L-6

自律型トラフィック制御ルータの設計と評価
Design of router for autonomous traffic control

白川　正知　古川　泰男
Masatomo SHIRAKAWA and Yasuo FURUKAWA

1.はじめに

われわれは、トラフィックに適応して自律的にルーティング制御を行うルータの開発を行っている。提案する適応制御は、トラフィック・フローを監視し、その履歴を元にルーティング時の転送先や転送スケジュール管理を行う手法をとる。[1, 2]

現在までに、基本制御アルゴリズムを構築し、簡単なネットワークモデルを想定したシミュレーションにより、その効果を確かめている。

本報告では、まず、提案する技術を実現するためのアーキテクチャを示す。それを元に、経路コストのための各種の計算パラメータに対する処理能力のシミュレーション結果を示す。

2.自律型トラフィック制御ルータのアーキテクチャ

本技術では、従来のルーティング機能だけではなく、トラフィックにしたがった適応制御、すなわちトラフィック状況を元にした優先制御機能を提供する。

以下では、既存技術との比較を行い、本技術に必要な構成を考える。

従来のルーティング技術では、最短経路コストを計算するためのアルゴリズム用に、ネットワークをエリア指定する必要があり、このことは困難な作業である。また、各ルータは、すべての経路情報をもたなければならず、データベースが大きくなると、その処理に計算時間がかかり、ルーティング処理に影響を及ぼす欠点がある。

一方、優先制御では、通信チャネルを占有したり、ラベルにより優先度を制御や管理したりする方法がおもに採られており、トラフィック状況に適応させる機構がない。このことは、優先制御の対象とはならない通信に対しては、帯域が十分確保できない場合、それらが排除されてしまう恐れがある。

以上の問題点を改善するために、本技術では、パケット交換時にネットワーク遅延も測定し、自動的にネットワーク・エリアを抽出するように動作する。また、ネットワーク構成の情報を格納するデータベースも、肥大化を抑制できる。それは、経路コストが低いルータをマスタ動作させ、このマスタ・ルータにより同一エリア内のルータの識別を行い、ルーティング情報としてスレイブ・ルータに配布する方式による。つまり、スレイブ・ルータは、管理するネットワークのルーティング・テーブル以外は、マスタ・ルータから受信するルータ識別表を持つ。そのルータ識別表は、各ルータ固有の識別子とインターフェイス識別子のみの単純な構成である。これらのことから、ルーティング・テーブルの肥大化を抑制する。

本技術の利点の一つは、実時間でのトラフィック監視に基づいた適応性[3]である。優先制御対象外の通信に対しても可能であれば、積極的にルーティングを行う。

以上のような本技術の特長を生かすためには、実時間でトラフィック監視[4]と経路コストの計算、スケジュール管理を同時に実現できることが求められる。よって、これらを独立して処理できる複数のパイプラインをもつ CPU が必要である。また、パケットのコピーを格納するバッファメモリとルーティング制御情報を格納するキャッシュメモリは同時にアクセスできることも必要である。図 1 に、ハードウェア構成の概略を示す。

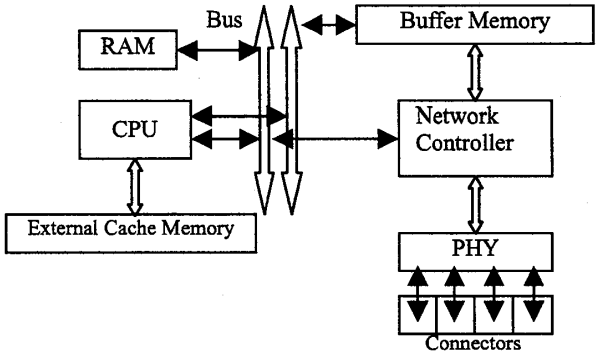

図 1．ルータ・ハードウェア構成

必要とされるメモリ容量は、先の報告[2]において示したように、バッファメモリ 8M あれば十分であることが分かっている。また、キャッシュメモリは 1MB 程度とする。

次に、ソフトウェア構成を図 2 に示す。マイクロカーネルは、すべてのハードウェアを管理し、ソフトウェアに実時間動作を提供するものである。また、TCP/IP スタックを実装している。ネットワークモニタ(1)は、マイクロカーネルの TCP/IP スタックから直接パケット情報を得る。ポート毎のトラフィック・フローの計算を行い、その履歴をキャッシュメモリ内に格納する。コスト計算用ソフトウェア(2)は、キャッシュメモリ内のトラフィック・フロー情報を取り出し、ポートすなわち経路ごとのコストを求める。その結果は、所定のメモリに格納する。スケジューラ(3)は、キューの管理を行う。ポート毎に設定されたキューにおいてコストにより、ポートごとに送出時間を調整する。

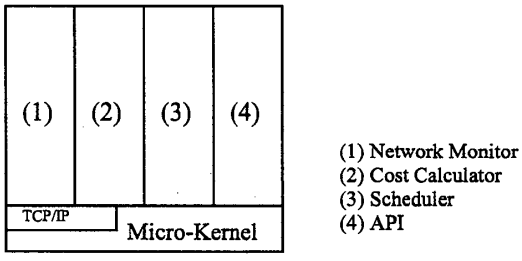

図 2．ルータ・ソフトウェア構成

豊橋技術科学大学　未来技術流動研究センター
Research Center for Future Technology, Toyohashi University of Technology

以上は、本技術を実現するための機能を提供するものであるが、API(Application Program Interface)(4)は、従来のルーティング・プロトコルに対する環境を提供する。新たに定義したルーティング情報の変換等を行う。OSPF(Open Shortest Path First)プロトコルなどを実装した場合、トラフィック条件により、本技術と従来プロトコルを切り換えて使用できる。すべてのソフトウェアは、FlashROM に格納し、ルータ起動時にアプリケーション用メモリに呼び出され、動作を開始できるようにする。

3.シミュレーション

図 1 と 2 で示した構成を C 言語により構築し、LAN(100BASE-TX)に接続した時、適応制御のためのパラメータを変えたときの提案ルータの処理能力をシミュレーションにより評価した。

ネットワーク構成は、セグメント間の単純なルーティングを行うものであり、図 3 に示す。パケットは eth0 側から入力され、eth1 側へ中継される。静的なルーティング情報となってしまうが、ルータの能力と制御パラメータによる変化を調べる。入力されるパケットのパターンを表 1 に示す。

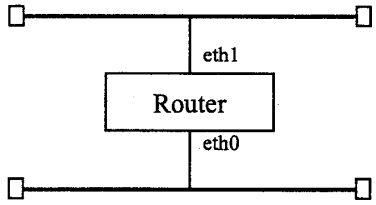

図 3.ネットワーク構成

表 1.入力パケットパターン

	パケットサイズ	パケット間隔
パターン 1	固定（64 バイト）	連続
パターン 2	ポアソン分布	ランダム
パターン 3	ランダム	ランダム

本技術では、コスト計算において、以下のように処理している。

（経路コスト）
＝（コスト履歴）＋（トラフィック変化量）/（帯域）
　－（バッファ空き容量）/（全バッファ容量）

上記評価式により、以下の場合コストが高くなり、パケット転送の待ち時間が調整される。

(1) コスト履歴、つまりコストの累積値が大きい
(2) 帯域に対して、フロー変化が相対的に大きい
(3) バッファ容量、つまり待ち行列内のパケット量が大きい

本シミュレーションにおけるパケット転送時の待ち時間の調整法として、コストに比例した待ち時間を与えることとする。待ち行列の単位待ち時間(= 1 タイムスロット)を w_u、経路コストを c_r とすると、

$$w = (1 + c_r)w_u$$

と定義する。ここで、直ちに転送を行う場合、経路コストを-1 とし、待ち時間を 0 とすることで、待ち行列の先頭に保存され、処理されるようにしている。

なお、作成したシミュレーション・ソフトウェアでは、動作遷移が重要である部分でのみ、システムクロックに基づいて同期をとっている。

シミュレーションでは、フレーム数 1,000,000 からなるフロー・パターン 1 から 3 を、それぞれ 1000 回試行し、スループットを計測した。その結果を表 2 に示す。

表 2.シミュレーション結果例

	平均スループット	平均遅延時間	誤り率
パターン 1	12.8 (MB/s)	0 (μs)	0 (%)
パターン 2	5.12 (MB/s)	1 (μs)	0.3 (%)
パターン 3	11.64 (MB/s)	1.1 (μs)	0.9 (%)

上記結果は、あて先アドレスはパターン 1 の場合、1 サイトのみ、その他は 100 種類用意し、ランダムに選択した場合のものである。シミュレーションの最小同期精度は 1(μs)である。すべてのフローは処理され、転送されており、処理による遅延時間も平均ではシミュレーションの同期精度程度となっている。また、フレーム順序が入れ替わったフレームが見られ、それを転送誤りとしてまとめた。あて先がランダムに変化する場合、フレーム順序を入れ替えてしまうものが多く見られた。これは、フローモニタの測定時間が 1μs と高速であったため、フレームサイズの変化に過剰に反応し、評価式条件(2)により、待ち時間が大きくなったことが原因と考えられる。

なお、これらの入力フローは、Opnet Modeler(R)により発生させたものである。

4.まとめ

本論分では、トラフィック・フローに応じ、ルーティングおよびフォワーディングを行うルータ制御技術を実現するために必要なルータの構成を示した。その構成に基づき、シミュレーションを行い、提案する適応制御技術によるルータ単体の性能を評価した。想定された条件下では、転送能力は十分であることがわかった。とくに、作成したアルゴリズムでは、処理時間によるパケット転送の遅延は、ほとんど見られなかった。ただし、トラフィック・フローによっては、その測定時間により転送時においてパケット等の転送順序に誤りが見られることが分かった。

今後は、同様の構成を持つルータを試作し、ネットワーク上での連携動作の検証実験を行う予定である。このとき、トラフィック・フローの測定時間などの制御パラメータを最適化も行う。

参考文献

[1]白川ほか:信学会総合大会, B-7-180 (2001)
[2]白川ほか: 信学会総合大会, SB-4-6 (2002)
[3]加藤ほか:情報処理学会第 63 回全国大会, 3-257 (2001)
[4]豊野ほか: 情報処理学会第 63 回全国大会, 3-269 (2001)