

オンライン処理とバッチ処理が混在する環境における ディスク I/O 制御方式

田辺雅則^{†1} 横山和俊^{†2} 長尾 尚^{†1} 乃村能成^{†1} 谷口秀夫^{†1}

概要：銀行のオンラインシステムに代表される業務システムでは、オンライン処理とバッチ処理が別計算機で毎日実行される。しかし、計算機の処理能力向上とともに、計算機資源を十分に利用しない時間帯が生じ、分割損が生じている。一方、両処理の計算機資源の利用量は時間帯で異なるため、両処理を同じ計算機で実行するシステム構成とすることで、分割損を軽減し、計算機資源の利用効率の向上が期待できる。本稿では、ディスク I/O、特に両処理の書き込みデータ長が異なることに着目し、ディスクに対する I/O 要求の優先度を制御する方式を提案し、シミュレーション結果を報告する。

キーワード：ディスク、I/O、スケジューリング、オンライン処理、バッチ処理

1. はじめに

銀行のオンラインシステムに代表される業務システムでは、オンライン処理とバッチ処理に分類される処理が毎日実行される。オンライン処理は、処理時間が比較的短いものが多く、サービスの応答時間が重視される。また、利用者からの要求にもとづく処理であるため、トランザクション量（単位時間あたりの処理量）は時刻変化し、あらかじめ計画して処理を実行できない。このため、計算機資源の利用として、トランザクション量の変化に合わせた最適配分が重視される。一方、バッチ処理は、処理時間が短いものから長いものまで様々であり、スループットが重視される。また、バッチ処理は、定型的な処理であるため、あらかじめ計画した処理として定めた時刻に実行を開始できる。このため、計算機資源の利用として、定めた時間で最適な配分と効率的な利用が重視される。

これらの性質の違う処理を実行するため、業務システムは、両処理を別の計算機で実行するシステム構成とすることが多い。しかし、近年、計算機の処理能力が著しく向上したため、計算機資源を十分に利用しない時間帯が生じ、分割損が生じている。オンライン処理とバッチ処理の計算機資源の利用量は時間帯で異なるため、両処理を同じ計算機で実行するシステム構成とすることで、分割損を軽減し、計算機資源の利用効率の向上が期待できる。両処理を同じ計算機で実行する場合、オンライン処理の応答時間に与える影響を軽減する必要がある。このため、バッチ処理よりもオンライン処理を優先して実行するための優先制御が重要である。しかし、従来の CPU 資源の優先制御だけでは、ディスクへの I/O 要求が多く発生している場合、応答時間に影響を与える問題がある。これは、ディスクへの I/O 要求は、先に要求された I/O 要求が実行されるため、先にバッチ処理の長い処理時間の I/O 要求が実行された場合は、

オンライン処理の I/O 要求が待たされることで発生する。したがって、オンライン処理とバッチ処理を同じ計算機で実行する場合、オンライン処理を優先的に処理するためには、CPU リソースの優先制御に加えて、I/O の優先制御が必要となる。

ディスクに対する I/O 要求に関して、仮想環境における課題を解決する研究が行われている[1][2][3][4][5]。これらの研究では、各仮想環境における I/O 要求の優先制御の有効性を明らかにしている。しかし、仮想環境における I/O 要求の優先制御は、各仮想環境の I/O の使用量を各仮想環境に対してあらかじめ割り当てた割合に近づけることや仮想環境の I/O 要求が他の仮想環境に与える影響を軽減することを目的としている。このため、アプリケーションに対する優先制御と目的が異なる。アプリケーション特性を考慮して I/O 要求の待ち時間を短縮する手法[6][7]や、I/O 要求の処理時間を短縮する手法[8]が提案されている。これらはいずれもオンライン処理の I/O 要求の問題を解決するものではない。また、処理の性質をもとに処理の動作を予測し、処理のタイムスライスによる割り込みを制御することで、処理時間を改善することが提案されている[9]。しかし、計算機資源の利用効率を向上する観点からは、I/O 要求の優先度も制御する必要がある。これに対して、I/O 要求の処理時間を調整する制御法が提案されている[10]。この提案では、高い優先度を持つ I/O 要求が優先的に実行されるように I/O 要求の処理時間を調整することで、優先度の高い処理の I/O 要求の完了待ちによる待ち時間を短くできることを示している。しかし、応答時間およびスループットを重視する業務システムでは、I/O 要求の処理時間を調整することによる性能劣化が問題となる。

本稿では、オンライン処理とバッチ処理を混在して実行する環境において、バッチ処理が実行するディスクへの I/O 要求がオンライン処理に与える影響を述べる。また、オンライン処理の I/O 要求を優先して実行するディスク I/O 制御方式を提案し、オンライン処理の I/O 要求が実行されるまでの待ち時間の短縮とオンライン処理の処理時間の改善

^{†1} 岡山大学大学院自然科学研究科
Graduate School of Natural Science and Technology,
Okayama University

^{†2} 高知工科大学
Kochi University of Technology

が可能であることを述べる。

2. オンライン処理とバッチ処理が混在する環境

2.1 業務システムの特徴

銀行システムに代表される業務システムの処理は、オンライン処理とバッチ処理からなり、時間帯によって実行する処理量が異なる。その違いは、大きく昼間帯と夜間/深夜帯に分けられる。昼間帯は、オンライン処理の実行が多く、バッチ処理は少数実行される。一方、夜間/深夜帯は、バッチ処理の実行が主たる処理となり、オンライン処理は少数実行される。

業務システムについて、1日の運用で実行されるオンライン処理とバッチ処理の計算機資源の利用状況を図1に示す。ここで、計算機資源とは、主にCPU、メモリ、および外部記憶装置である。昼間帯はオンライン処理の計算機資源利用が多く、夜間/深夜帯はバッチ処理の計算機資源利用が多い。しかし、いずれの場合も、他方の処理（昼間帯はバッチ処理、夜間/深夜帯はオンライン処理）の計算機資源利用は少ないものの存在する。

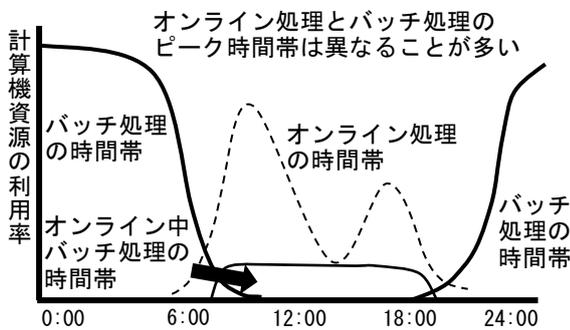


図1 一日の運用における計算機資源の利用率の推移

オンライン処理のトランザクション量は、時系列で変化するため、予測することは可能である。しかし、その予測精度は低く、また、何らかのイベントにより、極端なトランザクション量の増加が発生することも珍しくない。また、オンライン処理は、サービスに直結するため、適切な応答時間の保証が必須である。このため、オンライン処理が必要とする計算機資源として、余裕を持たせて用意している。

したがって、オンライン処理とバッチ処理を別の計算機で実行するシステム構成では、計算機資源の利用効率が低い時間帯と高い時間帯が発生し、計算機資源の分割損が発生する。また、オンライン処理を実行する計算機では、計算機資源に余裕を持たせるため、効率的とはいえない。

2.2 課題

オンライン処理とバッチ処理は、主に実行される時間帯が異なることから、両処理を混在して実行すること、つまり一つの計算機で実行することで、計算機資源の利用効率

を向上させることができる。しかし、両処理を混在して実行するには、以下の課題がある。

(課題1) 計算機資源の利用率の平準化法の確立
システム運用開始後において、計算機資源の追加は、費用や作業および継続したサービス提供の観点から困難である。このため、オンライン処理とバッチ処理を混在して実行する場合、あらかじめ処理を計画して実行できるバッチ処理に対しては、オンライン処理に影響を与えないシステム構成とする。しかし、運用開始後、計算機資源の利用状況が想定した状況と異なり、特定の時間帯において想定した以上の計算機資源を使用することがある。このような状況となった場合は、計算機資源の追加ができないため、オンライン処理とバッチ処理の計算機資源の利用率を平準化する。つまり、計画的に実行できるバッチ処理の実行計画を再設計し、計算機資源の利用率の平準化を図る手法の確立が必要である。

(課題2) 処理の性質に合わせたI/O制御法の確立
多くのオンライン処理は、利用者からの要求にもとづいて実行され、応答時間を重視することから、処理時間が短いものが多い。このため、CPUの利用時間が短く、ディスクへのI/O要求も少ないものが多い。一方、バッチ処理は、オンライン処理に比べ、多量なデータのI/O要求を実行し、CPUの利用量も多い。このため、オンライン処理の応答時間を保証するには、CPUの優先制御だけでなく、I/O要求の優先制御も必要である。具体的には、例えば、ディスクドライバはI/O要求を並行して処理することができないため、I/O要求の実行時間が長いバッチ処理が実行された場合、オンライン処理のI/O要求は待たされてしまう。この様子を図2に示す。図2では、オンライン処理(CPU)とバッチ処理(CPU)は、CPUリソースを利用し、計算処理を実行している。バッチ処理(CPU)は、I/O要求をディスクドライバに依頼し、ディスクドライバがI/O要求を処理する。少し遅れてオンライン処理(CPU)も同様にI/O要求をディスクドライバに依頼する。しかし、このときディスクドライバはバッチ処理のI/O要求を実行しているため、オンライン処理(CPU)のI/O要求は待たされる。バッチ処理(CPU)のI/O要求が完了した後、オンライン処理(CPU)のI/O要求が実行される。オンライン処理のI/O要求が、

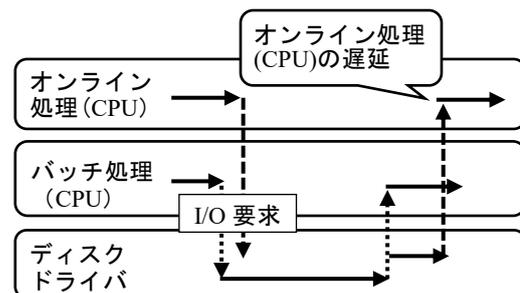


図2 I/O要求と処理の遅延

先に実行されているバッチ処理の I/O 要求と重なった場合、オンライン処理に対する CPU 割り当ての優先制御により、オンライン処理が優先的に処理されていた場合においても、バッチ処理の I/O 要求の完了までは、オンライン処理の I/O 要求は実行されない。これは、ディスクドライブは、ディスク装置の I/O 要求の完了による割り込み制御によって、I/O 要求の実行が完了した後の処理を開始するためである。このため、オンライン処理の I/O 要求は、バッチ処理の I/O 要求の完了を待たされる。つまり、CPU の優先制御だけでは、オンライン処理の処理時間を短縮できない状況が発生する。したがって、処理の性質に合わせた I/O 制御法の確立が必要である。

なお、(課題 1) で述べた手法の確立は、毎日運用する中で実行計画を状況に合わせて動的に見直す手法の確立であり、非常に難しい。そこで、以降では、(課題 2) について述べる。

3. データ長が異なる I/O 要求の処理時間

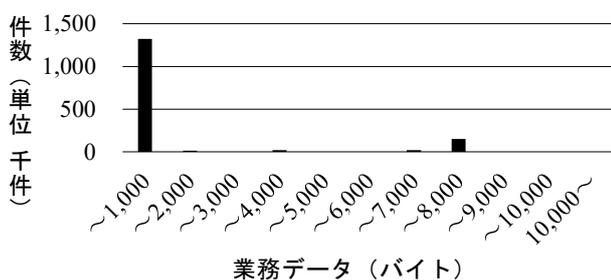
3.1 オンライン処理とバッチ処理の I/O 要求の比較

3.1.1 処理流れ

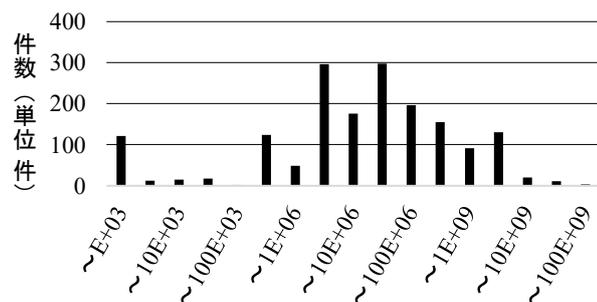
オンライン処理における 1 つのトランザクション処理の主な処理流れは、利用者からの要求内容をディスクに書き込み、処理を実行し、実行結果をディスクに書き込み、利用者にその結果を返す。バッチ処理の主な処理流れは、計算処理を実行し、結果をディスクに書き込むことを繰り返し行う。

3.1.2 ディスクに書き込むデータ長

オンライン処理とバッチ処理は、ディスクに書き込むデ



(A) オンライン処理



(B) バッチ処理

図 3 書き込みデータ長の分布

ータ長に違いがある。例えば、銀行オンラインシステムに代表される業務システムについて、オンライン処理とバッチ処理の業務データのデータ長の分布を図 3 に示す。図 3 (A)より、オンライン処理の書き込みデータ長は、1,000 バイト以下が大半を占める。一方、図 3 (B)より、バッチ処理の書き込みデータ長は、数 MB から 100MB のファイルサイズが多い。

3.2 書き込みデータ長と書き込み時間の分析

3.2.1 測定環境

データ書き込み時間の測定に使用した計算機の諸元を表 1 に示す。測定環境では、処理時間などを評価するための評価用ログ等を書き出す評価用ログ等ディスクと I/O 要求によりデータを書き出すデータ用ディスクがある。データ書き込み時間の測定はデータ用ディスクで行う。評価用ログ等をディスクに書き出すことが、データを書き込む時間を測定する I/O 要求に与える影響を軽減するため、データ用ディスクおよび I/O パスを評価用ログ等用ディスクと物理的に分ける。また、データ用ディスクは、ファイルシステムのキャッシュの影響をなくするため、UFS のファイルキャッシュを使用しない設定とする。評価用ログ等のディスクは、評価用ログの書き込み時間が評価プログラムの処理時間に与える影響を少なくするため、ファイルシステムのキャッシュを使用する設定とする。

表 1 測定に使用した計算機環境の諸元

CPU	Intel i5 3.2GHz 4 コア ハイパースレッドなし	
データ用 ディスク	I/O パス	SATA3.0
	DISK	7400 回転 32MB キャッシュ
	ファイルシステム	UFS
	ファイルシステム のキャッシュ	使用しない
評価用ロ グ等用デ ィスク	I/O パス	SATA6.0
	DISK	7400 回転 16MB キャッシュ
	ファイルシステム	UFS
	ファイルシステム のキャッシュ	使用する
OS	FreeBSD 11.0-RELEASE-p1	

3.2.2 評価プログラム

3.1.1 項で述べた各処理の流れを基に、評価プログラムを作成した。評価プログラムの処理流れを図 4 に示す。図 4(A)は、オンライン処理の 1 トランザクション処理を擬似した処理 (オンライン擬似処理) であり、要求内容データ書き出し、処理実行、結果データ書き出し、結果通知、からなる。この処理を繰り返し行う。図 4(B)は、バッチ処理を擬似した処理 (バッチ擬似処理) であり、計算処理と結果デ

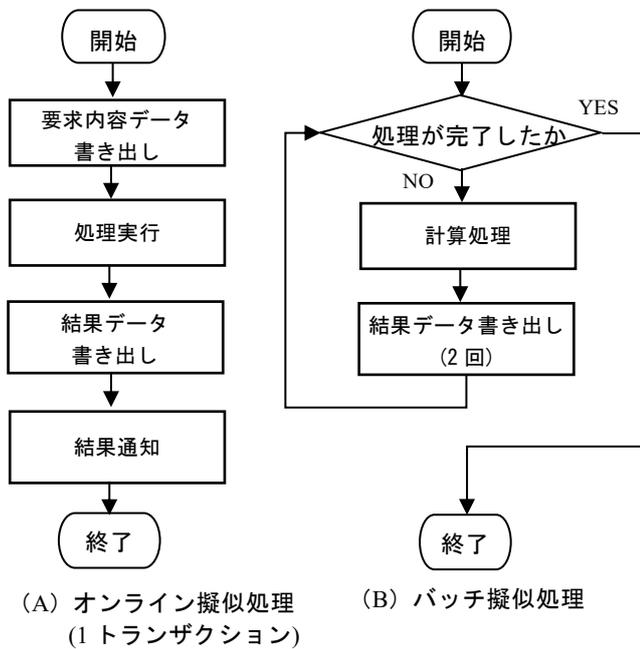


図4 評価プログラムの処理流れ

ータ書き出し (2 回) の処理を繰り返す。結果データ書き出しでは、大きなデータを 2 回に分けて書き出す。

書き出しデータ長は、オンライン擬似処理の場合 1,000 バイト、バッチ擬似処理の場合 1,000,000 バイトとした。

オンライン処理やバッチ処理の各 CPU 処理時間は総データ書き込み時間と同様であること、およびオンライン処理とバッチ処理が共存実行される環境の構築を満足するために、以下の設定とした。具体的には、オンライン擬似処理のトランザクション数を 50 回、バッチ擬似処理の繰り返し回数を 5 回とした。オンライン擬似処理の場合、CPU 処理時間を 0 秒として 50 トランザクションを実行した際の総データ書き込み時間は 92.63 ミリ秒であった。つまり 1 トランザクションのデータ書き込み時間は 1.85 ミリ秒である。また、バッチ擬似処理の場合、CPU 処理時間を 0 秒として 5 回実行を繰り返した際の総データ書き込み時間は 297.05 ミリ秒であった。つまり結果データ書き出し (2 回) のデータ書き込み時間は 59.41 ミリ秒である。処理実行 (CPU) の時間を両擬似処理のデータ書き込み時間と同様とするため、オンライン擬似処理 (1 トランザクション) の処理実行 (CPU) の時間を 2 ミリ秒とし、バッチ擬似処理 (CPU) の 1 回の計算処理の時間を 60 ミリ秒とした。なお、トランザクション処理の結果通知の処理は、他に比べ簡単な処理であるため 0 ミリ秒とした。以上より、トランザクション数が 50 のオンライン擬似処理は 192.5 ミリ秒、繰り返し回数が 5 のバッチ擬似処理は 597.05 ミリ秒とした。

3.2.3 結果と考察

2 つの評価プログラムを同時に、1 (各 1 つ)、2、5、10 と実行した場合のデータ書き込み時間について、書き込み

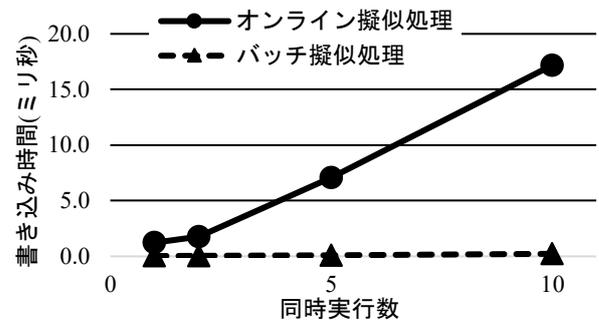


図5 同時実行数と平均書き込み時間の関係 (1,000バイト換算)

データ長とデータ書き込み時間の関係に着目するため、1,000 バイトあたりのデータ書き込み時間に換算した結果を図 5 に示す。

図 5 より、同時実行数が増加するにしたがって、オンライン擬似処理のデータ書き込み時間が大きく増加している。一方、バッチ擬似処理のデータ書き込み時間の増加は非常に小さい。これは、書き込みデータ長に関して、オンライン擬似処理より長いバッチ擬似処理の同時実行数の増加により、I/O 処理時間の長い write () システムコールの実行回数が増加し、1,000 バイトのデータを書き込むオンライン擬似処理の write () システムコールの実行が待たされる場合が増加してしまうためである。バッチ擬似処理は、オンライン擬似処理よりも write () システムコールの実行回数が少なく、同時実行数の増加の影響が小さい。

図 5 の同時実行数が 1 のデータ書き込み時間は、1,000 バイトの場合は 1.23 ミリ秒であり、1,000,000 バイトの場合は 1,000 バイト換算で 0.032 ミリ秒であった。書き込みデータ長は、大きい方が単位サイズあたりのデータ書き込み時間は短くなる。したがって、なるべく大きな単位で書き込み処理を実行する方が処理時間は短くなる。

大きなデータサイズを書き込むバッチ処理と小さなデータサイズを書き込むオンライン処理が同じ環境で実行される場合、実行数によってオンライン処理の I/O 要求の処理時間が長くなり、オンライン処理のレスポンスが劣化するといえる。

4. ディスク I/O 制御方式

4.1 I/O 制御機構

オンライン処理とバッチ処理を同じ計算機で実行する場合、書き込みデータ長の小さいオンライン処理が、バッチ処理の書き込みによる影響を受けやすい。そこで、オンライン処理の I/O 要求を優先的に処理するために、オンライン処理の書き込みデータ長が小さいことに着目し、書き込みデータ長が小さい I/O 要求を優先的に実行する優先制御を行うディスク I/O 制御方式を提案する。ディスク I/O 制御方式の優先制御の基準を以下に示す。

- 書き込みデータ長が小の I/O 要求を高優先度 (オン

ライン処理の I/O 要求に相当)

- 書き込みデータ長が大の I/O 要求を低優先度 (バッチ処理の I/O 要求に相当)

I/O 制御機構の構成を図 6 に示す。

(1) I/O 処理実行待ちキュー機能

各処理からシステムコールによって要求される I/O 要求を管理する。同じ優先度の I/O 要求は、到着順とする。I/O 要求は、小さいデータ長と大きいデータ長の 2 つのタイプに分類し、I/O 処理実行待ちキューは、それぞれのタイプに対応するキューとする。つなぐキューは、システムコールの引数で指定されたデータ長で決定する。

(2) I/O サイズ分割機能

I/O 要求のデータの大きさをハードウェアの性能に応じて、適切なサイズに分割する。データを分割することで、1 つ 1 つの I/O 要求の処理時間を短くすることができる。そのため、大きなデータを書き込む場合でも一定の大きさに分割して I/O 要求を実行するため、小さなデータを書き込む I/O 要求が待たされるケースを少なくできる。すなわち、オンライン処理の I/O 要求が実行されるまでの待ち時間を短くすることができる。

(3) I/O 優先制御機能

キューから I/O 要求を取り出して、ディスクドライブに I/O 要求を渡す。小サイズ・キューに I/O 要求が存在した場合は、小サイズ・キューから I/O 要求を取り出す。小サイズ・キューに I/O 要求がない場合は、大サイズ・キューから I/O 要求を取り出す。ディスクドライブは、I/O 要求を並行実行できないため、I/O 優先制御がディスクドライブに渡す I/O 要求は一度に 1 つである。ディスクドライブは、受け付けた I/O 要求を実行し、I/O 要求の実行完了後、ハードウェア割り込みにより、呼び出し元に戻る。

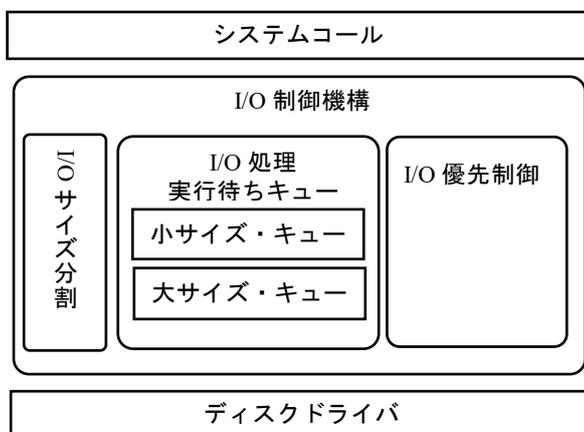


図 6 ディスク I/O 制御方式

4.2 I/O 制御機構の処理流れ

I/O 制御の処理の流れを図 7 に示し、説明する。

- ① 処理は I/O 要求のシステムコールを実行し、OS に I/O 要求の実行が要求される。

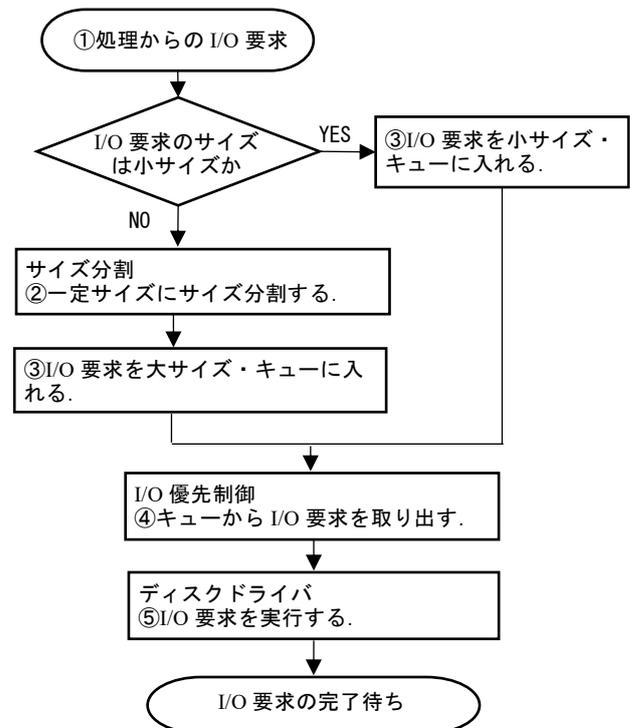


図 7 I/O 制御機構の処理流れ

- ② I/O 要求の書き込みサイズが大きい場合、一定サイズにサイズ分割する。
- ③ I/O 要求のサイズにもとづきキューに入れる。
- ④ キューから I/O 要求を取り出す。優先度の高い小サイズ・キューから優先的に I/O 要求を取り出す。
- ⑤ I/O 要求を実行し、処理結果を待つ。I/O 要求は、その割り込み処理により、呼び出し元に戻る。

5. シミュレーションによる評価

5.1 観点

I/O 要求が多数ある状態は、I/O 要求の実行待ちが多数発生するため、ディスクビジー率は高くなる。そこで、以下の 2 点を評価し、ディスク I/O 制御方式の有効性を示す、
(評価 1) ディスクビジー率と処理時間
(評価 2) ディスクビジー率と I/O 要求開始の待ち時間
ディスクビジー率は、高いケース、低いケース、その中間のケースの 3 通りとする。本シミュレーションにおけるディスクビジー率は、処理時間を分析する評価対象時間に対して I/O 要求が割り当てられている時間の割合とする。

本シミュレーションでは、以下の 3 つの制御方法に対して、前述の評価 1 と評価 2 を実施する、

- ①優先度制御なし
- ②CPU 優先制御のみ (オンライン処理優先)
- ③CPU+I/O 優先制御 (オンライン処理優先)

5.2 計算機資源の割り当て

5.2.1 I/O サイズ分割と計算機の諸元

I/O サイズ分割による書き込みデータのデータサイズは、

1,000 バイトの I/O 要求の処理時間（書き込み）と同様の処理時間となる，なるべく大きいデータサイズとする．具体的には，測定環境において，1,000 バイトのデータ書き込み時間が 0.926 ミリ秒，10,000 バイトのデータ書き込み時間が 1.26 ミリ秒であったため，I/O サイズ分割による書き込みデータのデータサイズを 10,000 バイトとする．

シミュレーションで用いる計算機は，CPU 数を 4 個，I/O パス数を 1 パスとし，1,000 バイトおよび 10,000 バイトの I/O 要求の処理時間（書き込み）を 1 ミリ秒とする．

5.2.2 CPU と I/O 要求の割り当て

シミュレーションにおける CPU 割り当てと I/O 要求割り当ての規則を以下に示す．

(1) CPU 割り当ての規則

- (A) 複数の処理が CPU の割り当てを待っている場合は，時間的に早く要求された処理を CPU に割り当てる．
- (B) CPU のタイムスライスは 10 ミリ秒とする．10 ミリ秒単位で，次の実行待ちの処理に CPU を割り当てる．
- (C) 処理の割り当てられていない空き CPU が複数ある場合，CPU の割り当てを待つ処理の CPU 割り当てが 1 回目場合は，いずれかの空き CPU に処理を割り当てる．2 回目以降の場合は，1 回目に割り当てた CPU に割り当てる．つまり，2 回目以降の CPU 割り当ては，1 回目に割り当てた CPU に固定とする．

(2) I/O 要求割り当ての規則

- (A) 同じ時間に複数の処理による I/O 要求があった場合，時間的に早く要求された I/O 要求を実行する．
- (B) I/O のタイムスライスは 1 ミリ秒とする．1 ミリ秒単位で，次の実行待ちをしている処理の I/O 要求があれば，その I/O 要求を割り当てる．

5.2.3 優先度制御による CPU と I/O 要求の割り当て

CPU および I/O 割り当ての優先度制御をシミュレーションに適用する．優先度制御の規則を以下に示す．

(1) CPU 割り当ての優先度制御の規則

- (A) 同じ時間に複数の処理が CPU の割り当てを待つ場合，優先度の高い処理を CPU に割り当てる．同じ優先度の場合は，時間的に早く要求された処理を先に CPU に割り当てる．

(2) I/O 割り当ての優先度制御の規則

- (A) 同じ時間に複数の処理が I/O 要求があった場合，優先度の高い処理の I/O 要求を割り当てる．
- (B) 実行待ちの I/O 要求の優先度を比較し，同じ優先度の場合は，直前まで実行中の I/O 要求を割り当てる．高い優先度の場合は，高い優先度の I/O 要求を割り当てる．

5.3 処理モデル

本シミュレーションの処理モデルは，オンライン疑似処理とバッチ疑似処理と同等とする．

表 2 にオンライン処理のモデルを示す．オンライン処理

は，3 通りのディスクビジー率のケースで同じ処理モデルを使用する．オンライン処理は，利用者からの要求にもとづいて実行され，計画的に資源を割り当てることができないため，CPU 使用率に余裕が出るようにシステムを設計することが多い．そこで，シミュレーションでは，オンライン処理の CPU 使用率が 50%程度となるようにオンライン処理の起動タイミングを決定する．

表 3 にケース 1（ディスクビジー率が低），ケース 2（ディスクビジー率が高と低の間），ケース 3（ディスクビジー率が高）のバッチ処理のモデルを示す．ディスクビジー率の増減は，バッチ処理の I/O 要求による総書き込みデータ長を変更して実現する．I/O サイズ分割による書き込みデータ長のデータサイズは，10,000 バイトとする．

表 2 オンライン処理のモデル

計算処理時間	100 ミリ秒
I/O 要求の間隔	処理開始と処理終了で各 1 回
総 I/O 要求数	2 回
1 回の書き込みデータ長	1,000 バイト

表 3 バッチ処理のモデル

	ケース 1	ケース 2	ケース 3
ディスクビジー率	低	中	高
計算処理時間	10 秒		
I/O 要求間隔	均等間隔		
総 I/O 要求数	20 回 (0.5 秒に 1 回の割合)		
I/O 要求の総処理時間	1,000 ミリ秒	2,000 ミリ秒	4,000 ミリ秒
総書き込みデータ長	10,000,000 バイト	20,000,000 バイト	40,000,000 バイト
1 回の書き込みデータ長	500,000 バイト	1,000,000 バイト	2,000,000 バイト
サイズ分割による 1 回の書き込みデータ長	10,000 バイト	10,000 バイト	10,000 バイト

5.4 結果と考察

処理時間や I/O 要求の開始の待ち時間等を評価するために使用するシミュレーションの評価区間は，オンライン処理とバッチ処理が混在する時間を評価区間とする．シミュレーションの開始直後は，I/O 要求がオンライン処理に偏り，シミュレーションの終了間際はバッチ処理に偏る傾向があるため，実行時間の中心から前後 5 秒とし，5 秒から 15 秒とする．シミュレーションにより計測した各ケースのディスクビジー率と総 CPU 使用率を表 4 に示す．また，表 5 にオンライン処理とバッチ処理の各 CPU 使用率を示す．

表4 ディスクビジー率と総CPU使用率

	ケース1 単位 (%)		ケース2 単位 (%)		ケース3 単位 (%)	
	DISK	CPU	DISK	CPU	DISK	CPU
優先制御なし	23.5	97.3	40.0	95.5	72.2	91.1
CPU優先制御のみ	16.5	80.0	28.2	79.5	53.7	80.3
CPU+I/O優先制御	16.4	80.0	27.9	79.6	52.8	80.0

表5 オンライン処理とバッチ処理のCPU使用率
(ON オンライン処理, BT バッチ処理)

	ケース1 単位 (%)		ケース2 単位 (%)		ケース3 単位 (%)	
	ON	BT	ON	BT	ON	BT
優先制御なし	49.8	47.5	50.2	45.3	49.9	41.2
CPU優先制御のみ	50.0	30.0	53.5	26.0	49.3	31.0
CPU+I/O優先制御	50.0	30.0	50.0	29.6	50.0	30.0

(1) 評価1 (平均処理時間)

各ケースのオンライン処理とバッチ処理の平均処理時間を図9に示す。図9より以下のことがわかる。

(A) ケース1から3のいずれもオンライン処理のI/O優先制御を実施した場合、オンライン処理の平均処理時間が改善される結果を得た。優先制御なしおよびCPU優先制御のみの場合、オンライン処理の平均処理時間は、ケース1、

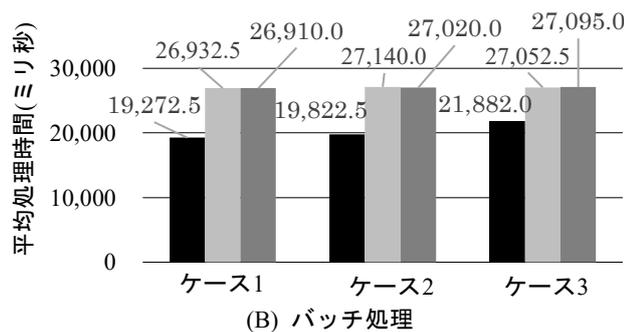
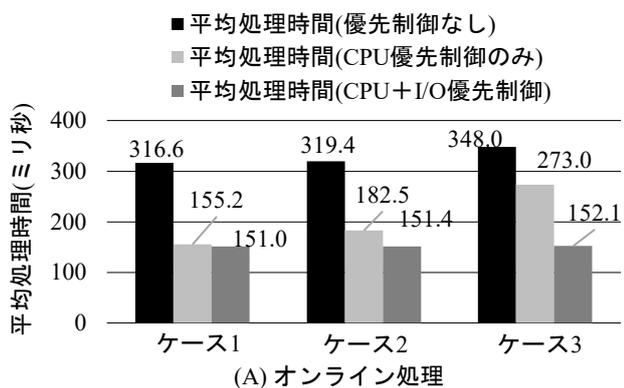


図9 優先制御方法の平均処理時間

ケース2、ケース3の順番でケース3が一番長くなっている。すなわち、I/O優先制御を実施しない場合、ディスクビジー率が高くなるにしたがって、オンライン処理の平均処理時間が長くなる傾向にあることがわかる。この理由は、バッチ処理によるI/O要求を実行している時間が増加し、I/O要求の開始待ち時間が長くなるオンライン処理が増加したためである。一方、CPU+I/O優先制御ありのケースでは、ケース1から3のいずれにおいても、オンライン処理の平均処理時間は微増である。

(B) バッチ処理について、CPU優先制御のみおよびCPU+I/O優先制御の平均処理時間は、ケース1から3のいずれのケースでも優先制御なしの平均処理時間に比べて長くなった。また、CPU優先制御のみおよびCPU+I/O優先制御の平均処理時間は、ケース1からケース3のいずれのケースでも違いは少ない。表6より、CPU優先制御およびCPU+I/O優先制御のバッチ処理のCPU使用率は、優先制御なしに比べて低い。つまり、オンライン処理に優先的にCPU割り当てが行われた結果、バッチ処理のCPU割り当てが減ったためであり、I/O優先制御による平均処理時間の増加はなかった。つまり、I/O優先制御は、バッチ処理の平均処理時間に影響を与えていない。

(2) 評価2 (平均I/O要求開始待ち時間)

各ケースのオンライン処理とバッチ処理がI/O要求を開始してから、I/O要求の処理開始までの平均I/O開始待ち時間を図10に示す。図10より以下のことがわかる。

(A) オンライン処理の平均I/O要求開始待ち時間は、CPU+I/O優先制御の場合、ケース1から3すべて8.0ミリ秒で

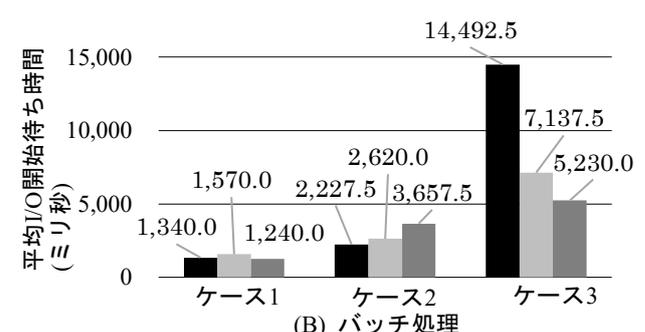
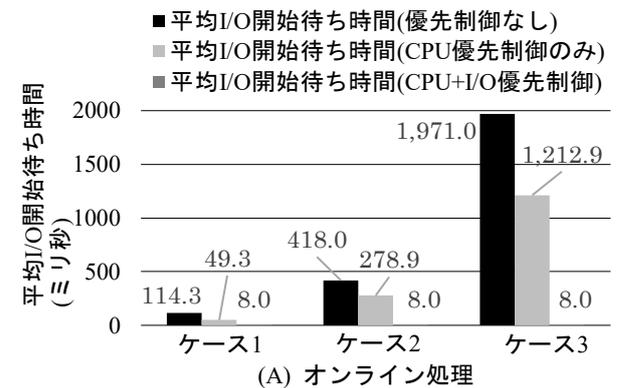


図10 優先制御方法の平均I/O要求開始待ち時間

あり、待ち時間の増加は見られない。しかし、優先制御なしおよび CPU 優先制御のみの場合、大幅に悪化している。つまり、ディスクビジー率が高い場合において、CPU+I/O 優先制御は、オンライン処理の平均 I/O 要求開始待ち時間に対して効果が顕著である。

(B) バッチ処理の平均 I/O 要求開始待ち時間は、CPU 優先制御のみおよび CPU+I/O 優先制御の場合、ケース 1 から 3 で両者に大きな差はない。また、CPU+I/O 優先制御の方が CPU 優先制御のみと比べて平均 I/O 要求開始待ち時間が短い。したがって、バッチ処理に対しても CPU+I/O 優先制御の効果がある。しかし、ケース 2 では、CPU+I/O 優先制御の平均 I/O 要求開始待ち時間が、CPU 優先制御のみと比べて少し長くなっている。CPU 割り当てのタイミングによっては、I/O 要求開始待ち時間が増加することがある。

6. おわりに

オンライン処理とバッチ処理が混在して実行する環境において、CPU の優先制御だけではオンライン処理が優先的に処理されない場合があることを示した。そして、オンライン処理の I/O 要求の書き込みサイズが小さいことに着目し、各処理の I/O 要求の実行を制御するディスク I/O 制御方式を提案した。シミュレーションにより、書き込みサイズの小さい I/O 要求を優先的に実行することで、オンライン処理の処理時間の短縮が可能であることを示した。

オンライン処理の I/O 要求の優先制御を実行するためには、バッチ処理の I/O 要求の処理時間を考慮したサイズ分割が必要である。一度に書き込むデータ長を小さくした場合、単位サイズあたりのデータ書き込み時間は増加する。書き込みデータ長の大きいバッチ処理のデータ書き込み時間の増加を防ぐためには、一定の大きさ以上とする必要がある。I/O サイズ分割による書き込みデータ長は、書き込みデータ長とデータ書き込み時間を用いて統計的な手法で決定するなどが考えられる。

残された課題として、I/O 分割による I/O サイズの決定方法および本制御方式の実装と評価がある。

参考文献

- [1] Filip Blagojevic, Cyril Guyot, Qingbo Wang, Timothy Tsai, Robert Mateescu and Zvonimir Bandic, "Priority IO Scheduling in the Cloud", 5th USENIX Workshop on Hot Topic in Cloud Computing, JUNE 25-26, 2013.
- [2] Mukil Kesavan, Ada Gavrilovska, Karsten Schwan, "On Disk I/O Scheduling in Virtual Machines", The 2nd Workshop on I/O Virtualization (WIOV' 10), March 13, 2010, USA
- [3] Ziye Yang, Haifeng Fang, Yingjun Wu, Chunqi Li, Bin Zhao, H. Howie Huang, "Understanding the Effects of Hypervisor I/O Scheduling for Virtual Machine Performance Interference", 2012 IEEE 4th International Conference on Cloud Computing Technology and Science
- [4] 北原友恵, 福本佳史, 小田哲, 小西隆介, 湯口徹, "マルチテナント環境における I/O 性能保証方式の提案", 情報処理学会研究報告システムソフトウェアとオペレーティングシステム

- Vol.2014-OS-128 No.7 2014/3/7
- [5] 水野 和彦, 今田 貴之, "仮想環境における I/O 性能評価及び最適化手法提案", 情報処理学会コンピュータシステム・シンポジウム論文集 p.86 - 93 2016
- [6] Ramon Nou, Jacobo Giralto, Toni Cortes, "Automatic I/O scheduler selection through online workload analysis", The 9th IEEE International Conference on Autonomic and Trusted Computing (ACT 2012), Sep.04-07, 2012
- [7] 片上達也, 田畑利宏, 谷口秀夫, "ファイル操作のシステムコール発行頻度に基づくバッファキャッシュ制御法の提案", 情報処理学会論文誌 コンピュータシステム Vol.3 No.1 50-60 (Mar. 2010)
- [8] 田邨 優人, 中島 耕太, 山本 昌生, 前田 宗則, "CPU 処理と I/O 処理の高速性を両立する I/O 制御機構", 情報処理学会研究報告システムソフトウェアとオペレーティングシステム Vol.2017-OS-141 No.22 2017/7/19
- [9] Yoshinori AOKI, Sukanya SURANAUWARAT, Hideo TANIGUCHI, "A Load Distribution Scheme for a New Transaction Service Considering the Pre-Loaded Services", IEICE TRANS. INF. & SYST., VOL. E82-D, No.22, Nov. 1999
- [10] 長尾尚, 谷口秀夫, "入出力要求数の制御によりサービス時間を調整する制御法の実現と評価", 電子情報通信学会論文誌, D Vol. J94-D No.7 pp.1047-1057