AnT における通信制御サーバ入れ替え処理の評価

藤原康行[†1] 井上喜弘[†1] 後藤佑介[†1]
山内利宏[†1] 乃村能成[†1] 谷口秀夫[†1]

マイクロカーネル構造を持つ ***AnT*** オペレーティングシステムについて，OS サーバの入れ替え機能が提案され，OS サーバの処理内容と入れ替え処理時間の関係，および入れ替え処理時間と応答時間の関係が報告されている．ここでは，OS サーバに処理を依頼するプロセス，あるいは OS サーバの依頼により処理を実行するドライバプロセスの処理内容と入れ替え処理時間の関係を明らかにする．また，UDP/IP 通信を行う通信制御サーバの入れ替えについて評価する．具体的には，通信制御サーバのプログラム構造を入れ替え可能な構成に変更した場合における性能への影響を明らかにする．さらに，入れ替え処理時間と送信処理時間の関係を明らかにする．

Evaluation of Dynamic Communication Control Server Replacement Processing for *AnT*

Yasuyuki Fujiwara,[†1] Yoshihiro Inoue,[†1]
Yusuke Gotoh,[†1] Toshihiro Yamauchi,[†1]
Yoshinari Nomura[†1] and Hideo Taniguchi [†1]

We proposed a dynamic OS server replacement mechanism, and reported a relationship between OS server processing contents and replacement time, and reported a relationship between replacement time and response time in ***AnT*** operating system based on micro kernel architecture. In this paper, we show a relationship between the processing contents of a process to request OS server or a driver process requested from OS server and replacement time. In addition, we evaluate replacing a communication control server executing UDP/IP communication. Specifically, we show effect for performance with making program architecture of communication control server exchangeable. Moreover, we show relationship between replacement time and processing time of packet transmission.

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

1. はじめに

計算機システムの重要性が増大するとともに，計算機システムに対して高い適応性と堅牢性が要求されている．特に，システムの基盤ソフトウェアであるオペレーティングシステム（以降，OS と略す）において，高い適応性と堅牢性を実現することは非常に重要である．

OS の適応性と堅牢性を高める手法として，マイクロカーネル構造 OS[1)–3)] がある．マイクロカーネル構造 OS は，OS 機能の一部をマイクロカーネルとして構築し，他の大半の OS 機能をプロセス（以降，OS サーバと略す）として実現する．OS 機能の追加や削除は OS サーバの追加や削除として実現できるため，システムが必要とする OS 機能を柔軟に構成でき，高い適応性を実現できる．また，OS 機能の不具合発生時の対処として，プログラム入れ替え機能[4),5)] がある．この機能は，モノリシックカーネル構造 OS において，不具合が生じた OS のプログラムモジュールの入れ替えで用いられる．多くの場合，プロセス相互の関係はプログラムモジュール相互の関係に比べて簡易であるため，マイクロカーネル構造 OS では，不具合が生じた OS サーバの入れ替え，つまりプロセスの入れ替えを比較的容易に実現でき，モノリシックカーネル構造 OS に比べて高い堅牢性を実現できる．

我々は，マイクロカーネル構造を持つ ***AnT*** オペレーティングシステム[6),7)]（An operating system with adaptability and toughness）について，OS サーバの入れ替え機能を提案[8)] し，OS サーバの処理内容と入れ替え処理時間の関係，および入れ替え処理時間と応答時間の関係を報告した[9)]．

ここでは，OS サーバに処理を依頼するプロセス，あるいは OS サーバの依頼により処理を実行するドライバプロセスの処理内容と入れ替え処理時間の関係を明らかにする．また，UDP/IP 通信を行う通信制御サーバの入れ替えについて評価する．具体的には，通信制御サーバのプログラム構造を入れ替え可能な構成に変更した場合における性能への影響を明らかにする．さらに，入れ替え処理時間と送信処理時間の関係を明らかにする．

2. OS サーバ入れ替え機能

2.1 サーバ間通信制御機構

AnT オペレーティングシステムにおけるプログラム間の呼び出し制御の基本機構[7)] を**図 1** に示す．カーネル（内コア）は，プロセスごとに通信のための依頼キューと結果キューを用意する．プロセスは，制御用 ICA（Inter-core Communication Area）とデータ用 ICA の授受（仮想空間上の剥がしと貼り付け）により，各 OS サーバへの処理依頼や結果取得を

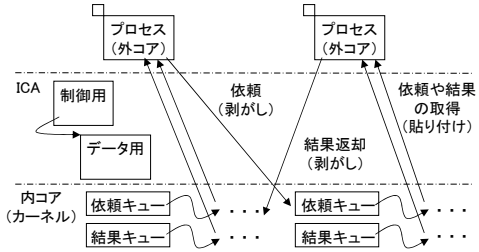


図 1　プログラム間の呼び出し制御の基本機構

行う．

2.2　OS サーバ入れ替え機能の実現方法

OS サーバ入れ替え機能[8] を実現するために次の対処を行う[9]．

通信先 OS サーバの特定法として，OS サーバが提供する OS 機能に識別子（機能識別子）を設け，機能識別子をもとに通信相手を特定する通信制御機構を設ける．さらに，OS サーバのプロセス識別子と機能識別子を関連付ける管理構造を設ける．

新旧 OS サーバ間の情報移譲法として，OS サーバ入れ替えにより移譲する情報を OS サーバ外部の ICA に保持する．また，OS サーバ入れ替え処理では移譲する情報の位置情報を新 OS サーバに通知する．

入れ替え処理時間の短縮法として，OS サーバとの通信情報を格納する領域をカーネル内に実現し，OS サーバの処理に原子性をもたせる．

本実現方法の特徴は，新旧 OS サーバ間の情報移譲法として ICA を利用していることにより，OS サーバ入れ替えの際に情報の複写を不要にしている点である．ICA[7] は，カーネルとプロセス（OS サーバを含む）間で共有できる領域であり，各々が独自に確保と解放を行うことができる．本実現方法では，OS サーバが情報格納域として ICA を確保し，その位置情報をカーネルに通知する．つまり，ICA はカーネルが占有する領域ではなく，OS サーバの情報を ICA に置くことは，カーネルの機能を最小化するマイクロカーネル構造 OS の考え方に反しない．一方，MINIX3[10] は，入れ替え対象 OS サーバとは別のデータ格納サーバに情報を持たせるため，内容一致のための更新処理が頻繁に発生してしまう欠点がある．さらに，入れ替えの際には，データ格納サーバから新 OS サーバへの情報の複写も発生してしまう．

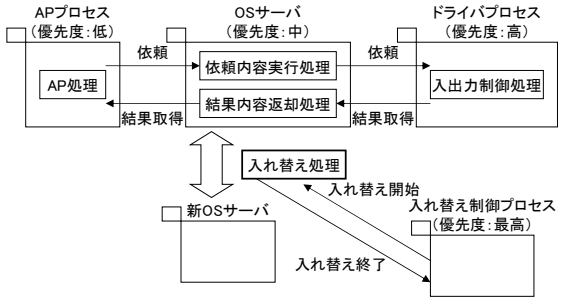


図 2　OS サーバ入れ替え処理の様子

2.3　考　　察

OS サーバ入れ替え処理の様子を**図 2** に示す．文献 9) では，次のことを明らかにしている．

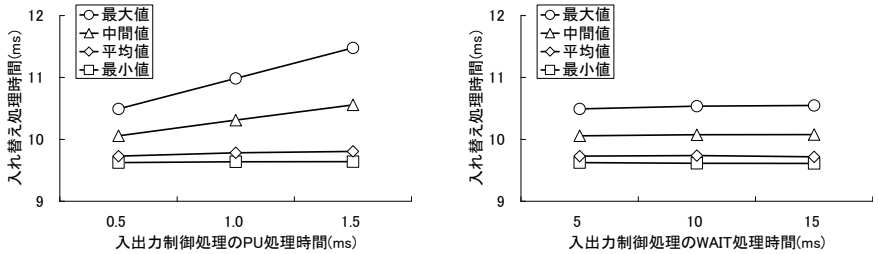
入れ替え処理時間は，OS サーバが依頼内容実行処理，または結果内容返却処理を実行中，およびドライバプロセスが入出力制御処理を実行中のいずれかにおいて，入れ替え開始が指示された場合に長大化する．入れ替え処理時間の長大化を防ぐためには，OS サーバ処理時間の短縮とともに，プロセスの優先度設定を工夫して高優先度プロセスの数を必要最小限とすることが重要である．また，応答時間は，OS サーバが依頼内容実行処理，または結果内容返却処理を開始する前，および AP プロセスが結果取得を行う直前のいずれかにおいて，入れ替え開始が指示された場合に長大化する．

3.　基 本 評 価

3.1　測 定 条 件

文献 9) では，入出力制御処理と AP 処理の時間が一定，および AP プロセス数が 1 個である場合において，依頼内容実行処理と結果内容返却処理の時間が入れ替え処理時間に与える影響，および入れ替え処理時間が応答時間に与える影響について述べた．ここでは，依頼内容実行処理と結果内容返却処理の時間が一定である場合において，入出力制御処理と AP 処理の時間，および AP プロセス数が入れ替え処理時間に与える影響を評価する．

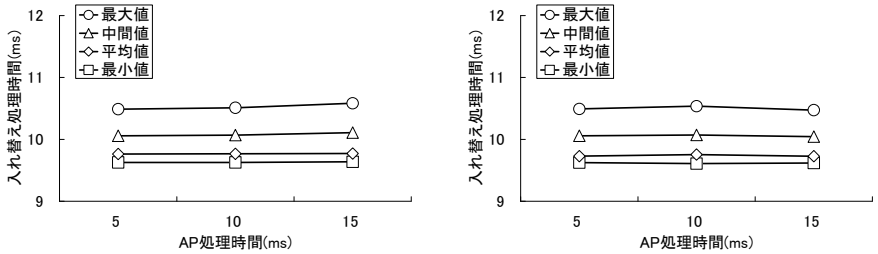
Intel® Celeron® プロセッサ（2.0GHz）を搭載した計算機上で ***AnT*** を走行させ，入れ替え処理時間を測定する．ここで，依頼内容実行処理と結果内容返却処理は 0.5 ミリ秒の PU 処理を実行する．また，AP 処理は PU 処理，または WAIT 処理を実行し，入出力制御処理は PU 処理と WAIT 処理を組み合わせて実行する．なお，PU 処理は特定の領域の



（A）入出力制御処理の WAIT 処理が 5 ミリ秒の場合　（B）入出力制御処理の PU 処理が 0.5 ミリ秒の場合

図 3　入出力制御処理時間が入れ替え処理時間に与える影響

（AP 処理 : WAIT 処理 5ms，AP プロセス数 : 1 個）



（A）AP 処理が PU 処理の場合　（B）AP 処理が WAIT 処理の場合

図 4　AP 処理時間が入れ替え処理時間に与える影響

（入出力制御処理 : PU 処理 0.5ms と WAIT 処理 5ms の組み合わせ，AP プロセス数 : 1 個）

インクリメントを繰り返すプロセッサ処理であり，WAIT 処理は指定時間だけ実行権を放棄するシステムコールを用いた待ち処理である．

4.3 節に後述するように，入出力制御処理の待ち処理時間は，入出力制御処理のプロセッサ処理時間や依頼内容実行処理時間，および結果内容返却処理時間の 10 倍以上である．このため，入出力制御処理の WAIT 処理時間を 5 ミリ秒〜15 ミリ秒とし，残り 3 種類の処理時間を 0.5 ミリ秒〜1.5 ミリ秒と設定した．

3.2　結果と考察

3.2.1　入出力制御処理時間の影響

AP 処理は 5 ミリ秒の WAIT 処理，AP プロセス数は 1 個である場合において，入出力制御処理の WAIT 処理時間を 5 ミリ秒としたときの測定結果を**図 3**（A）に示す．また，入出力制御処理の PU 処理時間を 0.5 ミリ秒としたときの測定結果を図 3（B）に示す．図 3 から，次のことがわかる．

(1)　入出力制御処理の PU 処理時間の増加に伴い，入れ替え処理時間が長大化

これは，入出力制御処理の PU 処理が完了するまで入れ替え処理が待たされるためである．入出力制御処理の PU 処理の実行中に入れ替え制御プロセスに制御移行し，入れ替え開始が指示された場合，ドライバプロセスは OS サーバに比べて高優先度であるため，ドライバプロセスに制御移行し，入出力制御処理の PU 処理が再開される．このとき，入出力制御処理の PU 処理が完了するまで入れ替え処理が待たされ，入れ替え処理時間が長大化する．

(2)　入出力制御処理の WAIT 処理時間の長短に関わらず，入れ替え処理時間は一定

これは，入れ替え処理の間にドライバプロセスに制御移行したとしても，直後に入れ替え処理が再開されるためである．入出力制御処理の WAIT 処理時間が短いと，入れ替え処理の間にドライバプロセスに制御移行する割合が高くなる．しかし，ドライバプロセスは OS サーバへ結果返却を行った後に依頼待ちに移行するため，OS サーバに制御移行し，入れ替え処理が再開される．なお，これら一連の処理に要する時間は，入れ替え処理での新 OS サーバ生成時に発生する磁気ディスクの入出力時間に比べて非常に短い．このため，入れ替え処理時間に与える影響は極めて小さい．

3.2.2　AP 処理時間の影響

入出力制御処理は 0.5 ミリ秒の PU 処理と 5 ミリ秒の WAIT 処理の組み合わせ，AP プロセス数は 1 個である場合において，AP 処理を PU 処理，または WAIT 処理としたときの測定結果を**図 4** に示す．図 4 から，次のことがわかる．

(1)　AP 処理の時間や内容に関わらず，入れ替え処理時間は一定

これは，AP 処理の実行中は直ちに入れ替え処理を開始できるためである．AP 処理の実行中に入れ替え制御プロセスに制御移行し，入れ替え開始が指示された場合，OS サーバは AP プロセスに比べて高優先度であるため，OS サーバに制御移行し，直ちに入れ替え処理を開始できる．このため，AP 処理の内容や時間は入れ替え処理時間に影響を与えない．

3.2.3　AP プロセス数の影響

入出力制御処理を 0.5 ミリ秒の PU 処理と 5 ミリ秒の WAIT 処理の組み合わせ，AP 処理を 5 ミリ秒の WAIT 処理としたときの測定結果を**図 5** に示す．図 5 より，以下のことがわかる．

(1)　AP プロセス数が 5 個以下のとき，AP プロセス数の増加に伴い，入れ替え処理時間

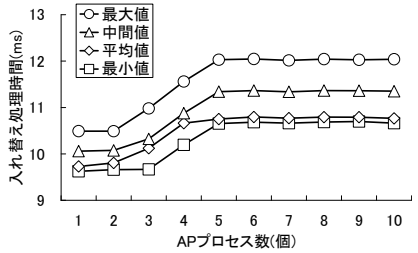


図 5　AP プロセス数が入れ替え処理時間に与える影響

（入出力制御処理：PU 処理 0.5ms と WAIT 処理 5ms の組み合わせ，AP 処理：WAIT 処理 5ms）

が長大化

これは，入れ替え開始待ちや入れ替え処理の間にドライバプロセスに制御移行し，入れ替え処理が待たされるためである．入出力制御処理の WAIT 処理時間が依頼内容実行処理時間に比べて長いため，AP プロセス数が増えるとドライバプロセスの依頼キューに処理依頼が連なるようになる．ここで，入れ替え開始待ち，すなわち依頼内容実行処理や結果内容返却処理といった OS サーバ処理の間に入れ替え制御プロセスに制御移行し，入れ替え開始が指示されたときの残存 OS サーバ処理の間，または入れ替え処理の間に入出力制御処理の WAIT 処理が完了すると，ドライバプロセスは OS サーバに比べて高優先度であるため，ドライバプロセスに制御移行する．ドライバプロセスは，OS サーバに処理結果を返却した後，連なっている処理依頼に対応する入出力制御処理の PU 処理を実行するため，（入れ替え開始待ちや入れ替え処理の間にドライバプロセスに制御移行した回数）×（入出力制御処理の PU 処理時間）分だけ入れ替え処理が長大化する．

(2)　AP プロセス数が 5 個以上のとき，AP プロセス数に関わらず，入れ替え処理時間は一定

これは，入れ替え処理の間にドライバプロセスに制御移行できる回数が上限に達したためである．AP プロセスから処理を依頼しない状態，つまり OS サーバやドライバプロセスが動作しない状態で OS サーバを入れ替えたところ，入れ替え処理時間は 10 ミリ秒程度であった．測定条件より，依頼内容実行処理時間と結果内容返却処理時間が 0.5 ミリ秒，および入出力制御処理の WAIT 処理時間が 5 ミリ秒であるため，依頼内容実行処理や結果内容返却処理を開始した直後に入れ替え制御プロセスに制御移行し，入れ替え開始が指示されたとしても，多くても 3 回（$=\lceil(0.5+10)/5\rceil$）しかドライバプロセスに制御移行せず，入れ替え処理時間が一定になる．

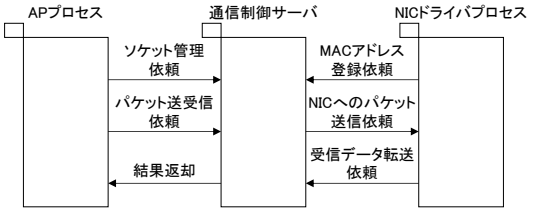


図 6　***AnT*** の通信機能の基本構造

4. 通信制御サーバ入れ替え

4.1 基 本 構 造

AnT の通信機能は，各プロトコルヘッダの設定やパケットを生成するプロトコル制御部と NIC（Network Interface Card）デバイスドライバ部をそれぞれ別プロセス（通信制御サーバと NIC ドライバプロセス）として実現している．***AnT*** の通信機能の基本構造を**図 6** に示す．

通信制御サーバは，AP プロセスと NIC ドライバプロセスに対し，ソケットの生成やパケットの送受信といった通信に利用する各種機能を提供する．通信制御サーバは，AP プロセスや NIC ドライバプロセスからの処理依頼を取得し，必要に応じて処理結果を依頼元プロセスへ返却する．

NIC ドライバプロセスは，通信制御サーバからのパケット送信依頼を取得し，NIC デバイスへパケットを転送する．また，NIC デバイスがパケットを受信すると，NIC デバイスから受信データを受け取り，通信制御サーバに対し，受信データ転送依頼を発行する．

4.2 プログラム構造と性能

通信制御サーバは，AP プロセスに関連付けるソケット情報やパケット生成に用いるプロトコル制御情報といった数多くの内部状態情報を保持している．そこで，2.2 節で述べたように，通信制御サーバの内部状態情報を ICA に格納するように変更する．具体的には，**図 7** に示すように，サービスに必要なデータを通信制御サーバプログラムのデータ部内ではなく，内部状態情報保存用 ICA に格納する．サービス提供時には内部状態情報保存用 ICA 上のデータを読み書きすることで，内部状態情報を通信制御サーバから分離する．

このプログラム構造の変更に伴う性能低下について評価する．**表 1** に示す送信計算機側

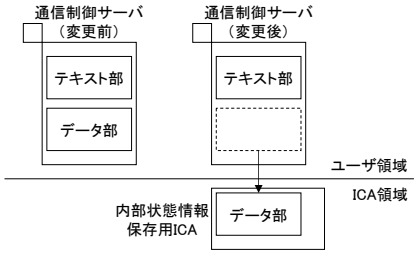


図 7　通信制御サーバプログラム構造の変更点

表 1　評価環境

	送信計算機（測定計算機）	受信計算機
OS	***AnT***	FreeBSD 4.3-RELEASE
CPU	Intel® Celeron® プロセッサ（2.0GHz）	Intel® Pentium® 4 プロセッサ（2.8GHz）
NIC	Realtek 8139 PCI Ethernet Card	Realtek 8139 PCI Ethernet Card

表 2　プログラム構造の変更に伴う性能低下

	変更前	変更後	差分
平均送信処理時間（ms）	14.95	15.56	0.61（4.1%）

で変更前，または変更後の通信制御サーバを走行させ，UDP/IP 通信において 1024 バイトのデータを 100 回送信したときの送信処理時間を測定する．

測定を 100 回試行したときの平均送信処理時間を**表 2** に示す．表 2 より，プログラム構造の変更に伴う性能低下は 4.1%程度であることがわかる．これは，プログラム構造の変更に伴い，内部状態情報へのアクセス速度が低下したためと考えられる．プログラムのデータ部内の各データは，コンパイル時やリンク時にそれぞれ静的な位置情報（アドレス）が割り当てられる．このため，必要とするデータへのアクセスはアドレス指定で直接行うことができる．一方，ICA はプログラム実行時に動的に割り当てられる．このため，必要とするデータへのアクセスはポインタ操作を通して間接的に指定する必要があり，アクセス速度が低下する．

4.3　入れ替え処理時間と送信処理時間

ここでは，データ送信中に通信制御サーバを入れ替えたときに発生する入れ替え処理時間が送信処理時間に与える影響について評価する．具体的には，UDP/IP 通信において，1024 バイトのデータを 100 回送信する間に 1 度だけ通信制御サーバを入れ替えたときの入れ替え処理時間と送信処理時間を測定する．この測定を 100 回試行した．なお，入れ替え開始のタイミングは疑似乱数を用いてランダムとした．

表 3　送信処理における通信制御サーバと NIC ドライバプロセスの処理時間

通信制御サーバ	依頼内容実行処理	0.0075ms のプロセッサ処理
	結果内容返却処理	0.0022ms のプロセッサ処理
NIC ドライバプロセス	入出力制御処理	0.0104ms のプロセッサ処理と 0.1131ms の待ち処理

ここで，送信処理における通信制御サーバと NIC ドライバプロセスの処理時間は**表 3** の通りであった．また，AP プロセスから処理を依頼しない状態，つまり通信制御サーバや NIC ドライバプロセスが動作しない状態で通信制御サーバを 10 回入れ替えたところ，入れ替え処理時間は，最小 32.8 ミリ秒，平均 33.3 ミリ秒，最大 34.6 ミリ秒であった．なお，この処理時間のばらつきは磁気ディスク装置の回転待ちやシーク待ちが原因と考えられる．また，上記の入れ替え処理時間は，3.2.3 節の（2）で述べた入れ替え処理時間（10 ミリ秒程度）に比べて大きい．これは，3.2.3 節の OS サーバプログラムサイズが 8KB 程度であるのに対し，本章で用いた通信制御サーバプログラムサイズが 40KB 程度と大きいため，新しい通信制御サーバ生成時に発生する磁気ディスクの入出力時間が増加したためである．

得られた結果を**図 8** に示す．図 8 から，次のことがわかる．

(1)　入れ替え処理時間の増加量は小さい

入れ替え処理時間は最小で 32.9 ミリ秒程度，最大で 35.2 ミリ秒程度であり，通信制御サーバや NIC ドライバプロセスが動作しない状態で入れ替え処理を行ったときの処理時間に近い．これは，表 3 に示すように通信制御サーバや NIC ドライバプロセスの処理時間が新しい通信制御サーバ生成時に発生する磁気ディスクの入出力時間に比べて非常に短いため，入れ替え処理時間に与える影響が極めて小さいといえる．

(2)　送信処理時間の増加量は大きい

送信処理時間は最小で 48.6 ミリ秒程度，最大で 51.0 ミリ秒程度である．表 2 に示すように通信制御サーバを入れ替えない場合の平均送信処理時間は 15.6 ミリ秒程度であるため，最小で 33.0 ミリ秒程度，最大で 35.4 ミリ秒程度，送信処理時間が増加している．この増加分は，入れ替え処理時間に相当しており，入れ替え処理の影響を受けているといえる．

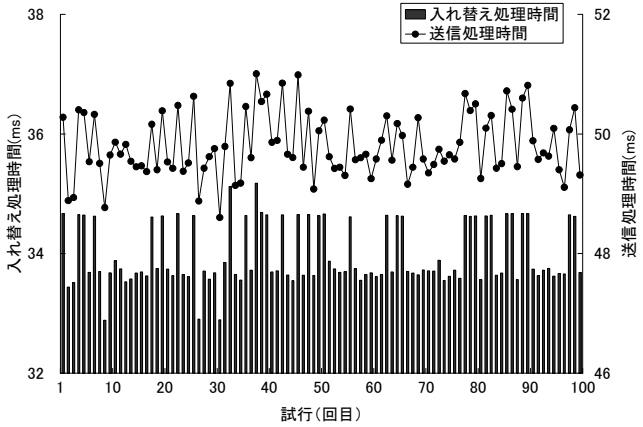


図 8 入れ替え処理時間と送信処理時間

5. ま と め

AnT オペレーティングシステムに実現した OS サーバ入れ替え機能について，OS サーバに処理を依頼するプロセス，あるいは OS サーバの依頼により処理を実行するドライバプロセスの処理内容と入れ替え処理時間の関係を明らかにした．評価では，入出力制御処理のプロセッサ処理時間の増加に伴い，入れ替え処理時間が長大化することを示した．また，入出力制御処理の待ち処理時間，および AP 処理の時間や内容は，入れ替え処理時間に影響を与えないことを示した．さらに，AP プロセス数の増加に伴い，入れ替え処理中に高優先度プロセスが実行され，その処理時間分だけ入れ替え処理時間が長大化することを示した．また，UDP/IP 通信を行う通信制御サーバについて，プログラム構造を入れ替え可能な構成に変更することで 4.1%程度性能が低下することを示した．さらに，UDP/IP 通信におけるデータ送信中に通信制御サーバを入れ替えたとき，送信処理時間に 33.0 ミリ秒〜35.4 ミリ秒程度の影響を与えることを示した．

残された課題として，入れ替え処理時間の短縮と入れ替え処理がサービスに与える影響の抑制がある．

参 考 文 献

1) David L. Black, David B. Golub, Daniel P. Julin, Richard F. Rashid, Pichard P. Draves, Randall W. Dean, Alessandro Forin, Joseph Barrera, Hideyuki Tokuda, Gerald Malan, and David Bohman, “Microkernel Operating System Architecture and Mach,” Journal of Information Processing, Vol.14, No.4, pp.442-453, 1992.
2) Jochen Liedtke, “Toward Real Microkernels,” Commnuications of The ACM, Vol.39, Issue 9, pp.70-77, 1996.
3) Andrew S. Tanenbaum, Jorrit N. Herder, and Herbert Bos, “Can we make operating systems reliable and secure?,” IEEE Computer Magazine, Vol.39, No.5, pp.44-51, 2006.
4) 谷口 秀夫, 伊藤 健一, 牛島 和夫, “プロセス走行時におけるプログラムの部分入替え法,” 電子情報通信学会論文誌 (D-I), Vol.J78-D-I, No.5, pp.492-499, 1995.
5) Linux Journal Staff, “Kernel Korner: Dynamic Kernels - Modularized Device Drivers,” Linux Journal, Vol.1996, Issue 23, No.7, 1996.
6) 谷口 秀夫, 乃村 能成, 田端 利宏, 安達 俊光, 野村 裕佑, 梅本 昌典, 仁科 匡人, “適応性と堅牢性をあわせ持つ ***AnT*** オペレーティングシステム,” 情報処理学会研究報告, 2006-OS-103, Vol.2006, No.86, pp.71-78, 2006.
7) 岡本 幸大, 谷口 秀夫, “***AnT*** オペレーティングシステムにおける高速なサーバプログラム間通信機構の実現と評価,” 電子情報通信学会論文誌, Vol.J93-D, No.10, 2010, 掲載予定.
8) 藤原 康行, 岡本 幸大, 田端 利宏, 乃村 能成, 谷口 秀夫, “***AnT*** における OS サーバ入れ替え機能,” マルチメディア通信と分散処理ワークショップ論文集, Vol.2008, No.14, pp.201-206, 2008.
9) 谷口 秀夫, 藤原 康行, 後藤 佑介, 田端 利宏, 乃村 能成, “***AnT*** における OS サーバ入れ替え機能の評価,” マルチメディア通信と分散処理ワークショップ論文集, Vol.2009, No.9, pp.261-266, 2009.
10) Jorrit N. Herder, Herbert Bos, Ben Gras, Philip Homburg, and Andrew S. Tanenbaum, “Failure Resilience for Device Drivers,” Proceedings of the 37th Annual IEEE/IFIP Conference on Dependable Systems and Networks, pp.41-50, 2007.