

マルチメディア情報アクセスに基づくトラフィックマネジメント方式

20-5

創価大学 工学研究科  
飛鷹 洋一 勅使河原 可海  
hidaka@t.soka.ac.jp

1. はじめに

現在の一般的なLANでは、LAN外にアクセスする際、そのLANのユーザは、LAN内の回線と比べて極めて低速度ないくつかの共有回線を介して外にアクセスしているが現状である。また、共有回線の使用は、そのネットワークのユーザに対して平等かつ自由に許されている。したがって、このようなLANにおける問題点として、共有回線に、多数多様のネットワークリソースの要求あった場合、ボトルネック問題が生じ、さらには、一人ユーザが、大量のリソースを要求した際、他のユーザは、その一人のユーザのためだけに、大きな影響を受けることになる。そこで、本論文では、それらの問題に対して、従来、ネットワーク下位層だけで判断してきたトラフィック制御を、上位層およびネットワーク全体で判断することで、限られたネットワーク資源で、効率的に運用する総合的トラフィックマネジメントのシステム設計及び評価を行う。

2. システムの構成

現在の全く制限のない共有回線を持つネットワークでは、マルチメディア情報のネットワークリソースの消費量の差や、利用者のネットワーク使用用途の違いなどが、問題を引き起こしていると考えられる。そのことから、本論文では、LANの全体のユーザのマルチメディア情報アクセスをトラフィック制御の評価の基準対象とし、総合的トラフィックマネジメントを行うことで、ユーザのネットワークレスポンスの不公平問題や、ネットワークリソースの不効率的消費の問題を解決する。また、システム構成は、一局集中型によるシステム低下およびシステムの障害危険性を避けるため、全体を管理するトラフィックマネジメントサーバ、部門を管理する部門サーバ、およびLAN内のコンピュータの3層構造で構成されており、LANに繋がっているコンピュータのトラフィックをダイナミックに管理し、ネットワーク全体のレスポンス向上をはかる。

3. トラフィックマネジメントサーバと部門サーバ

トラフィックマネジメントサーバ（以下：TMS）は、図1に示すように、部門サーバから、稼働APP集計マネージャを用いて、部門サーバのAPP監視マネージャから、共有回線を使用したネットワークAPPの名前、実行時間、集計部門サーバの位置を取

Traffic Management Methods  
based on end-user multimedia information access  
Youchi Hidaka, Yoshimi Teshigawara  
Graduate school of Engineering, Soka University

得し、TMSの実行APP蓄積データベースに格納する。また、定期的に、共有回線を使用するパケットを、トラフィック収集マネージャで、プローブで収集したデータのPORT番号、データ量、時間、要求位置などを、RMON2<sup>[1]</sup>を用いて回収し、TMSのトラフィックデータベースに格納する。そして、参照データベースとして、APP名、使用PORT番号などをパラメータとして持つ、以下に示す、APP帯域要求レベルデータベースを参照し、実行APP蓄積データベースとトラフィックデータベースを解析し、最適帯域割り当てアルゴリズムSを用いて、部門サーバの利用可能帯域幅を設定する。

また、部門サーバから、帯域不足の要求があれば、帯域幅監視マネージャを用いて、他の部門サーバの利用帯域幅を集計し、再度、TMSで検討し、もし、利用可能帯域幅の変更が可能なら、利用可能帯域幅の変更及び、他の部門サーバの利用可能帯域幅も変更する。

3.1 APP要求レベルデータベース

APP要求レベルデータベースは、APP名、PORT番号、要求最低帯域幅、要求最高帯域幅、リアルタイム要求度、APP優先度のパラメータを持っている。つまり、LANで使用されるネットワークAPPが、どの程度帯域を必要とするのか、どのような帯域を必要とするのか、他のAPPに比べてどの程度の優先度があるかの情報である。このデータベースを参照することで、トラフィックデータベースのPORT番号の対応づけができるものは行い、最適割り当てアルゴリズムの一つの有用なパラメータとして用いたり、最適割り当てアルゴリズムの検討に用いる。

しかし、ここで問題となってくるのが、パラメータに対して、ユーザのマルチメディア情報に対する評価尺度が関係してくるので、完全な答えを求める方法が、現在、存在しないということである。また、ネットワークアプリケーションは時々刻々と新しいものが出

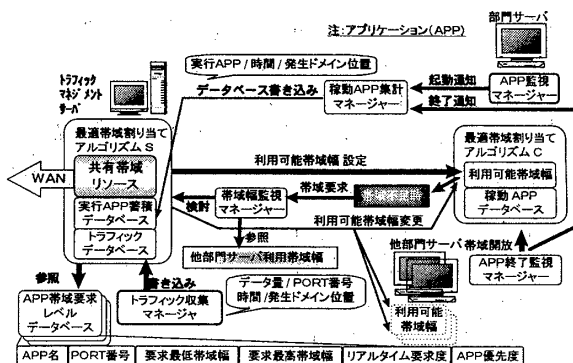


図1. トラフィックマネジメントサーバと部門サーバ

てくるので、これらのパラメータを完全な定式化しなければ、柔軟に対応していくことができないという点である。

研究では、これらのパラメータを、シミュレーションおよび実験で求め、それを採用することを考える。

### 3.2 最適割り当てアルゴリズムS

最適割り当てアルゴリズムSは、主に、共有回線の帯域幅(C)、LANのクライアントが実行したAPPの蓄積データベース(CApp)、また、共有回線を使用したデータのトラフィックデータベース(Td)を基に、前述のAPP帯域要求レベルデータベース(AppL)を用いて評価し、資源配分関数(Fs)で、LAN全体のリソースの効率化が最大になるように、部門サーバに利用可能帯域幅割り当てる。これらのパラメータの関係は式で表すと、

$$Fs = f(C, AppL, CApp, Td)$$

AppL: トラフィック評価関数

となる。

### 4. 部門サーバとクライアント

部門サーバは、図2に示すように、まず、APP監視マネージャおよびAPP終了監視マネージャを用いてクライアントのネットワークAPPの稼動状況を把握する。そして、それらの情報を基に、TMSから割り当てられた利用可能帯域幅を、最適割り当てアルゴリズムCを用いて、担当クライアントのユーザの満足度の総計が最大になるように、ダイナミックに帯域を割り当てる。また、TMSから割り当てられた利用可能帯域幅が過不足状態になった場合は、TMSに帯域要求を試みる。一方、クライアントは、APP監視エージェント、APP待機エージェント、APP実行監視エージェント、帯域幅開放エージェントの4つのエージェントを用いて、部門サーバと連携し、使用する帯域幅の調整を逐次行う。以下に、それぞれのエージェントの役割を示す。

- 1) APP監視エージェント  
稼動するネットワークAPPを監視し、部門サーバのAPP監視マネージャに通知する。
- 2) APP待機エージェント  
部門サーバから利用可能帯域幅を受け取り、クライアントの利用可能帯域幅の変更する。
- 3) APP実行監視エージェント  
担当のネットワークAPPの実行状態を監視し、もし、部門サーバから帯域変更の要求があれば利用可能帯域幅の変更を行う。
- 4) 帯域幅開放エージェント  
担当APPの終了を部門サーバのAPP終了監視マネージャに通知し、そのAPPが使用していた帯域幅の開放を行う。

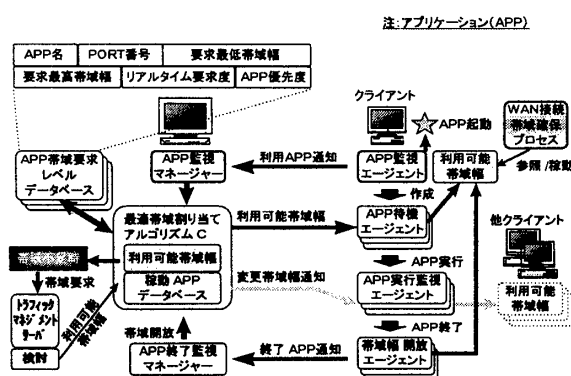


図2. 部門サーバとクライアントの関係

### 4.1 最適割り当てアルゴリズムC

最適割り当てアルゴリズムCは、主に、クライアントの稼動APPデータベース(Ad)と、TMSから割り当てられた利用可能帯域幅(Cb)を基に、APP帯域要求レベルデータベース(AppL)と、レスポンスタイムなどを考慮したユーザの満足度を要素とする関数(U)を用いて、担当クライアントのユーザの満足度の総計が最大になるように割り当てを行う。このための資源配分関数(Fc)は、次のように表わされる。

$$Fc = f(U, Cb, AppL, Ad)$$

### 5. 帯域制御とネットワークAPP利用状況の取得

帯域制御に関しては、LANのプロトコルは、TCP/IP環境を標準とするので、将来、有望視されているIETFのRSVP<sup>[2]</sup>を用いる。また、ネットワークAPP利用状況の取得に関しては、利用者の繊細な情報を得ることができ、今後のネットワーク管理に欠かせなくなりつつあるDMTFのDMI<sup>[3]</sup>を用いて取得することを考える。

### 6. おわりに

今回のシステムは、シミュレーションでの評価は可能であるが、実際、実装して評価するとすると、帯域確保に用いるRSVPの改良、およびTCP/IP以外のプロトコル及びDMI不対応クライアントの考慮などが問題となってくる。今後は、これらの問題点を解決し、実装を目標としたシステムの設計を行い、評価を行っていきたい。

### 参考文献

- [1] Internet Engineering Task Force (RFC 1757)  
"Remote Network Monitoring Management Information Base version 2"  
Internet Draft, 27 May 1996
- [2] Internet Engineering Task Force  
"RSVP Version 1 Functional Specification"  
Internet Draft, July 1995
- [3] Desktop Management Task Force  
"Desktop Management Interface Specification"  
Version 2.0, 27 March 1996