

ファイルの使用頻度に基づくバッファキャッシュ制御法の提案

片上達也† 田端利宏‡ 谷口秀夫‡

†岡山大学工学部 ‡岡山大学大学院自然科学研究科

1. はじめに

バッファキャッシュ制御法はブロックのアクセスに着目したものが多く、そこで、我々は、ファイル情報に基づく方法として、ファイルの格納ディレクトリを考慮したバッファキャッシュ制御法を提案している[1].

ここでは、応用プログラム（以降、AP）が操作したファイルの情報（以降、ファイル操作情報）を収集し、ファイルの使用頻度に基づき、優先して保持するブロックを決定し制御するバッファキャッシュ制御法を提案する。

2. ファイルの使用頻度に基づく制御法

2.1 基本方式

APはファイル毎に入出力を行う。これに着目し、再びアクセスされる可能性の高いファイルを構成するブロックを優先的にキャッシュする方式を提案する。この方式を図1に示し、以下に説明する。

- (1) バッファキャッシュを保護領域と通常領域に分割する。
- (2) ファイルアクセスが発生すると、アクセスしたファイルを構成するブロックを保護領域で保護するか決定するため、ファイル操作情報をログ情報表に収集する。
- (3) 収集した情報を基に重要度判定表を更新し、ファイル毎の重要度を計算する。
- (4) 新規ブロック読み込みが発生した場合、重要度を基に、ブロックのファイルが重要であるか判定し、重要であれば保護領域へ、重要でなければ通常領域へ格納する。
- (5) バッファキャッシュのブロック置き換え時には、通常領域からLRUで置き換え対象のブロックを選択する。通常領域にブロックがなければ、保護領域内で最も重要度の低いファイルを構成するブロックを全て通常領域に移し、通常領域からLRUで置き換え対象のブロックを選択する。

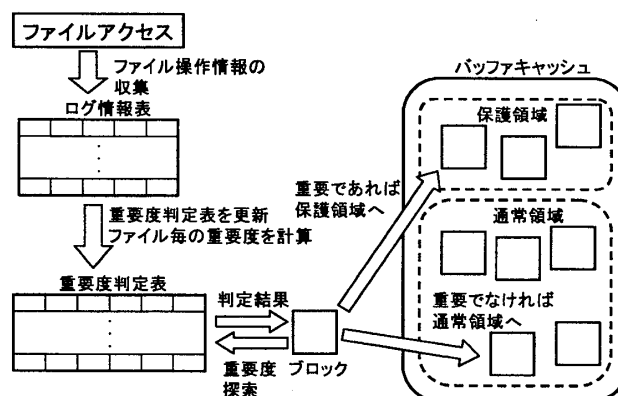


図1 基本方式

2.2 重要度の考え方

重要度とは、アクセスしたファイルを構成するブロックを保護領域で保護するか決定するために使用する値である。重要度は、過去にファイルアクセスを行った情報を基に、これからのファイルアクセスを予測し、計算したものである。

重要度には、以下の2つの目標がある。

- (1) 再利用される可能性の高いファイルほど重要度が高くなること
- (2) サイズの大きいファイルを保護することによって発生するバッファキャッシュの占有を防ぐために、サイズの高いファイルほど重要度が低くなること

2.3 課題

提案方式には、以下の3つの課題が存在する。

- (1) 重要度の計算方針
- (2) 収集するファイル操作情報
- (3) 重要度判定表の更新方法

以上の課題に対して、2.4節にて対処を説明する。

2.4 対処

2.4.1 重要度の計算方針

2.2節で述べた重要度の目標を達成するために、重要度の計算にはファイル毎に以下の情報を利用する。

- (1) 参照数
- (2) オープン回数
- (3) ファイルサイズ

参照数とは、現在ファイルをオープンしているプロセスの数である。オープン回数とは、ファイルをオープンするシステムコールが発行さ

Proposal of Buffer Cache Management Based on Use Frequency of File

Tatsuya Katakami†, Toshihiro Tabata‡ and Hideo Taniguchi‡

†Faculty of Engineering, Okayama University

‡Graduate School of Natural Science and Technology, Okayama University

れた回数である。ファイルサイズとは、アクセスしたファイルの大きさである。

2.4.2 収集するファイル操作情報

ファイルを特定するために、iノード番号を利用する。また、2つのファイルの重要度が等しい場合、それぞれのファイルを区別するために、最新オープン時刻を利用する。

提案方式では、ファイル操作情報として、重要度計算に必要な情報、iノード番号、及び最新オープン時刻を用いる。

2.4.3 重要度判定表の更新方法

ログ情報表のファイル操作情報を重要度判定表に反映させるために、判定表更新契機の度に重要度判定表を更新する。判定表更新契機は、間隔が短いと重要度判定表は頻りに更新され、間隔が長くと重要度判定表はあまり更新されない。判定表更新契機は、ファイルアクセスの状況にあわせて変動するのが好ましい。このため、前回の重要度判定表の更新から一定回数のファイルアクセスが行われたときを判定表更新契機とする。

しかし、ログ情報表の大きさを固定とした場合、一定回数のアクセスが行われる前に、ログ情報表があふれることがある。よって、判定表更新契機として、さらに、ログ情報表に一定数のファイル操作情報が蓄えられたときを採用する。

上記の判定表更新契機の度に、ログ情報表にあるファイル操作情報を重要度判定表に反映させて、重要度判定表の更新を行う。このとき、古い情報がいつまでも有効であることを防ぐために、古い情報の影響を小さくする必要がある。このため、重要度判定表の更新時に、重み関数を用いた重み付け手法を採用する。

2.5 期待される効果

提案方式の利用によって、期待される効果を以下に述べる。

- (1) 重要度の計算を繰り返すことにより、キャッシュヒット率が向上する。
- (2) ファイルサイズを意識することにより、バッファキャッシュよりも大きなサイズのファイルが読み込まれた場合のバッファキャッシュ占有を阻止し、重要なファイルを構成するブロックを保護する。

3. 実現方式

3.1 処理の流れ

提案方式の処理の流れを図2に示し、以下に説明する。

openまたはcloseシステムコールが発行される

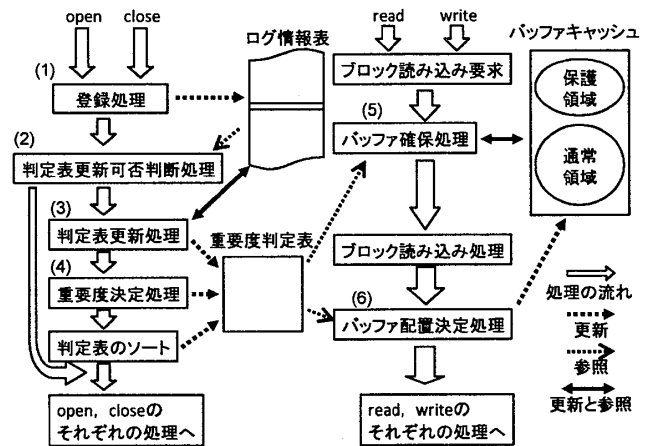


図2 処理の流れ

と、(1)登録処理によって、ログ情報表にファイル操作情報を収集する。次に、(2)判定表更新可否判断処理を行い、判定表の更新条件を満たさない場合は、openまたはcloseシステムコールのそれぞれの処理に戻る。判定表の更新条件を満たす場合には、(3)判定表更新処理によって、重要度判定表を更新する。判定表の更新が終了したら、(4)重要度決定処理によって重要度を計算し、openまたはcloseシステムコールのそれぞれの処理に戻る。

readまたはwriteシステムコールが発行され、ブロック読み込み要求が出されたときに、(5)バッファ確保処理によって、通常領域からLRUで置き換え対象のブロックを選択し、バッファの確保を行う。通常領域にブロックがなければ、保護領域内で最も重要度の低いファイルを構成するブロックを全て通常領域に移し、通常領域からLRUで置き換え対象のブロックを選択して、バッファの確保を行う。

ブロック読み込み処理が終了すると、(6)バッファ配置決定処理によって、重要度を基に、保護領域に配置するか、通常領域に配置するかを決定し、readまたはwriteシステムコールのそれぞれの処理に戻る。

4. おわりに

ファイルの使用頻度に基づくバッファキャッシュ制御法を提案し、実現方式を述べた。残された課題として、APを用いた評価がある。

謝辞 本研究の一部は、科学研究費補助金 若手研究(B) (課題番号 18700030) による。

参考文献

- [1] 田端 利宏, 小峠 みゆき, 乃村 能成, 谷口 秀夫, “ファイルの格納ディレクトリを考慮したバッファキャッシュ制御法の実現と評価,” 電子情報通信学会論文誌, Vol. J91-D, No. 2, Feb. 2008. (掲載予定)