

5L-8

富士通VANでのOSI-VTプロトコル変換ゲートウェイについて  
タスクプライオリティの観点からのスループット考察

小谷野正博  
富士通(株) 複合情報通信システム本部

1. はじめに

OSI-VTの損保適用では、元受各社ホストはOSI-VTプロトコルで、代理店設置端末は既存の各社メーカープロトコルで接続するため、各社VANでOSI-VTプロトコルとメーカープロトコルを変換するゲートウェイを開発することになった。富士通VANではOSI-VTとFNA/F6650及びBSC-C/ABCとの変換を行っている。今後、VANへの收容端末数が急激に増加することから、VAN側ではゲートウェイのトランザクション処理件数を把握して設備計画に備える必要がある。

ところで、1台のゲートウェイのトランザクション処理件数は、1件の処理時間とトランザクション到着率をパラメータとして与えれば、待ち行列理論から目安を導き出せる。さらに、トランザクション処理件数を上げるには、処理時間を短く(走行ステップ数の削減、高性能マシンの使用)すれば良いことは明らかだ。本稿ではゲートウェイのマルチタスクのタスクプライオリティの設定方法が、高トラフィック時のゲートウェイ内滞在時間に影響を及ぼすことに注目し、損保適用に当たり現実のゲートウェイで滞在時間を最小にするプライオリティの設定方法をシミュレーションにより探ったものである。

2. ゲートウェイのタスク構造と処理時間分布

ホストから端末へ(下り)1画面分(24行×80列)のフルスクリーンデータを送信し、端末からホストへ(上り)入力項目データを返答する。この繰り返しが通常1業務で5~7回のおこなわれる。また、有るマシンで実測したところ、処理時間で上りは50~80msに対して、下りデータは500~800msで、処理タスク構造と処理時間分布の関係は概ね図1、表1のとおりであった。

3. マルチタスク処理でのタスクプライオリティ

待ち行列理論の「Littleの結果」から、到着率をλ, トランザクションのシステム内平均滞在個数をN, 平均滞在時間をTとすると、

$$N = \lambda T$$

の関係にある。つまり、平均滞在時間Tを小さくす

Gateway for OSI-VT protocol convert of Fujitsu-VAN(A study of throughput from task-priority viewpoint)

Masahiro KOYANO

Integrated Information Network Systems Group  
FUJITSU LIMITED

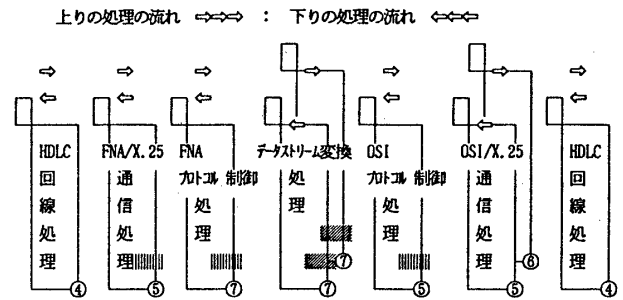


図1. タスク構造とタスクプライオリティ

表1. 現実ゲートウェイの処理時間分布 (%)

方向性	HDLC	FNA	カトコ	ストリーム	カトコ	OSI	HDLC
上り	15	10	7.0	35	3.0	20	10
下り	5	8	3.5	70	1.5	5	7

るには、λを固定にしたとき、システム内に滞在するジップ数Nを少なくする方向に働かせれば良い。また、マルチタスク処理の性質として、平均滞在時間は一般にジョブスケジューリング方式に依存し、残余サービス時間の最も短いジョブを優先処理するスケジューリング方式が平均滞在時間を最小にすることが知られている。(参考文献1) さらに、タスクプライオリティの設定次第で平均滞在時間が変わることが報告されている。(参考文献2)

以上を確認するため、以下の仕様のシミュレーションプログラムを作成し実施した。

①M/D/1の待ち行列を対象とする。つまり、

- ・発生データ時間間隔は指数分布に従う。
- ・処理時間は一様分布に従う。
- ・CPUは1つである。

②データはシステムの中に入ると、次々とサブプログラムを渡り歩く。このサブプログラムには、タスクプライオリティが設定されている。次のサブプログラムへの待ちキューは、キューの最後にリンクする。

③CPUはタスクプライオリティの高いものから与えられる。

④あるサブプログラムが実行中にそれよりタスクプライオリティが高いデータが発生すると、割り込みを起こして、実行を中断し、高いプライオリティのサブプログラムに制御を渡す。なおOSのオーバーヘッドは無視する。

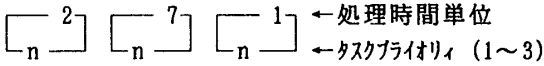
4. シミュレーション結果1

(1)タスク構成とタスクプライオリティの設定

以下の様に、3ジョブに分けると、タスクプライオリティの設定の仕方は $3^3 = 27$ 通りになる。

なお、数字の低い方を高プライオリティとする。

処理方向 ⇒



(2) シミュレーション条件

発生データ数 70ヶ。3回トライし、1トライではどのタイプも発生データは同一とした。1回目、2回目、3回目と発生データの乱数系を変えた。

(3) 結果

滞在時間を昇順に並べると表2の様になった。

なお、これはどのトライでも同じであった。

(4) 評価

一度受け付けたデータをひたすら処理するように、CPUを与えるプライオリティ設定が良い性能値を示している。このことは、入り側のプライオリティ設定を低く、出側のプライオリティ設定を高くすることである。

表2. タスクプライオリティと滞在時間

順位	タスクプライオリティ	滞在時間	順位	タスクプライオリティ	滞在時間
1	322, 321, 312, 311, 211	12.87	14	333, 222, 111	14.70
6	332, 331, 221	13.16	17	323, 313, 212	16.47
9	232, 231, 131, 121	13.76	20	233, 133, 122	17.19
13	132	13.81	23	112, 113, 123, 213, 223	19.30

5. 現実ゲートウェイでの制約

我々のゲートウェイは、図1の様なタスク構成を持つが、4章の評価から上り処理と下り処理のプライオリティ設定はちょうど逆に設定されたとき、データの滞在時間が最小となることが判る。しかし、現実には、既存の通信プロダクツ (OSI 下層制御, OSI 上位層制御, FNA 制御, 等) を組み合わせ使用し、その上に、上りと下りのプロトコルのマッピング処理や、データストリーム変換処理を新たに開発している為、既存通信プロダクツは上りや下り処理が1タスクであったり、また、タスク毎にタスクプライオリティの設定ができるものではなかった。

そこで、これら制約はあるものの、新規作成分については、上り下り処理毎にタスクプライオリティの設定を可能とした。

図1の無印はタスクプライオリティの設定が固定、 $\square$ は上り下り1タスクで可能、 $\times$ は上りタスク、 $\circ$ は下りタスクで別個に可能を示す。

6. シミュレーション結果2

(1) シミュレーションプログラムの拡張

実際のデータ処理形態に合わせて、以下の様に仕様を拡張した。

⑤上りデータを処理した後、回線經由ホスト処理後(有るデレイの後)、応答として下りデータが発生する。  
 ⑥上りデータ処理、下りデータ処理はそれぞれ別個の実行順序、タスクプライオリティ処理時間を有する。

(2) 最適タスクプライオリティの検出

4章の(4)評価に基づき、タスクプライオリティを変えて、最適の設定値を見いだした。

図2にその結果を示す。

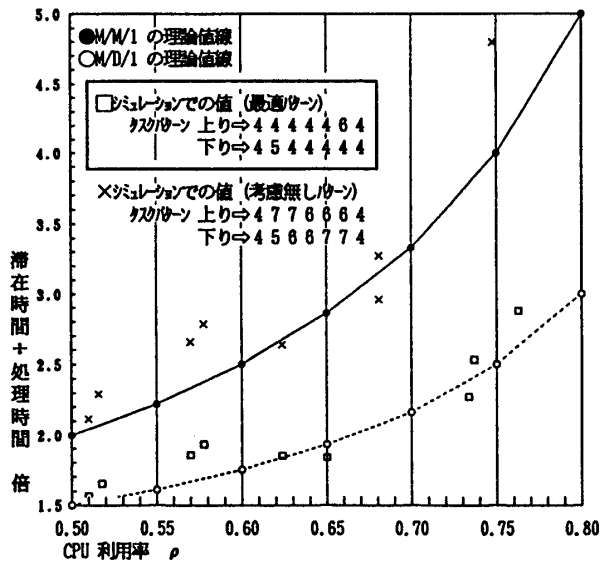


図2. 滞在時間のシミュレーション

□は現実システムで採り得る最適タスクパターンとその値、×は特に考慮していないときのタスクパターンとその値である。

7. おわりに

今回、損保適用ゲートウェイの開発に当たり、性能問題を待ち行列問題からも取り組んだ。その結果、ゲートウェイのタスク構成は、「上り下りで自由なタスクプライオリティ設定が可能であること」を学んだ。

図2で最適パターンの結果の多くがM/M/1とM/D/1の間に存在していることから、今後は、タスクプライオリティの設定のシミュレーション結果と待ち行列理論の「Pollaczek Khintchineの公式」との関係調べる必要が有りそうだ。

参考文献

- オペレーティングシステム① 性能解析 情報処理学会  
「2.1 処理効率と応答性」  
「2.5 多重プログラミング・システムの機能」
- 専用オンラインシステムにおけるマルチタスクシステム性能解析  
情報処理学会論文誌 Vol.21 No.4 July