

Antにおいて効率的な負荷分散を可能にする一括処理機能

弓削 隼一† 佐藤 将也‡ 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

マイクロカーネル構造 OS では、OS 機能の一部をプロセス (OS サーバ) として実現する。このため、マルチコア環境において、OS サーバを各コアに分散配置することで OS 処理を分散できる。マイクロカーネル構造 OS である *Ant* オペレーティングシステムには、一括処理依頼機能^[1]と要求箱通信機能^[2]がある。一括処理依頼機能は、同種または異種の複数の処理を一度に OS サーバへ依頼する機能である。要求箱通信機能は、同種の 1 つの処理を要求箱に登録し、各 OS サーバが要求箱から処理を取得する機能である。本稿では、両機能の長所を併せもつことで効率的な負荷分散を可能にする一括処理機能について述べる。

2 Antのサーバプログラム間通信機構

2.1 基本機構

サーバプログラム間通信の基本機構を図 1 に示す。サーバプログラム間通信は、コア間通信データ域 (ICA: Inter-core Communication Area) を利用してプロセス間のデータ授受を複製レスで行う。通信制御に利用する ICA (制御用 ICA) には、OS サーバに渡す引数や通信制御の情報 (依頼情報) を格納する。また、カーネルは、全てのプロセスに、通信制御のための依頼キューと結果キューを用意する。

以下に、サーバプログラム間通信の処理流れを示す。

- (1) 依頼元プロセスが処理依頼を行うと、カーネルは、依頼先プロセスの依頼キューに依頼情報を格納した制御用 ICA を登録する。
- (2) 依頼先プロセスは、依頼キューから依頼情報を格納した制御用 ICA を取得し、依頼された処理を実行する。
- (3) 依頼先プロセスが結果返却を行うと、カーネルは、依頼元プロセスの結果キューに処理の結果 (結果情報) を格納した制御用 ICA を登録する。
- (4) 依頼元プロセスは、結果キューから結果情報を格納した制御用 ICA を取得し、処理を終了する。

2.2 一括処理依頼機能

一括処理依頼機能の様子を図 2 に示す。一括処理依頼機能は、応用プログラム (AP) プロセスが同種または異種の複数の処理を一括して OS サーバへ依頼する

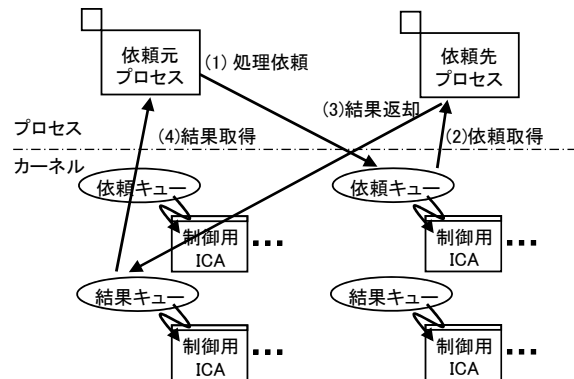


図 1 サーバプログラム間通信の基本機構

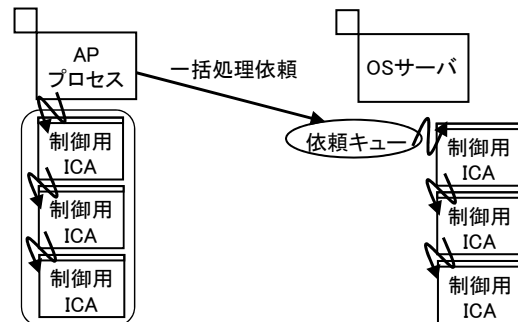


図 2 一括処理依頼機能

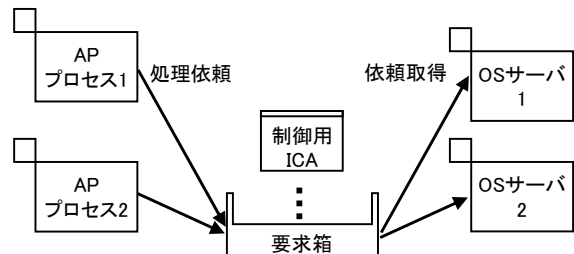


図 3 要求箱通信機能

機能である。また、AP プロセスは、一括処理依頼した全ての処理の結果を一括して取得する。

2.3 要求箱通信機能

要求箱通信機能の様子を図 3 に示す。要求箱通信機能は、AP プロセスが同種の 1 つの処理を依頼して要求箱に登録し、各 OS サーバが要求箱から依頼を取得する機能である。要求箱は処理種別ごとに生成される。このため、AP プロセスは、処理種別ごとに依頼を登録する要求箱を選択する。

2.4 一括処理依頼機能と要求箱通信機能の比較

一括処理依頼機能と要求箱通信機能の比較を表 1 に示す。一括処理依頼機能は、複数の処理依頼や結果取得を一括して行うため、処理依頼や結果取得に伴う通信オーバーヘッドを削減できる。しかし、AP プロセスは、

Batch Processing for Efficient Load Balancing in *Ant*.
 Hayato Yuge†, Masaya Sato‡, Hideo Taniguchi‡
 †Faculty of Engineering, Okayama University
 ‡Graduate School of Natural Science and Technology, Okayama University

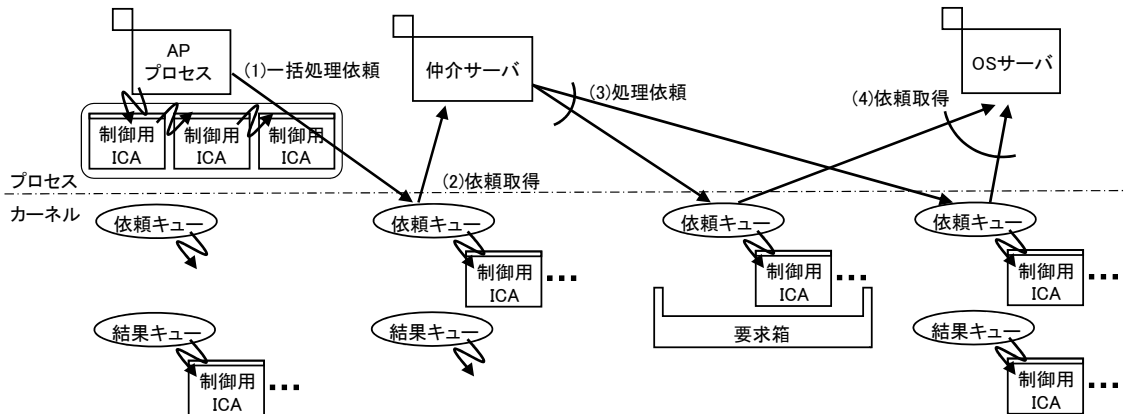


図4 一括処理機能

表1 一括処理依頼機能と要求箱通信機能の比較

	一括処理 依頼機能	要求箱 通信機能
通信オーバーヘッドの削減	○	×
負荷分散	×	○
即応性	○	×

依頼先 OS サーバを指定する必要があるため、負荷分散が難しい。

要求箱通信機能は、複数の OS サーバによる依頼取得が可能になるため、OS 処理の負荷を分散できる。しかし、要求箱通信機能は、通信オーバーヘッドを削減できない。また、要求箱に依頼が登録されていない場合、OS サーバは WAIT 状態となる。つまり、OS サーバによる依頼取得は、最大でポーリング周期時間だけ遅れるため、即応性を保証できない。

3 一括処理機能

3.1 問題と対処

一括処理依頼機能と要求箱通信機能の長所を併せもつ機能として一括処理機能を述べる。表1の問題点への対処として、仲介サーバを導入する。

仲介サーバは、一括処理依頼機能と同じインタフェースで依頼を受け取り、OS サーバまたは依頼の処理種別に対応する要求箱へ処理依頼を行う。これにより、通信オーバーヘッドの削減と負荷分散を可能にする。さらに、仲介サーバは、WAIT 状態である OS サーバを探索し、WAIT 状態の OS サーバがある場合は、その OS サーバへ処理依頼を行うことで即応性を保証する。

3.2 基本構造

一括処理機能を図4に示し、以下で説明する。

- (1) AP プロセスは、一括処理依頼機能のインタフェースを利用し、複数の処理を仲介サーバへ依頼する。
- (2) 仲介サーバは、依頼キューから制御用 ICA を取得する。
- (3) 仲介サーバは、取得した依頼の処理種別に対応する OS サーバの状態を調べる。
(A) WAIT 状態の OS サーバがある場合、当該 OS サー

表2 OS サーバ状態調査システムコール

形式	<code>unsigned int is_idle(servertypeid)</code> <code>unsigned int servertypeid;</code>
引数	<code>servertypeid</code> : 処理種別
戻り値	成功: OS サーバの識別子 失敗: ERROR (0xffffffff)
機能	引数で指定した処理種別の OS サーバについて、WAIT 状態であるものを示す。なお、WAIT 状態の OS サーバが複数ある場合は、最初に生成されたものを示す。

バに処理依頼を行う。

(B) WAIT 状態の OS サーバが無い場合、処理種別に対応する要求箱へ依頼を登録する。

(4) OS サーバは、自身の依頼キューまたは要求箱から制御用 ICA を取得し、依頼された処理を実行し、処理の実行結果を依頼元の AP プロセスへ返却する。

一括処理機能としての処理は (2) と (3) であり、(1) と (4) は従来と同様の処理である。また、OS サーバの状態を調べるシステムコールの仕様を表2に示す。

4 おわりに

一括処理依頼機能と要求箱通信機能の長所を併せもつことで効率的な負荷分散を可能にする一括処理機能を提案した。一括処理機能は、AP プロセスが一括処理依頼機能のインタフェースを利用するため、通信オーバーヘッドを削減できる。また、仲介サーバが、OS サーバの負荷状況に合わせて依頼先を選択するため、即応性を保証しつつ複数の処理を負荷分散できる。残された課題として、実装と評価がある。

参考文献

- [1] 佐藤 将也, 谷口 秀夫, 村岡 勇希, 山内 利宏: OS 処理の分散実行を効果的に利用できる一括処理依頼機能の実現と評価, 情報処理学会論文誌, Vol.60, No.2, pp.430-439 (2019).
- [2] 寺本 大風, 佐藤 将也, 山内 利宏, 谷口 秀夫: OS サーバ処理の負荷分散を可能にする *AnT* の要求箱通信機能, 情報処理学会研究報告, Vol.2018-OS-144, No.6, pp.1-8 (2018).