

オンライン処理とバッチ処理の処理負荷を分散制御する 入出力制御法の仮想環境における評価

田辺雅則^{1,a)} 横山和俊² 島谷隼生³ 佐藤将也¹ 長尾 尚¹ 谷口秀夫¹

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実行される。しかし、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。そこで、両処理を同じ計算機で実行するシステム構成が可能である。このシステム構成においては、両処理の負荷を長期だけではなく短期にも時間的に分散させ、バッチ処理の負荷がオンライン処理の応答時間に悪影響を与えないようにすることが強く求められる。著者らはオンライン処理とバッチ処理のファイル I/O の特徴として入出力データ長に着目した入出力制御法を提案し評価した。しかし、計算機の処理能力向上により、業務システムをプロセッサ数の多い物理計算機、さらには仮想計算機で実行することも想定される。そのため、これらの計算機においても両処理の負荷を長期だけではなく短期にも時間的に分散させ、バッチ処理の負荷がオンライン処理の応答時間に悪影響を与えないようにすることが必要である。本稿では、プロセッサ数の多い物理計算機、仮想計算機における入出力制御法の有効性について評価した結果を述べる。

1. はじめに

オンライン処理とバッチ処理が行われる業務システムの代表例として、銀行のオンラインシステムがある。オンライン処理は応答時間への要求が厳しいため、バッチ処理の負荷がオンライン処理の応答時間に悪影響を与えないようにすることが強く求められる。このため、オンライン処理とバッチ処理は別計算機で実行される。しかし、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、計算機の処理能力向上とともに、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、計算機資源の利用効率の向上が期待できる。オンライン処理は主に昼間帯に実行されるため、バッチ処理を主に夜間帯に実行することで、長期の時間的な負荷分散は可能である。しかし、バッチ処理の増加とともに、バッチ処理を昼間帯に実行することも必要になっているため、短期の時間的な負荷分散をできることが必要である。

計算機資源の利用量を処理の特徴にあわせて負荷分散できる方法として、プロセス優先度によるプロセッサ割り当て制御法がある。しかし、プロセスの処理やプロセスが発行したシステムコール処理は、この制御法によって負荷分散できるが、割り込み処理はこれらの処理に比べ優先して実行されるため負荷分散できない。例えば、低優先度のプロセスが依頼した入出力要求による割り込み処理は、高優先度のプロセスの処理より優先して実行される。したがって、入出力処理が比較的多いオンライン処理やバッチ処理においては、不十分である。著者らは、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で、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ことに着目し、書き込みデータ長を考慮した入出力制御法を提案し、物理計算機における有効性を評価した[1]。一方、計算機の高性能化により、実システムでは、プロセッサ数の多い物理計算機の利用や計算機統合のための仮想計算機の利用が増えている。さらに、クラウド上のサービスとして仮想計算機を利用することも多くなっている。このため、プロセッサ数の多い物理計算機や仮想計算機における評価が重要である。

本稿では、プロセッサ数の多い物理計算機および仮想計算機における入出力制御法の評価を述べる。具体的には、オンライン処理とバッチ処理のアプリケーションモデルを使用し、プロセッサ数が 4 個と 8 個の物理計算機および仮想計算機において、提案制御法によりオンライン処理の処理時間が短縮可能であることを示す。

2. オンライン処理とバッチ処理の処理負荷を分散制御する入出力制御法^[1]

2.1 オンライン処理とバッチ処理の特徴

オンライン処理とバッチ処理は、ディスクに書き込むデータ長に大きな違いがある。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、バッチ処理の書き込みデータ長は、数 MB から 100MB である。

オンライン処理の平均処理時間は、約 200 ミリ秒から 370 ミリ秒である。一方、バッチ処理の処理時間は、短いものから長いものまで様々である。一日に実行されるバッチ処理の処理時間を単純平均した場合、バッチ処理の平均処理時間は約 30 秒である。

2.2 ディスク I/O 制御方式

データ長を考慮したディスク I/O 制御方式の基本構造を図 1 に示し、以下に説明する。

(1) I/O サイズ分割機能

書き込みデータ長が非常に大きいものについて分割し(図 1 では規定値として「大サイズ」としている)、一つの I/O 要求の処理時間を短くする。これにより、書き込みデータ長が非常に大きいバッチ処理の I/O 要求の処理時

1 岡山大学大学院自然科学研究科
Graduate School of Natural Science and Technology, Okayama University,
Okayama 700-8530, Japan
2 高知工科大学情報学群
School of Information, Kochi University of Technology,
Kochi 782-8502, Japan
3 岡山大学工学部情報系学科
Department of Information Technology, Faculty of Engineering,
Okayama University, Okayama 700-8530, Japan
a) tanabems@s.okayama-u.ac.jp

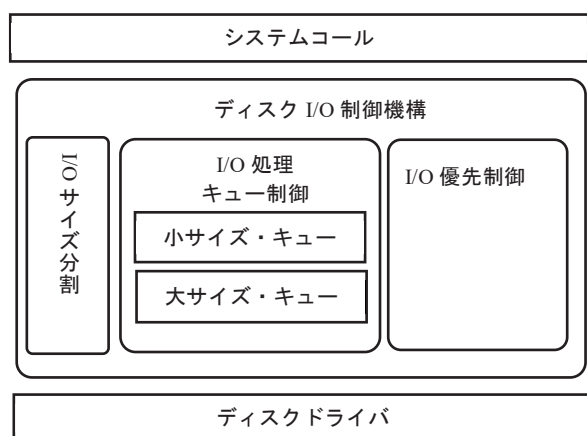


図1 ディスク I/O 制御方式の基本構造

間を分割して小さくできるため、オンライン処理への影響を抑制できる。

(2) I/O 処理キュー制御機能

I/O サイズの違い毎に実行待ちキューを用意して管理する。各キューは I/O 要求到着順である。小サイズ・キューは、I/O サイズが小さい I/O 要求のキューであり、オンライン処理の I/O 要求に相当する。一方、大サイズ・キューは、I/O サイズ分割機能によって分割された I/O 要求のキューであり、バッチ処理の I/O 要求に相当する。

(3) I/O 優先制御機能

小サイズ・キューの I/O 要求を大サイズ・キューの I/O 要求より優先させて、ディスクドライバに I/O 要求の実行を依頼する。つまり、I/O サイズが小さい I/O 要求を優先して実行する。なお、ディスクドライバ処理中に新たな I/O 要求が発生しても、優先制御を確実にを行うために、I/O 優先制御がディスクドライバに依頼する処理は一度に一つである。これにより、データ長が短いオンライン処理の実 I/O を優先的に実行し、バッチ処理の I/O 要求完了後の処理の影響を抑える。

3. 評価環境

3.1 観点

文献[1]と同様に、以下の観点で評価する。

(観点1) オンライン処理の負荷量 (TPS) とオンライン処理の処理時間の関係を明らかにすること。ディスク I/O 制御方式を使用したオンライン処理の処理時間が、プロセス優先度によるプロセッサ割り当て制御法を使用したオンライン処理の処理時間よりも短くなること。

(観点2) ディスク I/O 制御方式を使用したオンライン処理の I/O 要求の処理時間が短くなること。また、プロセス優先度によるプロセッサ割り当て制御法ではオンライン処理の I/O 要求の処理時間が短くならないこと。

また、評価対象の優先制御方式は、以下の3つである。

① I/O 要求優先 (オンライン処理優先)

ディスク I/O 制御方式を使用した優先制御によって、オ

ンライン処理の I/O 要求を優先的に実行する。

② CPU 優先 (オンライン処理優先)

プロセス優先度によるプロセッサ割り当て制御法によって、オンライン処理を優先的に実行する。オンライン処理の I/O 要求は優先的に実行しない。

③ 優先なし

I/O 要求の優先制御およびプロセス優先度によるプロセッサ割り当て制御を行わない。

物理計算機のプロセッサ数 (コア数) は、4 個と 8 個の 2 種類とする。ディスク I/O 制御方式の評価に使用する物理計算機の諸元を表 1 に示し、仮想計算機の諸元を表 2 に示す。オンライン処理およびバッチ処理がデータを書き出すために使用する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 のキャッシュを使用しない設定とする。

バッチ処理の I/O 要求について、I/O 要求を処理する効率は、書き込みデータ長が大きいほど良い。ところが、書き込みデータ長が大きいほど、オンライン処理の I/O 要求の処理開始までの待ち時間は長くなり、オンライン処理への影響が大きい。そこで、大サイズはオンライン処理の I/O 要求の処理時間と同等で、かつ、できるだけ大きいサイズに設定する。具体的には、大サイズはオンライン処理の 1,000 バイトのデータ書き込み時間と同程度の書き込み時間となるデータ長とする。本評価では、分割後の I/O サイズにつ

表1 物理計算機環境の諸元

		計算機 1	計算機 2
プロセッサ数 (コア数)		4	8
CPU		Intel Xeon E5-2609 1.70GHz ハイパースレッドなし	
メモリ		8GB	
ディスク	I/O パス	SATA3.0	
	DISK	5400 回転 64MB キャッシュ	
	ファイルシステム	UFS キャッシュは使用しない	
OS		FreeBSD 11.0-RELEASE-p1	

表2 仮想計算機環境の諸元

		計算機 3	計算機 4
プロセッサ数 (コア数)		4	8
メモリ		8GB	
ディスク	ファイルシステム	UFS キャッシュは使用しない	
OS		FreeBSD 11.0-RELEASE-p1	
VMM		bhyve	
物理計算機環境		計算機 2	

いて、大サイズを 10,000 バイトとする。

3.2 評価プログラム

評価用プログラムの処理内容と起動条件は、文献[1]と同様にした。ディスク I/O 制御方式を評価するため、ある銀行システムのオンライン処理とバッチ処理を擬似した評価用プログラムを使用する。具体的には、オンライン処理は、外部からデータをオンラインで受信し、集計・加工処理を実行し、その結果を外部に送信する。このようなオンライン処理では、送受信するデータはディスク等に保存される。そこで、評価用プログラムのオンライン処理は、受信したデータをディスクに保存し、その後、データベース等を使用して集計・加工処理を実行後、処理結果データをディスクに保存し、処理結果データを送信する。また、オンライン処理は外部からの多数の要求を受け付けて処理を実行する。バッチ処理は、銀行におけるデータベース等に格納されているデータ等の集計・加工処理を実行する。バッチ処理は、大量データを順次処理するため、データベースよりも高速なアクセスが期待できるファイルを使用することが多い。そこで、評価用プログラムのバッチ処理は、データベースからデータを取り出し、集計・加工処理を実行し、処理結果データをファイルに書き出す。また、オンライン処理が実行される時間帯では、少数のバッチ処理が実行される。

図 2 に評価用プログラムの処理流れを示す。図 2 (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（オンライン処理）であり、図 2 (B) は、バッチ処理を擬似した処理（バッチ処理）である。具体的な処理内容を以下に述べる。

書き出しデータ長は、オンライン処理を 1,000 バイト、バッチ処理を 1,000,000 バイトとする。オンライン処理の処理流れは、オンライン処理の処理実行（初期処理）と処理実行（後処理）をそれぞれ 20 ミリ秒、処理実行（主処理）を 100 ミリ秒、DB 処理実行結果待ちを 50 ミリ秒とし、全体の処理時間を 190 ミリ秒とする。バッチ処理の処理流れは、処理実行（初期処理）を 50 ミリ秒、処理実行（主処理）を 50 ミリ秒、DB 処理実行結果待ちを 100 ミリ秒、処理実行（後処理）を 50 ミリ秒とし、全体の処理時間が 30 秒程度となるように、繰り返し回数を 150 回とする。

オンライン処理とバッチ処理の起動は、オンライン処理の負荷、および、バッチ処理数が多い場合と少ない場合の組み合わせとする。また、両処理の同時起動により、プロセッサ資源がボトルネックとなることを避けるため、最大の同時起動数は、CPU 使用率が 90%程度になるように決める。評価用プログラムの起動条件を以下に示す。

(1) オンライン処理

(a) オンライン処理の到着は、0～3 秒の間隔でランダムに到着するものとする。

(b) 同時起動数は 10, 20, 30, 40 とする。

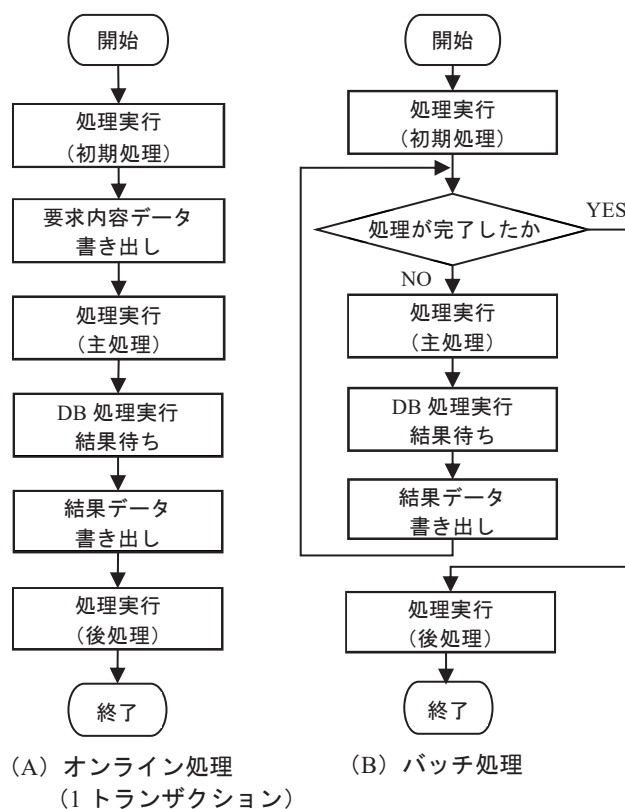


図2 評価用プログラムの処理流れ

(2) バッチ処理

(a) バッチ処理は、同時に起動したバッチ処理の完了後、次のバッチ処理を同時に起動する。

(b) バッチ処理の同時起動数は、多い場合として 4、少ない場合として 2 とする。

4. 結果と考察

4.1 処理時間の評価

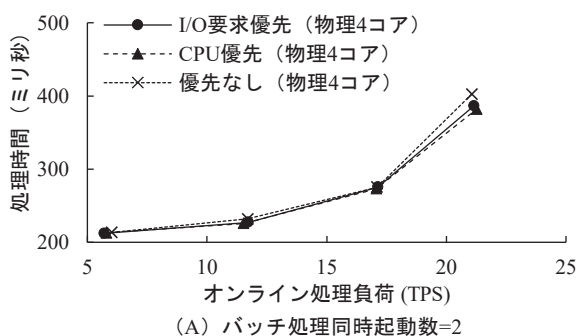
4.1.1 物理計算機における評価

(1) オンライン処理

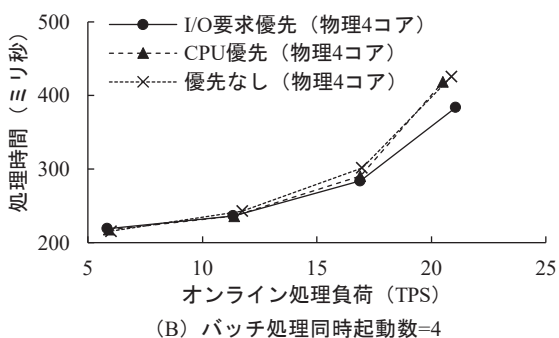
計算機 1 を使用したオンライン処理の処理時間を図 3 に示し、計算機 2 を使用したオンライン処理の処理時間を図 4 に示す。図 3 および図 4 より以下のことがわかる。

(a) プロセッサ数の少ない計算機では、バッチ処理の同時起動数とオンライン処理の負荷が多い時、つまり計算機の処理負荷が高い時、I/O 要求優先の効果は大きい。例えば、計算機 1 では、バッチ処理の同時起動数 4 の場合、オンライン処理の負荷量が 21.1TPS (I/O 要求優先) と 20.5TPS (CPU 優先) を比較すると、I/O 要求優先の処理時間は、CPU 優先より 9.1% (34.8 ミリ秒) 短い。一方、バッチ処理の同時起動数 2 の場合、オンライン処理の負荷量が 21.2TPS (I/O 要求優先) と 21.3TPS (CPU 優先) を比較すると、I/O 要求優先の処理時間は、CPU 優先より 1.1% (4.4 ミリ秒) 長い。

(b) プロセッサ数の多い計算機では、計算機の処理負荷が高い時、CPU 優先の効果は大きい。例えば、計算機 2 では、



(A) バッチ処理同時起動数=2



(B) バッチ処理同時起動数=4

図3 オンライン処理の処理時間 (計算機1)

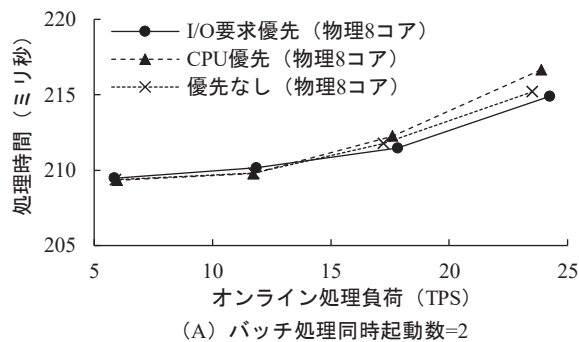
バッチ処理の同時起動数4の場合、オンライン処理の負荷量が23.2TPS (I/O 要求優先)と22.8TPS (CPU 優先)を比較すると、I/O 要求優先の処理時間は、CPU 優先より1.4% (3.1 ミリ秒) 長い。一方、バッチ処理の同時起動数2の場合、オンライン処理の負荷量が24.2TPS (I/O 要求優先)と23.9TPS (CPU 優先)を比較すると、I/O 要求優先の処理時間は、CPU 優先より0.8% (1.8 ミリ秒) 短い。

プロセッサ数の少ない計算機では、計算機の処理負荷が高い時、プロセッサ資源の競合が発生しやすい。また、バッチ処理のI/O 要求のような処理時間の長く、他の処理より優先して実行されるI/O 要求の割り込み処理が多数実行される時、オンライン処理はプロセッサ資源の競合の影響をより受けやすくなると考えられる。このような状況で、オンライン処理のI/O 要求がI/O 要求優先によって優先して実行されることで、プロセッサ資源がオンライン処理により多く割り当てられるようになったと考えられる。このため、I/O 要求優先の効果が大きくなったと考えられる。

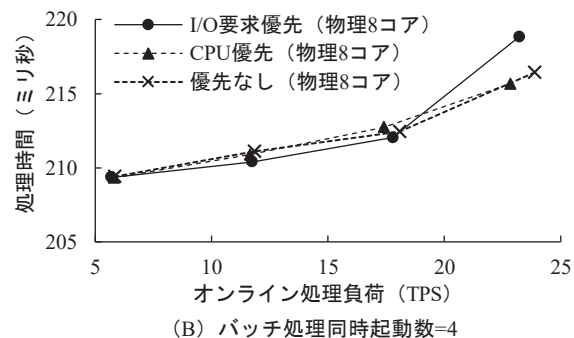
(2) バッチ処理

計算機1を使用したバッチ処理の処理時間を図5に示し、計算機2を使用したバッチ処理の処理時間を図6に示す。図5および図6より以下のことがわかる。

(a) I/O 要求優先のバッチ処理の処理時間は、バッチ処理の同時起動数およびオンライン処理の負荷量にかかわらず、CPU 優先より長い。また、I/O 要求優先による処理時間への影響は、プロセッサ数の少ない計算機の方が大きい。例えば、バッチ処理の同時起動数2の場合、計算機1では、オンライン処理の負荷量が21.2TPS (I/O 要求優先)と21.3TPS (CPU 優先)を比較すると、I/O 要求優先の処理時間は、CPU 優先より16.2% (13.1 秒) 長い。一方、計算機



(A) バッチ処理同時起動数=2



(B) バッチ処理同時起動数=4

図4 オンライン処理の処理時間 (計算機2)

2では、オンライン処理の負荷量が24.2TPS (I/O 要求優先)と23.9TPS (CPU 優先)を比較すると、I/O 要求優先の処理時間は、CPU 優先より7.0% (2.1 秒) 長い。さらにバッチ処理の同時起動数4の場合、計算機1では、オンライン処理の負荷量が21.2TPS (I/O 要求優先)と21.3TPS (CPU 優先)を比較すると、I/O 要求優先の処理時間は、CPU 優先より28.8% (30.7 秒) 長い。一方、計算機2では、オンライン処理の負荷量が24.2TPS (I/O 要求優先)と23.9TPS (CPU 優先)を比較すると、I/O 要求優先の処理時間は、CPU 優先より12.8% (4.1 秒) 長い。

(b) プロセッサ数の多い計算機では、I/O 要求優先のバッチ処理の処理時間は、バッチ処理の同時起動数にかかわらず、オンライン処理の負荷量に対してほぼ同じである。プロセッサ数の多い計算機は、プロセッサ数の少ない計算機よりプロセッサ資源の競合が発生しにくいいため、バッチ処理への影響は一定になると考えられる。

以上より、バッチ処理のI/O 要求の短期の時間的な負荷分散によって、バッチ処理の処理時間は長くなる。なお、バッチ処理は計画的に実行することにより、長期の時間的な負荷分散が可能であるため、I/O 要求優先による影響を軽減することが可能である。

4.1.2 仮想計算機における評価

(1) オンライン処理

図7に計算機3を使用したオンライン処理の処理時間を示し、図8に計算機4を使用したオンライン処理の処理時間を示す。図7および図8より以下のことがわかる。

(a) I/O 要求優先は、計算機3および計算機4のいずれの仮想計算機においても、バッチ処理の同時起動数およびオンライン処理の負荷量にかかわらず、オンライン処理の処

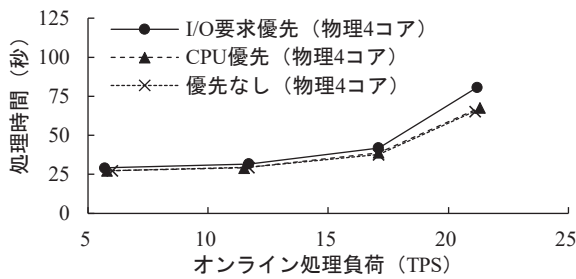
理時間を短くできる。

(b) 仮想計算機の方が、物理計算機より I/O 要求優先の効果は大きい。例えば、バッチ処理の同時起動数 4 の場合、図 3 (B) より、計算機 1 では、オンライン処理の負荷量が 21.1TPS (I/O 要求優先) と 20.5TPS (CPU 優先) を比較すると、I/O 要求優先の処理時間は、9.1% (34.8 ミリ秒) 短い。一方、計算機 3 では、オンライン処理の負荷量が 21.1TPS (I/O 要求優先) と 20.4TPS (CPU 優先) を比較すると、I/O 要求優先の処理時間は、CPU 優先より 13.3% (53.2 ミリ秒) 短く、I/O 要求優先の効果は大きい。

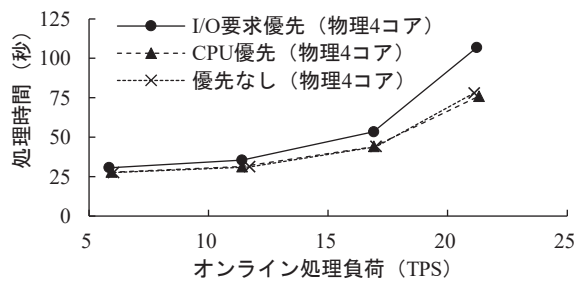
(c) I/O 要求優先は、プロセッサ数が少ない仮想計算機において、より効果が大きい。例えば、計算機 3 では、バツ

チ処理の同時起動数 4 のオンライン処理の負荷量が 21.1TPS (I/O 要求優先) と 20.4TPS (CPU 優先) のオンライン処理の処理時間は、CPU 優先より 13.3%短い。一方、計算機 4 では、バッチ処理の同時起動数 4 のオンライン処理の負荷量が 23.5TPS (I/O 要求優先) と 23.3TPS (CPU 優先) を比較すると、I/O 要求優先のオンライン処理の処理時間は、CPU 優先より 2.6% (5.9 ミリ秒) 短い。

(d) オンライン処理の負荷量が多くなるにしたがって、I/O 要求優先の処理時間と CPU 優先の処理時間の差は大きくなる。例えば、計算機 3 では、バッチ処理の同時起動数 4 の場合、オンライン処理の負荷量が 6.0TPS (I/O 要求優先) と 6.0TPS (CPU 優先) を比較すると、I/O 要求優先の処理

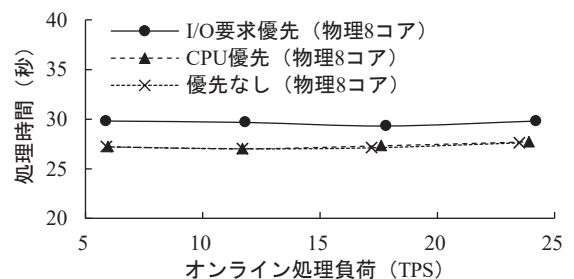


(A) バッチ処理の同時起動数=2

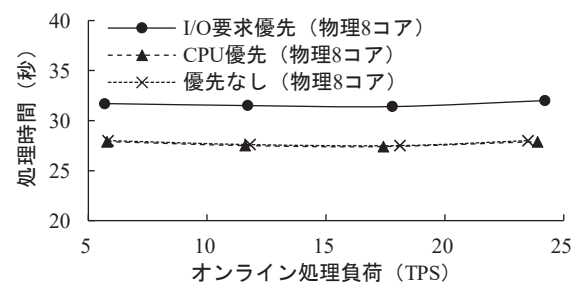


(B) バッチ処理の同時起動数=4

図 5 バッチ処理の処理時間 (計算機 1)

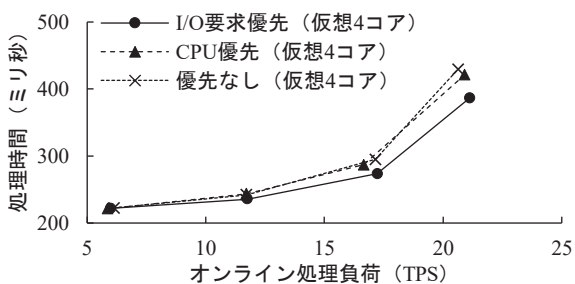


(A) バッチ処理の同時起動数=2

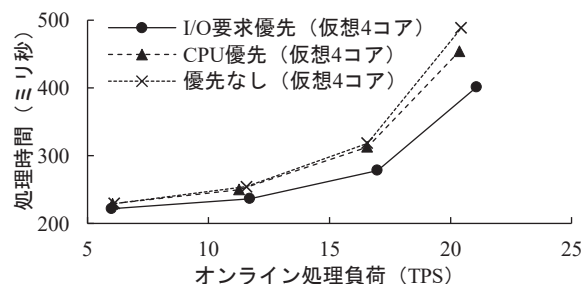


(B) バッチ処理の同時起動数=4

図 6 バッチ処理の処理時間 (計算機 2)

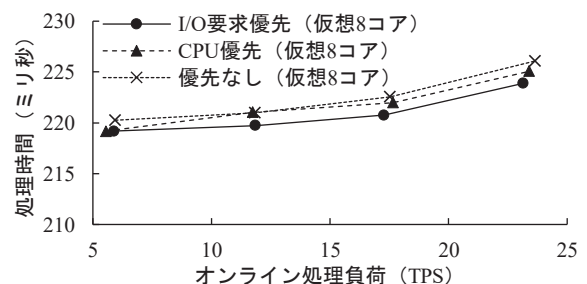


(A) バッチ処理の同時起動数=2

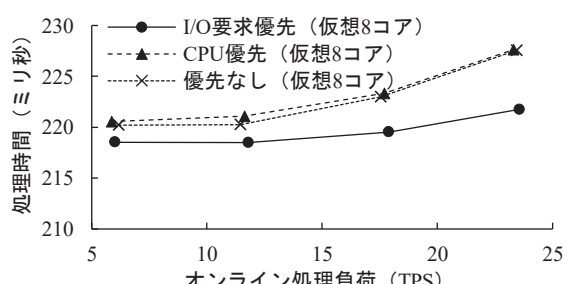


(B) バッチ処理の同時起動数=4

図 7 オンライン処理の処理時間 (計算機 3)



(A) バッチ処理の同時起動数=2



(B) バッチ処理の同時起動数=4

図 8 オンライン処理の処理時間 (計算機 4)

時間は、CPU 優先より 3.3% (7.4 ミリ秒) 短い。一方、オンライン処理の負荷量が 21.1TPS (I/O 要求優先) と 20.4TPS (CPU 優先) を比較すると、I/O 要求の処理時間は、CPU 優先より 13.3% (53.2 ミリ秒) 短くなり、処理時間の差は大きくなっている。つまり、I/O 要求優先の効果は、オンライン処理の負荷量の多い方が大きくなる。

(e) I/O 要求優先は、オンライン処理の負荷量が増加した場合、オンライン処理の処理時間の増加を抑えることができる。例えば、計算機 4 では、バッチ処理の同時起動数 4 の場合、オンライン処理の負荷量が 23.3TPS (CPU 優先) と 5.9TPS (CPU 優先) を比較すると、I/O 要求の処理時間の差は、7.0 ミリ秒である。一方、オンライン処理の負荷量が 23.5TPS (I/O 要求優先) と 6.0TPS (I/O 要求優先) を比較すると、処理時間の差は、3.2 ミリ秒であり、処理時間の増加が抑えられていることがわかる。

(2) バッチ処理

図 9 に計算機 3 を使用したバッチ処理の処理時間を示し、図 10 に計算機 4 を使用したバッチ処理の処理時間を示す。図 9 および図 10 より以下のことがわかる。

(a) I/O 要求優先のバッチ処理の処理時間は、バッチ処理の同時起動数およびオンライン処理の負荷量にかかわらず、CPU 優先より長い。また、I/O 要求優先による影響は、プロセッサ数の少ない計算機の方が大きい。例えば、I/O 要求優先と CPU 優先について、バッチ処理の同時起動数 2 のバッチ処理の処理時間を比較する。計算機 4 では、オンライン処理の負荷量が 23.1TPS (I/O 要求優先) と 23.4TPS (CPU 優先) を比較すると、I/O 要求の処理時間は、CPU 優先より 22.1% (9.4 秒) 長い。一方、計算機 3 では、オンライン処理の負荷量が 21.1TPS (I/O 要求優先) と 20.9TPS (CPU 優先) を比較すると、I/O 要求優先の処理時間は、

CPU 優先より 43.4% (61.3 秒) 長くなる。

(b) 計算機 3 では、I/O 要求優先の処理時間は、バッチ処理の同時起動数にかかわらず、オンライン処理の負荷量が多くなるにしたがって長い。一方、計算機 4 では、I/O 要求優先の処理時間は、バッチ処理の同時起動数にかかわらず、オンライン処理の負荷量に対してほぼ同じである。これは、計算機 1 および計算機 2 の物理計算機と同様である。

4.2 I/O 要求の処理時間の評価

計算機 1 を使用したオンライン処理の I/O 要求の処理時間を図 11 に示し、計算機 2 を使用したオンライン処理の I/O 要求の処理時間を図 12 に示す。図 11 および図 12 より以下のことがわかる。

(a) 計算機 1 では、バッチ処理の同時起動数にかかわらず、I/O 要求優先の I/O 要求の処理時間は、CPU 優先および優先なしより短く、I/O 要求優先の効果大きい。

(b) 計算機 2 では、処理負荷が最も小さい時と最も大きい時、I/O 要求優先の I/O 要求の処理時間は、CPU 優先より長い。また、この時以外では、I/O 要求優先の I/O 要求の処理時間は CPU 優先より短く、I/O 要求優先の効果大きい。

(c) 計算機 2 では、CPU 優先のオンライン処理の I/O 要求の処理時間は優先なしより長くなることがある。例えば、バッチ処理の同時起動数 2 の場合、オンライン処理の同時起動数にかかわらず、CPU 優先の I/O 要求の処理時間は優先なしより長い。

プロセッサ数の多い計算機では、プロセッサ資源の競合が発生しにくいいため、I/O 要求の処理時間は短くなるが、I/O 要求の処理時間の短い I/O 要求は、バッチ処理の I/O 要求の影響を受けやすくなる。このため、I/O 要求優先によってオンライン処理の I/O 要求を優先的に実行した場合や CPU 優先によりオンライン処理を優先的に実行した場合、

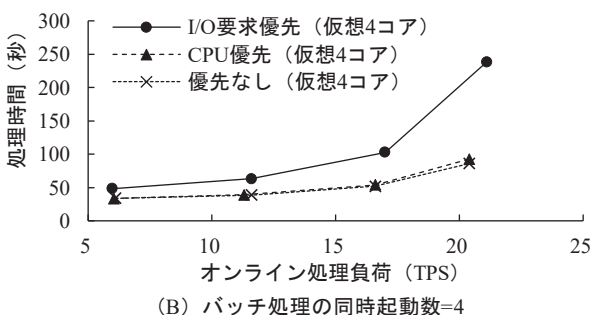
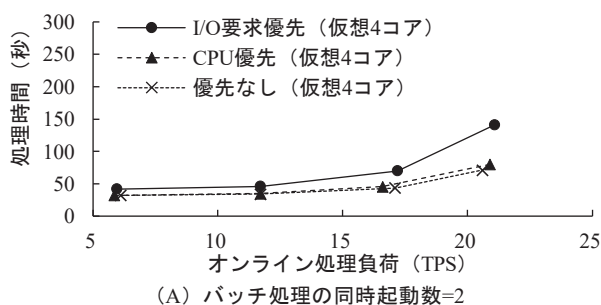


図 9 バッチ処理の処理時間 (計算機 3)

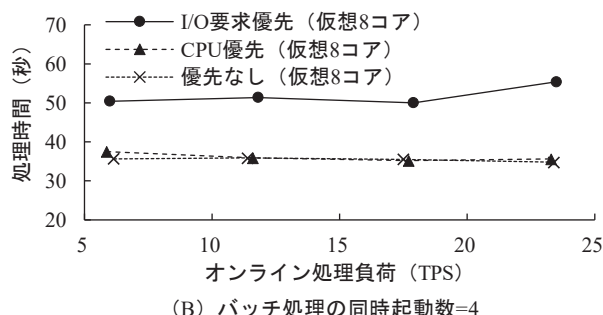
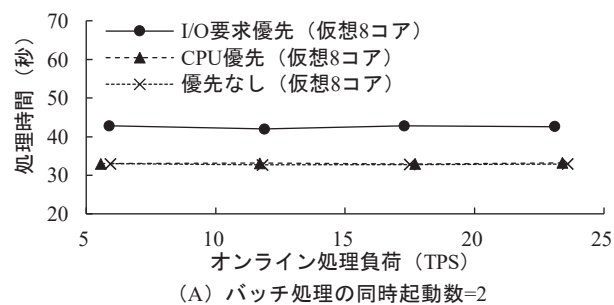


図 10 バッチ処理の処理時間 (計算機 4)

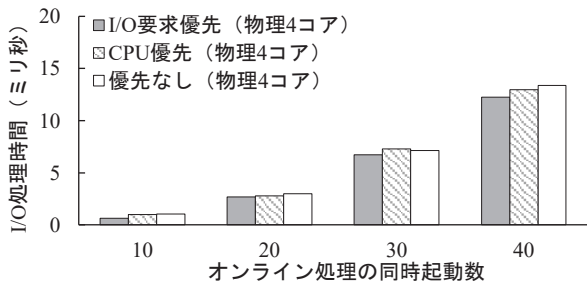
I/O 要求の処理時間が長くなることがあると考えられる。

次に、計算機 3 を使用したオンライン処理の I/O 要求の処理時間を図 13 に示し、計算機 4 を使用したオンライン処理の I/O 要求の処理時間を図 14 に示す。図 13 と図 14 より以下のことがわかる。

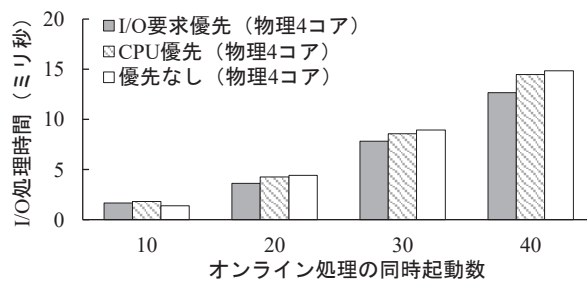
(d) 計算機 3 および計算機 4 のいずれの計算機においてもバッチ処理の同時起動数およびオンライン処理の負荷量にかかわらず、I/O 要求優先の I/O 要求の処理時間は、CPU 優先より短い。つまり、仮想計算機では、I/O 要求優先は、I/O 要求の処理時間を CPU 優先に比べて短くでき、I/O 要求の短期の時間的な負荷分散による入出力制御は有効であ

る。さらに、I/O 要求優先の効果は、仮想計算機が物理計算機より大きい。これは、VMM のプリエンプション禁止区間が物理計算機より長いことが考えられる。バッチ処理の処理時間の長い I/O 要求の実行後に、並行してオンライン処理の I/O 要求が実行された場合、オンライン処理の I/O 要求の実行開始までの待ち時間は長くなる。I/O 要求の処理時間が短い場合、I/O 要求の実行開始までの待ち時間の増加の影響が大きいため、仮想計算機の効果が大きくなる。

(e) 計算機 4 では、CPU 優先の I/O 要求の処理時間は、バッチ処理の同時起動数 4 の場合、オンライン処理の同時起動数にかかわらず、優先なしより長い。

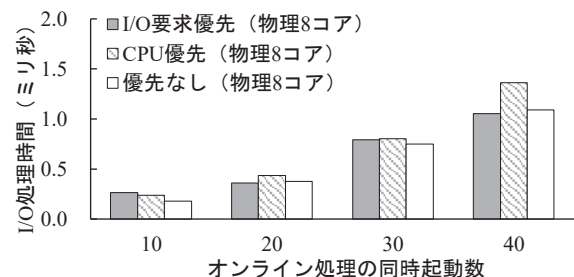


(A) バッチ処理の同時起動数=2

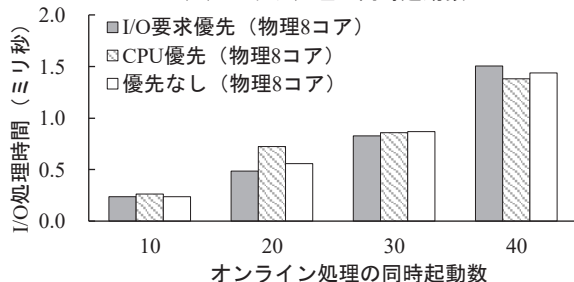


(B) バッチ処理の同時起動数=4

図 11 オンライン処理の I/O 要求の処理時間 (計算機 1)

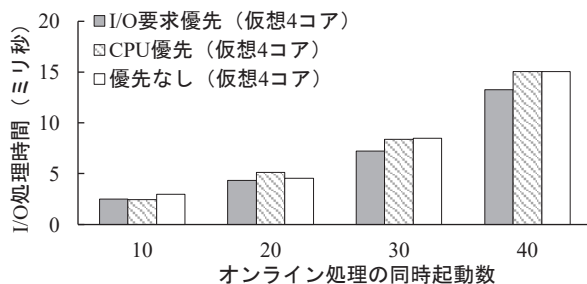


(A) バッチ処理の同時起動数=2

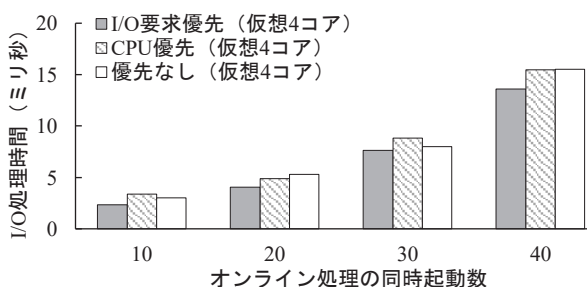


(B) バッチ処理の同時起動数=4

図 12 オンライン処理の I/O 要求の処理時間 (計算機 2)

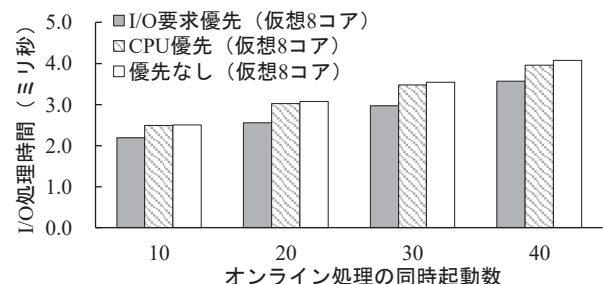


(A) バッチ処理の同時起動数=2

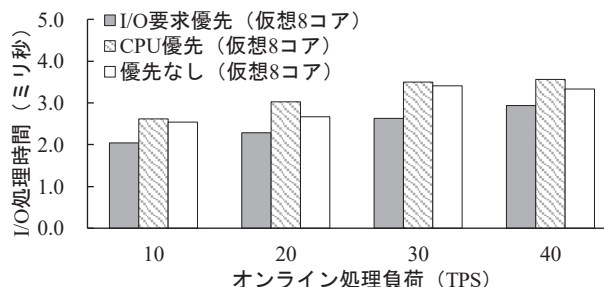


(B) バッチ処理の同時起動数=4

図 13 オンライン処理の I/O 要求の処理時間 (計算機 3)



(A) バッチ処理の同時起動数=2



(B) バッチ処理の同時起動数=4

図 14 オンライン処理の I/O 要求の処理時間 (計算機 4)

5. 関連研究

文献[2][3][4][5]は、I/O 要求に対するシーク時間や回転待ち時間により、I/O 要求の処理時間が大きくなることに着目し、書き込みデータのディスク上の書き込み位置などの情報と I/O 要求の処理時間を予測し、I/O 要求を効率的に実行するスケジューリング手法を提案している。文献[6]は、入出力スループットの向上を目的として、入出力要求を並び替えて HDD のシーク時間を最小とする I/O 要求のスケジューリング手法を提案している。文献[7]は、アプリケーションの I/O 要求の実行にあたって、I/O 要求の優先度、I/O 要求の開始時間と終了時間をもとにして、優先的に実行する I/O 要求を決定し、それらの I/O 要求を複数まとめ、スループットの向上するために I/O 要求の実行順番を入れ替える方式を提案している。この方法では、オンライン処理の I/O 要求をまとめて実行することになり、オンライン処理の応答時間の最大値が大きくなり好ましくない。文献[8][9]は、I/O 要求の処理時間を調整する制御法を提案している。この制御法は、高い優先度を持つ I/O 要求が優先的に実行するために低い優先度の I/O 要求の処理時間を調整し、高い優先度を持つ I/O 要求を優先的に実行する。しかし、オンライン処理の負荷量やバッチ処理の起動数が多い業務システムでは、同時に多数の I/O 要求が実行されるため、I/O 要求の処理時間を調整することが難しくなる。

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[10][11][12]。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。

6. おわりに

オンライン処理とバッチ処理が実行される環境において、物理計算機と仮想計算機におけるディスク I/O 制御方式の有効性を評価した。プロセッサ数が少ない物理計算機では、ディスク I/O 制御方式は、バッチ処理の同時起動数が多く、オンライン処理の処理負荷量が多い時、プロセス優先度によるプロセッサ割り当て制御法に比べて、オンライン処理の処理時間を短くできることを示した。また、プロセッサ数が多い物理計算機では、ディスク I/O 制御方式とプロセス優先度によるプロセッサ割り当て制御法のオンライン処理の処理時間は、ほぼ同じであることを示した。一方、仮想計算機では、バッチ処理の同時起動数およびオンライン処理の負荷量にかかわらず、ディスク I/O 制御方式は、プロセス優先度によるプロセッサ割り当て制御法に比べて、オンライン処理の処理時間を短くできることを示

し、ディスク I/O 制御方式によって I/O 要求の短期の時間的な負荷分散を行う入出力制御は有効であることを示した。

I/O 要求の処理時間について、物理計算機および仮想計算機のいずれの計算機においても、バッチ処理の同時起動数およびオンライン処理の負荷量にかかわらず、ディスク I/O 制御方式は、プロセス優先度によるプロセッサ割り当て制御法に比べて、オンライン処理の I/O 要求の処理時間を短くできることを示した。また、プロセッサ数が多い仮想計算機では、I/O 要求の処理時間は、バッチ処理の同時起動数が多い場合、プロセス優先度によるプロセッサ割り当て制御法では、優先制御をしない場合に比べて、悪化することを示した。ディスク I/O 制御方式による入出力制御は、I/O 要求の処理時間に対しても有効であることを示した。

参考文献

- 1) 田辺雅則, 横山和俊, 長尾尚, 谷口秀夫, “オンライン処理とバッチ処理のファイル I/O の特徴を生かして処理負荷を分散制御する入出力制御法,” 情報処理学会研究報告マルチメディア通信と分散処理, 2019-DPS-178, No.12, 2019/02/25.
- 2) Lan Huang and Tzi-cker Chiueh, “Implementation of a Rotation-Latency-Sensitive Disk Scheduler,” Technical Report ECSL-TR81, SUNY, Stony Brook University at New York, Mar. 2000.
- 3) Sitaram Iyer and Peter Drushel, “Anticipatory scheduling: A disk scheduling framework to overcome deceptive idleness in synchronous I/O,” Proc. of 18th ACM Symposium on Operating Systems Principles, Oct. 2001.
- 4) Paolo Valente and Fabio Checconi, “High Throughput Disk Scheduling with Fair Bandwidth Distribution,” IEEE Transactions on Computers, Vol. 59, No. 9, pp.1172-1186, Sep. 2010.
- 5) 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.
- 6) Y. Son, H. Y. Yeom and H. Han, “Optimizing I/O Operations in File Systems for Fast Storage Devices,” IEEE Transactions on Computers, Vol.66, No.6, pp.1071-1084, June 2017.
- 7) J. Bruno, J. Brustoloni, E. Gabber, M. Mcshea, B. Ozden and A. Silberschatz, “Disk Scheduling with Quality of Service Guarantees,” Proceedings IEEE International Conference on Multimedia Computing and Systems, 7-11 June 1999.
- 8) 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.
- 9) 長尾尚, 田辺雅則, 横山和俊, 谷口秀夫, “入出力スループットの低下を抑制する入出力性能の調整法の実現と評価,” 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. 2019-OS-145, No.2, 2019/02/21.
- 10) 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.
- 11) 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.
- 12) 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.