

ファイル読み込み処理における *AnT* オペレーティングシステムの OS サーバの優先度逆転抑制法の評価

Evaluation of Reduction Method for Priority Inversion of OS server
for *AnT* in File Reading

鴨生悠冬[†]

Yuuto Kamou

山内利宏[†]

Toshihiro Yamauchi

谷口秀夫[†]

Hideo Taniguchi

1. はじめに

プロセス優先度に基づくスケジュールには、優先度逆転の発生により優先サービスの処理が遅れてしまう問題がある。特に、マイクロカーネル OS では、OS 機能を OS サーバとしてプロセスで実現するため、OS 処理にも影響を与えてしまう [1]。我々は、*AnT* オペレーティングシステム (以降、*AnT*) における OS サーバ間通信での優先度継承による優先度逆転の抑制法を提案した [2]。本稿では、ファイル読み込み処理における提案手法の抑制効果を評価した結果を報告する。

2. 優先度逆転抑制法

2.1 優先度逆転の問題と対処

マイクロカーネル OS では、OS 機能を OS サーバとして実現する。このため、AP プロセスと OS サーバ間の通信において、以下の優先度逆転問題が発生する。
(問題 1) OS サーバの依頼キューに先に繋がれた低優先度 AP プロセスの依頼により、後に繋がれた高優先度 AP プロセスの依頼の OS サーバ実行が遅延する。
(問題 2) 低優先度 AP プロセスの依頼を OS サーバが処理する間、高優先度の AP プロセスの実行開始が遅延する。

上記の問題は、処理依頼や処理結果の取得を FIFO 順に行うこと、および OS サーバの優先度が AP プロセスの優先度より常に高いことに起因する。そこで、以下の対処を行う。

- (対処 1) 依頼元 AP プロセスの優先度に基づく取得
- (対処 2) 依頼元 AP プロセスの優先度を OS サーバに継承

2.2 抑制法

2.1 節で述べた対処を行わない制御法も含め、優先度逆転を抑制する制御法を表 1 に示し、以下に説明する。なお、ICA 優先度は、OS サーバ間通信時に授受する依頼情報に格納した依頼元 AP プロセスの優先度である。

- (1) 固定優先度 FIFO 法 (FPFIFO) は、OS サーバの優先度を AP プロセスの優先度よりも高く設定し、依頼取得を FIFO 順に行う既存手法である。
- (2) 固定優先度 ICA 優先度順法 (FPICA) は、依頼取得を ICA 優先度順に行う制御法である。
- (3) 可変優先度取得時変更法 (VPGET) は、依頼取得を ICA 優先度順に行うとともに、依頼取得時に OS サーバの優先度を ICA 優先度に変更する制御法である。
- (4) 可変優先度登録時変更法 (VPSET) は、依頼取得

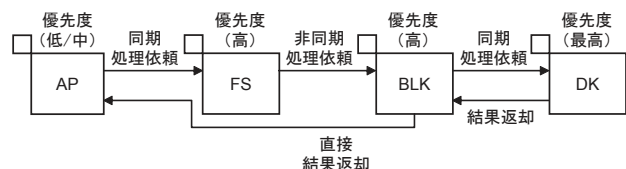


図 1 *AnT* におけるファイル複写の様子
(優先度は初期状態)

を ICA 優先度順に行うとともに、依頼登録時と依頼取得時に OS サーバの優先度を ICA 優先度に変更する制御法である。

3. 評価

3.1 測定内容

ファイル読み込み処理を例として、提案手法の優先度逆転の抑制効果を評価する。具体的には、高優先度 AP プロセス (AP_h) と n 個の低優先度 AP プロセス (AP_l) が同時にファイル複写する際の AP_h の処理時間を測定した。複写するファイルの大きさは 128KB であり、各プロセスは 4KB 単位の read&write を繰り返す。計算機は、Intel® Core i7 搭載計算機を用い、RDTSC 命令を使用して実測し、処理時間は 5 回の平均である。なお、1 つの AP プロセスのみを動作させた場合の処理時間 (基本処理時間) は、約 200 ミリ秒であった。

AnT におけるファイル複写の様子を図 1 に示す。AP は、ファイルサーバ (FS) に同期処理依頼を行う。FS は、ブロックサーバ (BLK) に非同期処理依頼を行う。BLK は、ディスクドライバ (DK) に同期処理依頼を行う。DK は、BLK に結果返却する。BLK は、FS を介さず、AP に直接結果返却する。

なお、制御法の効果を明らかにするため、ディスクの読み書きに DMA を使用せず、読み込みに PIO (割込み有)、書き込みに PIO (割込み無) を使用した。

3.2 結果と考察

高優先度 AP プロセスの処理時間を図 2 に示す。図 2 より、以下のことが分かる。

- (1) AP_l が 1 個の場合、全制御法の AP_h 処理時間は、同一である。これは、 AP_h と AP_l が交互にファイル複写を行い、OS サーバへの処理依頼時に、依頼キューに処理依頼が 1 つも繋がっていないためである。したがって、優先度逆転は発生しない。
- (2) AP_l が 2 個の場合、VPGET と VPSET の AP_h 処理時間は、FPFIFO と FPICA より短い。FPFIFO と FPICA では、OS サーバの優先度を常に AP の優先度

[†] 岡山大学大学院自然科学研究科, Graduate School of Natural Science and Technology, Okayama University

表1 優先度逆転を抑制する制御法の概要

通番	制御法	依頼取得順	OSサーバの優先度変更	
			依頼登録時	依頼取得時
(1)	固定優先度 FIFO (FPFIFO)	FIFO 順	なし	なし
(2)	固定優先度 ICA 優先度順 (FPICA)	ICA 優先度順	なし	なし
(3)	可変優先度取得時変更 (VPGET)	ICA 優先度順	なし	ICA 優先度に変更
(4)	可変優先度登録時変更 (VPSET)	ICA 優先度順	ICA 優先度より OS サーバの優先度が低い場合, ICA 優先度に変更	ICA 優先度より OS サーバの優先度が高い場合, ICA 優先度に変更

より高くする。このため、BLK が AP_h に直接結果返却した後、BLK が AP_l の write 依頼を取得し DK に依頼すると、DK が結果を BLK に返却し、BLK が AP_l に直接結果返却するまで、 AP_h は、結果を取得できない。一方、VPGET と VPSET では、OS サーバの優先度を ICA 優先度に変更する。このため、BLK が AP_h に直接結果返却した後、BLK が AP_l の write 依頼を取得する時に、BLK の優先度は AP_l の優先度に変更され、 AP_h に実行権が渡され、 AP_h は結果を取得でき、次の依頼を発行できる。

(3) FPFIFO の AP_h 処理時間は、 AP_l 数に比例し、1 増加するたびに約 200 ミリ秒（基本処理時間と同等）増加する。したがって、 AP_h 処理時間は、全ての AP_l 処理時間の影響を受ける。ここで、FPFIFO における優先度逆転回数を表 2 に示す。 AP_l 数にほぼ比例して優先度逆転回数が増加しており、上記を裏付けている。

(4) FPICA の AP_h 処理時間は、ほぼ AP_l 数に比例するが、増加率は FPFIFO より低い。これは、FPICA が ICA 優先度順に依頼を取得すること AP_h に起因している。OS サーバの依頼キューの末尾に AP_h の依頼が存在する場合を考える。FPFIFO では、先に繋がれた AP_l の依頼が OS サーバ実行されるまで、 AP_h の依頼の OS サーバ実行が遅延する。一方、FPICA では、始めに AP_h の依頼の OS サーバ実行される。したがって、FPICA の AP_h 処理時間は、一部の AP_l の依頼の影響のみを受ける。

(5) VPGET や VPSET の AP_h 処理時間は AP_l 数に比例せず、 AP_l が 4 個以上では、増加しない。この場合、 AP_h 処理時間は、 AP_h の依頼前に DK が取得した AP_l の依頼の影響を受ける。このことから、 AP_h の依頼前に DK が AP_l の依頼を取得している確率は、 AP_l が 4 個まで増加し、5 個以上では変化しないと推察する。すなわち、VPGET と VPSET が AP_l から受ける影響は、ある一定の AP_l の依頼であり、 AP_l の処理内容が同じであれば、同時に走行する AP_l の個数が際限なく増えた場合でも AP_h の処理時間は一定以上増えないといえる。

(6) VPSET の AP_h 処理時間は、VPGET に比べ、常に短い。これは、VPSET が依頼登録時に ICA 優先度を継承するため、優先度逆転の発生を最小限に抑制できることに起因する。

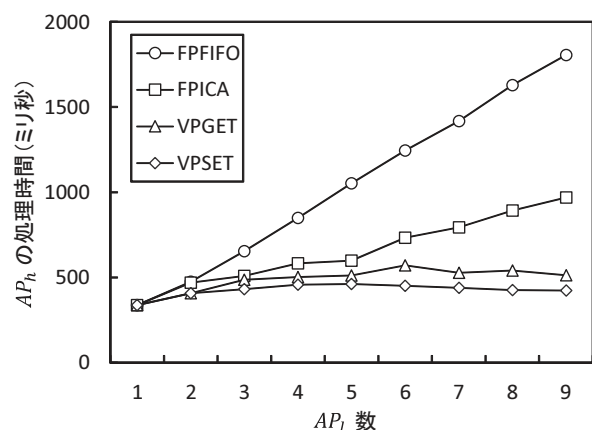


図2 高優先度 AP プロセスの処理時間

表2 FPFIFO における優先度逆転回数

AP _l 数	1	2	3	4	5	6	7	8	9
回数	0	12	42	75	108	143	170	214	242

4. おわりに

AnT における OS サーバの優先度逆転抑制法の評価として、高優先度 AP プロセス AP_h と n 個の低優先度 AP プロセス AP_l が同時にファイル複写する際の高優先度 AP プロセスの処理時間を測定し、評価した。FPFIFO と FPICA の AP_h 処理時間は、 AP_l 数に比例し、増加した。一方、VPGET や VPSET の AP_h 処理時間は、 AP_l 数に比例せず、 AP_l が 4 個以上では増加しなかった。また、抑制効果は $FPFIFO < FPICA < VPGET < VPSET$ であり、提案手法が優先度逆転を抑制できることを確認した。残された課題として、提案手法のマルチコアプロセッサ上で動作する OS への適用がある。

参考文献

- [1] Kitayama, T., Nakajima, T., and Tokuda, H.: RT-IPC: An IPC Extension for Real-Time Mach, USENIX Microkernels and Other Kernel Architectures Symposium, vol.39, no.9, pp.91-104 (1993).
- [2] 鴨生 悠冬, 山内 利宏, 谷口 秀夫: *AnT* オペレーティングシステムにおける OS サーバ間通信での優先度継承による優先度逆転の抑制法, 情報処理学会研究報告, vol.2015-OS-133, no.15, pp.1-8 (2015).