

プログラム実行速度調整による通信帯域制御

田端 利宏 乃村 能成 谷口 秀夫

九州大学 大学院システム情報科学研究所

1 はじめに

インターネットの普及により、複数の計算機を関連づけるサービスが数多く登場している。これらのサービスは、実時間的な処理からバッチ的な処理まで多種多様である。しかも、各サービスは、同時に共存して実行される。このようなサービス実行環境を利便性が高い形で提供するには、サービスが行う通信をうまく制御できる必要がある。そこで、本論文では、コンテンツの通信を行うプログラムの実行速度を調整する手法として、プロセッサの割り当ての程度を制御することで帯域を自由に制御する手法を示す。

2 プログラム実行速度調整法

プロセッサの割り当ての程度を制御するプログラムの実行速度調整方式^[1]では、OS のスケジューラが、プログラムの実行と停止を繰り返すことで、プログラムの実行速度を調整する。プログラム実行の単位時間をタイムスロットと名付け、タイムスロットの一定の連続した時間(タイムブロックと名付ける)内で要求された実行速度に必要な個数分のタイムスロットの時間だけ、スケジューラがプロセスにプロセッサを割り当てる。

要求性能に基づき n 個のタイムスロットを割り当てる場合、 i 番目に割り当てるタイムスロットの位置を、 $(1 \text{ タイムブロック内のタイムスロット数}) \times (i - 1) / n$ とし、OS がタイムスロットをプロセスに割り当てる。

3 実験内容と結果

3.1 実験内容

2 台の計算機を 100Mbps の Ethernet を用いてスイッチングハブを介して接続し、3 回実験を行った。送信側計算機(プロセッサ: Celeron 533MHz, OS: **Tender**^[1], NIC: DEC DE500-BA)では、send システムコールを発行して、ソケットを利用して 1MB のデータを送信するプログラムを実行した。また、受信側計算機(プロセッサ: Celeron 1GHz, OS: Linux 2.4.19, NIC: Intel Ether Express Pro 100)では、recv システムコールを発行し、ソケットを利用して 1MB のデータを受信するプログラムを実行した。このとき、送信側計算機では、データを送信するプログラムの調整性能(%), および

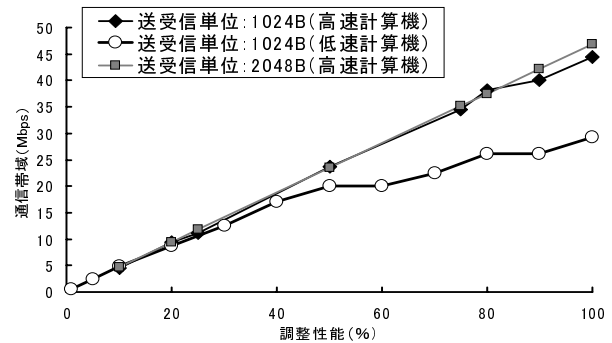


図 1 調整性能と通信帯域の関係

send システムコールで送信要求するデータの単位を変化させ実験した。また、受信側では、送信側で send システムコールで送信要求するデータの単位に合わせて、recv システムコールで受信要求するデータの単位を設定した。通信の開始と終了の時刻、および各バケットの到着時刻の測定を受信側計算機で行った。具体的には、受信側計算機の Ethernet ドライバに受信完了時刻を記録する関数を組み込んだ。なお、送信側計算機でのタイムスロットの大きさは $1ms$ で、タイムブロックの長さは $1s$ として実験した。

3.2 実験結果

3.2.1 通信帯域

上記の実験環境に加え、送信側と受信側計算機を低速計算機(送信側, プロセッサ: Pentium 133MHz, OS: **Tender**, NIC: DEC DE500-BA, 受信側, プロセッサ: Pentium III 750MHz, OS: BSD/OS 3.1, NIC: DEC DE500)に置き換えた場合について、通信帯域を調整した結果を図 1 に示す。図 1 から、高速計算機の場合、送受信単位に関係なく、調整性能に合わせて通信帯域を調整できていることがわかる。一方、低速計算機の場合、調整性能に比例した性能に調整できていないことがわかる。これは、送信側計算機のプロセッサ性能が低く、NIC の性能を十分に発揮できていないためであると推察できる。

3.2.2 パケット到着間隔

送受信単位 1024B と 2048B のときのパケット到着間隔の関係を図 2 と図 3 に示す。図 2 と図 3 から、以下のことがわかる。

(1) 送受信単位 2048B (図 3) では、送受信単位 1024B (図 2) に対し、到着間隔 $100\mu s$ 未満のパケットの割合が多い。

(2) 両者とも、パケット到着間隔 $400\mu s$ 以上のパケットが存在し、その到着間隔のばらつきが異なる。つまり、

Communication Band Control by Regulating Program Execution Speed

Toshihiro TABATA, Yoshinari NOMURA and Hideo TANIGUCHI

Faculty of Information Science and Electrical Engineering, Kyushu University

E-mail: {tabata, nom, tani}@csce.kyushu-u.ac.jp

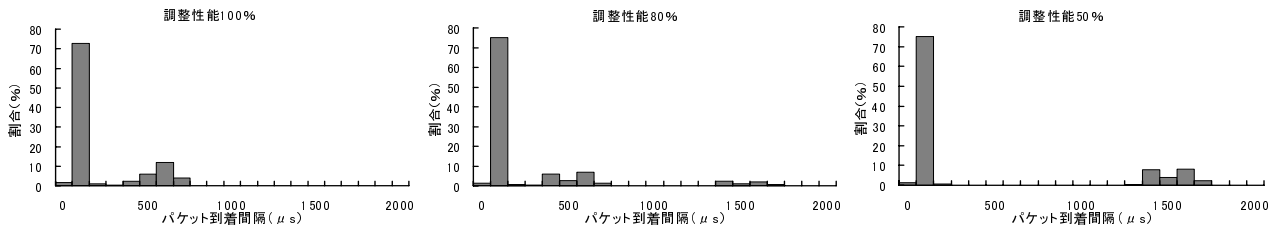


図 2 パケット到着間隔 (送受信単位: 1024B)

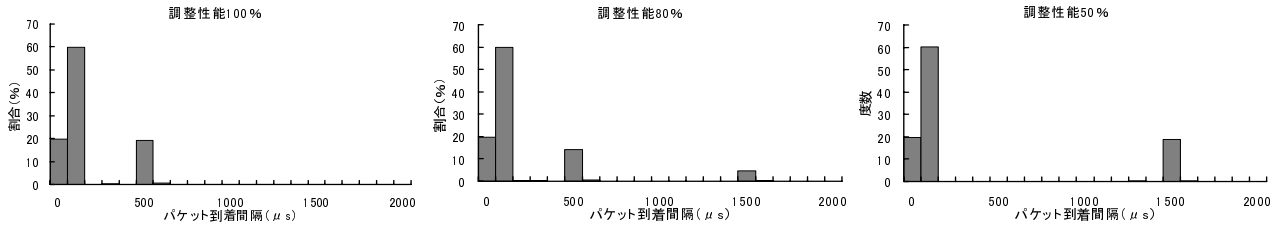


図 3 パケット到着間隔 (送受信単位: 2048B)

表 1 到着間隔 100µs 未満のパケット数 (調整性能 100%)

送受信 単位	パケットの大きさ (単位:100B)							
	2	5	7	8	11	13	14	16
1024B	2	3	2	1	29	0	0	0
2048B	3	504	3	0	0	0	0	0

送受信単位 1024B の場合、到着間隔がばらつき、送受信単位 2048B の場合、到着間隔が集中している。例えば、送受信単位 1024B、調整性能 100% の場合、それらのパケットは 400µs の範囲 (400µs 以上 800µs 未満) に存在する。一方、送受信単位 2048B、調整性能 100% の場合、それらのパケットは、100µs の範囲 (500µs 以上 600µs 未満) に存在する。

(3) 図 2 と図 3 で、調整性能を低くすると、調整性能 100% で 400µs 以上のところにあった到着間隔の長いパケットの分布が、より到着間隔の長い方に移動していることがわかる。

以下に、上記の各項目について考察する。項目 (1) について、到着間隔 100µs 未満のパケットの大きさを表 1 に示し、説明する。送受信単位 2048B の場合、送信時にパケットが分割された結果、504 個のパケットが 400B 以上 500B 未満の大きさになったと推察できる。これらのパケットはその大きさが小さいため、その到着間隔が短いと推察できる。

項目 (2) について、NIC からパケットを送信する処理は、パケットの連続送信と、次の送信までの待ちの繰り返しである。この待ちのため、パケットの到着間隔の長いパケットが多く存在すると推察できる。

項目 (3) について、調整性能を 100% 未満に設定すると、タイムスロットを割り当てられない時間が生じることが原因と推察できる。具体的には、タイムスロットを割り当てられない間、送信側プログラムの送信処理が停止する。例えば、調整性能 50% では、タイムスロット

を一つおきに割り当てる。このため、パケットの送信処理は 1ms 間毎に停止し、パケット到着時間が調整性能 100% のときに比べて 1ms 長くなるパケットが生じる。次に、調整性能 80% では、四つ連続してタイムスロットを割り当て、一つのタイムスロット分だけ空けて、また四つのタイムスロットを割り当てる処理を繰り返す。このとき、4ms 連続してタイムスロットを割り当てるため、調整性能 100% で 400µs 以上の到着間隔のパケットが、調整性能 100% とほぼ同じ間隔で到着することがある。一方、タイムスロットの非割当時間のため、一部のパケットは到着間隔が 1ms 増加する。以上のことから、この場合の到着間隔が長くなるパケットの割合は、タイムスロットが連続して割り当てられる確率で決まるといえる。例えば、調整性能 80% の場合、一つタイムスロットの割当時間終了時に、1/4 の確率で次のタイムスロットが割り当てられていないため、調整性能 100% で到着間隔の長いパケットのうち 1/4 のパケットについて到着時間が長くなる。

4 おわりに

プログラム実行速度調整機能を利用して、通信帯域を調整できることを示した。また、到着間隔が長くなるパケットについては、そのパケット到着間隔が、タイムスロットの非割り当て時間の長さ分だけ長くなることを示した。さらに、到着間隔の長くなるパケットの割合は、タイムスロットが連続して割り当てられる確率で決まると示した。

今後の課題として、調整性能とパケット到着間隔の関係の分析がある。

参考文献

- [1] 田端利宏, 谷口秀夫: *Tender* オペレーティングシステムの資源「演算」によるプログラム実行速度調整機能の実現と評価, 情報処理学会論文誌, Vol.40, No.6, pp.2523-2533 (1999).