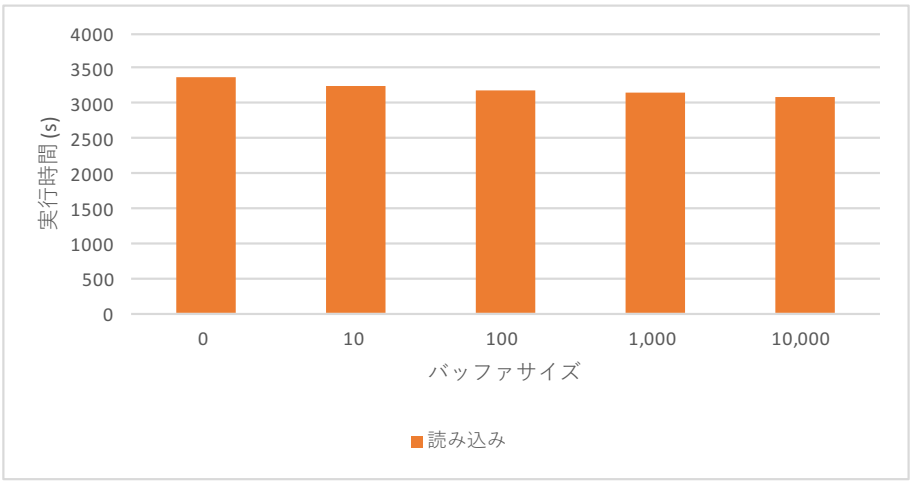

図 5.63: 各バッファサイズにおける計測時の読み込み実行時間 (先行書き込み 100 万回, 計測 IO 数 100 万回)

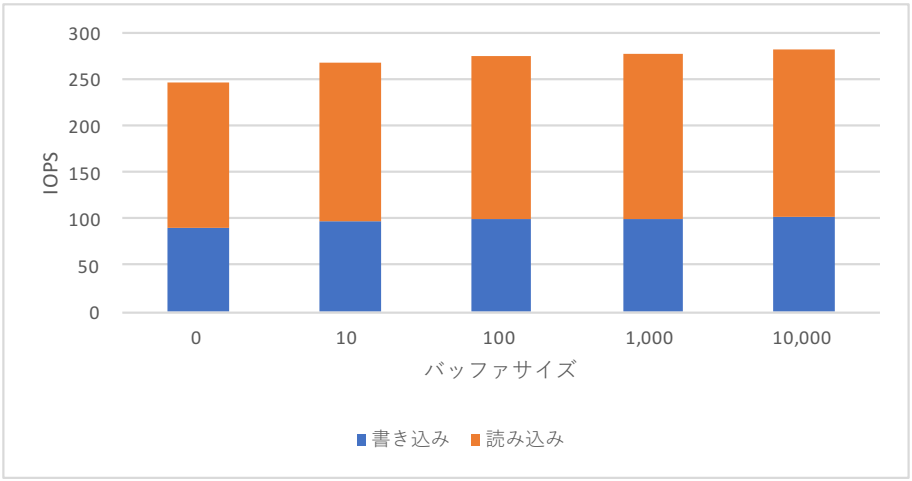

図 5.64: 各バッファサイズにおける計測時の IOPS (先行書き込み 100 万回, 計測 IO 数 100 万回)

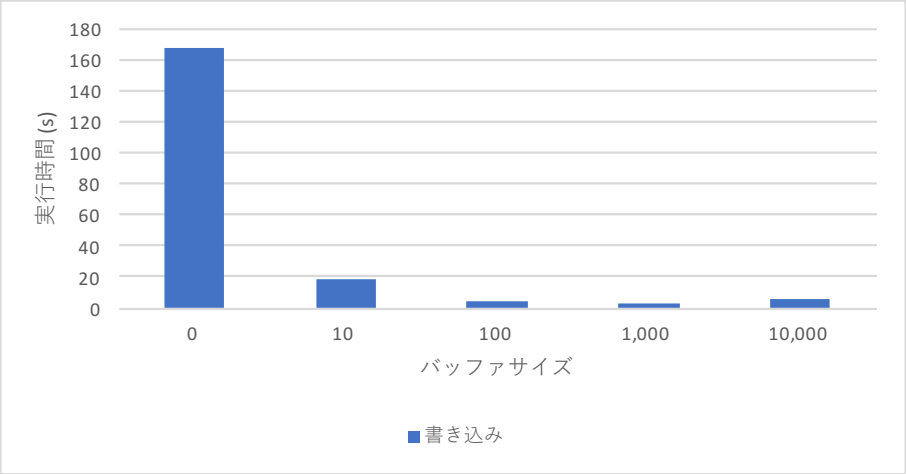

図 5.65: 各バッファサイズにおける計測時のスループット (先行書き込み 100 万回, 計測 IO 数 100 万回)

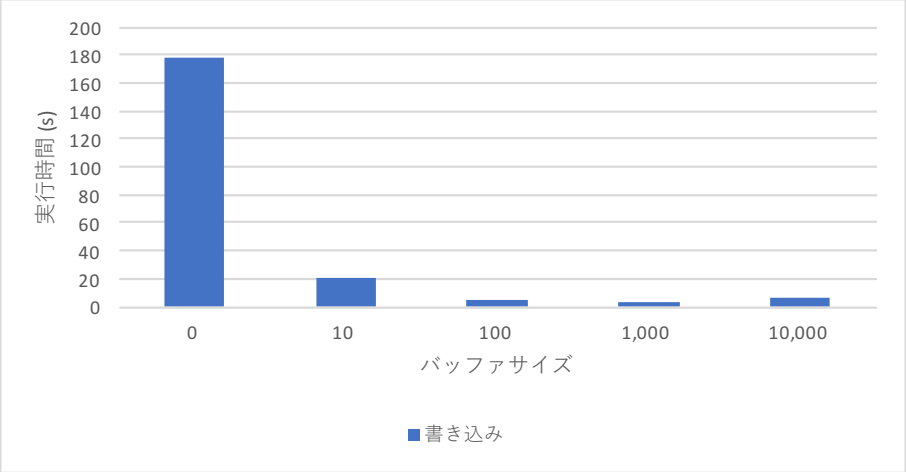

図 5.66: 各バッファサイズにおける計測時の書き込み実行時間 (先行書き込み 1000 万回, 計測 IO 数 100 万回)

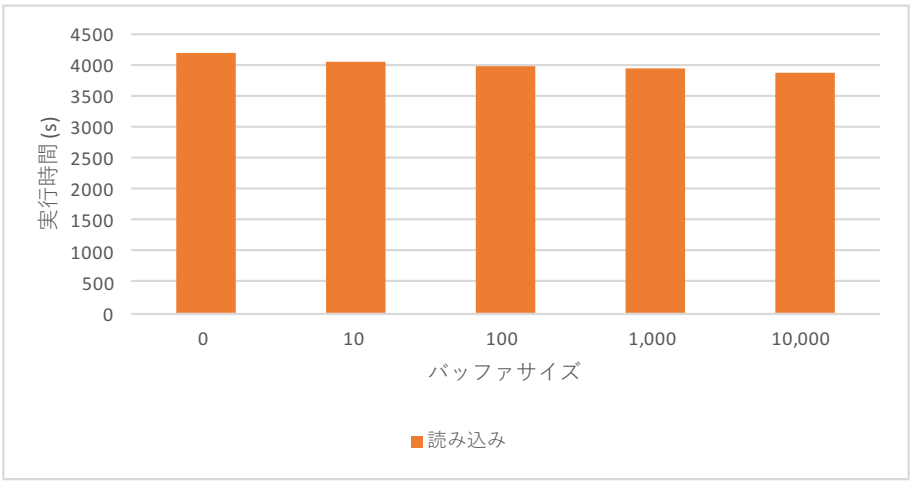

図 5.67: 各バッファサイズにおける計測時の読み込み実行時間 (先行書き込み 1000 万回, 計測 IO 数 100 万回)

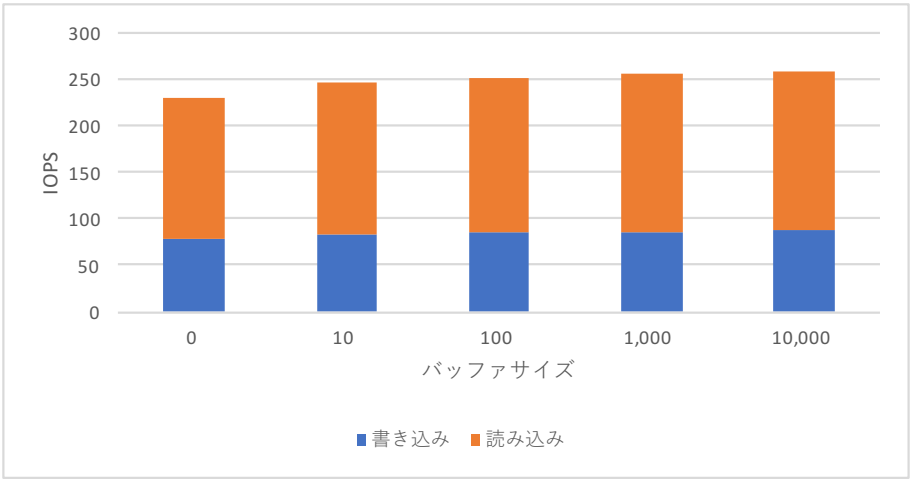

図 5.68: 各バッファサイズにおける計測時の IOPS (先行書き込み 1000 万回, 計測 IO 数 100 万回)

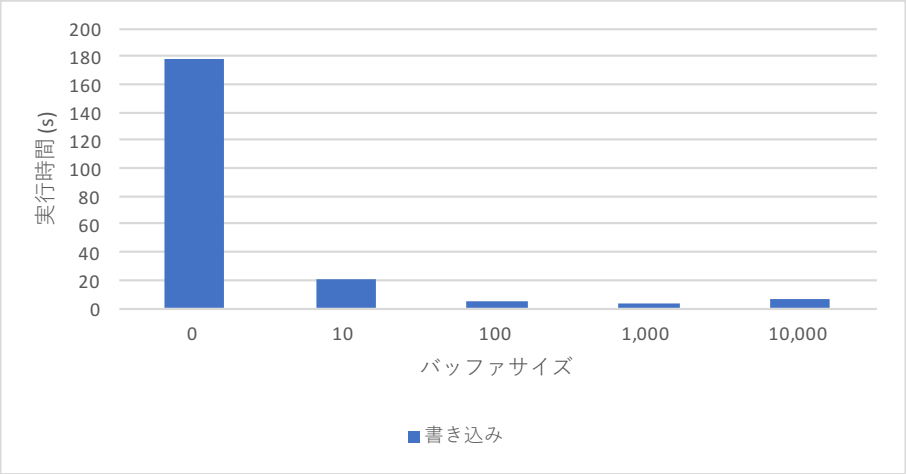

図 5.69: 各バッファサイズにおける計測時のスループット (先行書き込み 1000 万回, 計測 IO 数 100 万回)

第6章　まとめと今後の展望

6.1　まとめ

本稿では, まずマイクロベンチマークを用いて Drive Managed 方式 SMR ディスクにおける書き込み負荷に対する読み込み性能の観測を行った. その結果, 書き込み負荷によらずランダム読み込みでは性能が悪化すること, および書き込み後は書き込み量の増大に伴い長期化する読み込みレイテンシのバースト的な増大が生じること明らかになった. 次に, データベース負荷を模した IO トレースを用いた Host Managed 型 SMR ディスクの入出力性能特性の検証を行った. その結果, 書き込み畳み込みによって IOPS やスループットが向上することを確認した。

6.2　今後の課題

今後の課題としては, 本研究で明らかになった Drive Managed 方式 SMR ディスクの性能特性がデータベース処理性能に与える影響を観察すること, Host Managed 方式 SMR ディスクの実機においてデータベース性能特性を検証することなどが挙げられる.

謝辞

本研究を進めるにあたって, 私が師事する合田先生と早水さんには研究のみならず私生活面も含めひとかたならぬお世話になり深甚なる感謝を表させていただきます.

また, 私淑する喜連川先生, ミーティングで助言をくださった豊田先生, 横山先生, 吉永先生はじめ研究室の皆々様にもこの場を借りて深い感謝の意を表させていただきます.

2017 年 2 月 1 日

参考文献

[1] K. Goda and M. Kitsuregawa. The history of storage systems. In *Proceedings of the IEEE, 100.Centennial-Issue*, pp. 1433–1440, 2012.

[2] R. Wood, M. Williams, A. Kavcic, and J. Miles. The feasibility of magnetic recording at 10 terabits per square inch on conventional media. In *IEEE Trans. Magn., vol. 45, no. 2*, pp. 917–923, Feb. 2009.

[3] A. Amer, D. D. E. Long, E. L. Miller, J.-F. Paris, and T. Schwarz. Design issues for a shingled write disk system. In *26th IEEE Symposium on Mass Storage Systems and Technology*, pp. 1–12, 2010.

[4] M. Dunn and T. Feldman. Shingled magnetic recording models, standardization, and applications. In *SNIA Storage Developer Conference Tutorial*, 2014.

[5] INCITS T10 Technical Committee, et al. Information technology-zoned block commands (zbc). draft standard. 2014.

[6] INCITS T13 Technical Committee, et al. Information technology-zoned ata commands (zac). 2014.

[7] David Hall, John H. Marcos, and Jonathan D. Coker. Data handling algorithms for autonomous shingled magnetic recording hdds. In *IEEE Transactions on Magnetics 48.5*, pp. 1777–1781, 2012.

[8] Y. Cassuto, M. A. A. Sanvido, C. Guyot, D. R. Hall, and Z. Z. Bandic. Indirection systems for shingled-recording disk drives. In *2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST)*, pp. 1–14, 2010.

[9] Le Moal Damien, Zvonimir Bandic, and Cyril Guyot. Shingled file system host-side management of shingled magnetic recording disks. In *IEEE International Conference on Consumer Electronics (ICCE)*, 2012.

[10] Chao Jin et al. Hismrfs: A high performance file system for shingled storage array. In *30th Symposium on Mass Storage Systems and Technologies (MSST)*, 2014.

[11] Garth Gibson and Milo Polte. Directions for shingled-write and two-dimensional magnetic recording system architectures: Synergies with solid-state disks. 2009.

[12] Abutalib Aghayev, Mansour Shafaei, and Peter Desnoyers. Skylight—a window on shingled disk operation. In *13th USENIX Conference on File and Storage Technologies FAST 15*, pp. 135–149, 2015.

[13] Fenggang Wu, Ming-Chang Yang, Ziqi Fan, Baoquan Zhang, Xiongzi Ge, and David H.C. Du. Evaluating host aware smr drives. In *8th USENIX Workshop on Hot Topics in Storage and File Systems (HotStorage 16)*, 2016.

[14] Damien Le Moal and Adam Manzanares. libzbc. In *https://github.com/hgst/libzbc*, 2017.

[15] SNIA IOTTA Repository. Tpcc traces 1. In *http://iotta.snia.org/traces/131*, 2017.

[16] Rekha Pitchumani et al. Emulating a shingled write disk. In *IEEE 20th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, 2012.

発表文献

1. 佐藤佑紀, 早水悠登, 合田和生, 喜連川優, 最近の磁気ディスクドライブに於ける高遅延特性の観測とデータベース処理性能への影響の考察. 電子情報通信学会第 9 回データ工学と情報マネジメントに関するフォーラム／第 15 回日本データベース学会年次大会（DEIM), H5-4, 2017 年 3 月.

2. 佐藤佑紀, 早水悠登, 合田和生, 喜連川優, 入出力トレースを用いた Host-managed 方式 SMR 型ディスクドライブの性能エミュレーション環境の構築（仮題）. The 2nd cross-disciplinary Workshop on Computing Systems, Infrastructures, and Programming (xSIG 2018), 2018 年 5 月.（投稿予定）