

RSVP を用いた動画アプリケーションについて

後藤 幸功[†] 長野 央[‡] 荒木 啓二郎[†]

[†]九州大学 大学院システム情報科学研究科, (財)九州システム情報技術研究所
[‡]有限会社ドットアスタ

概要

インターネットを使用した動画アプリケーションは、品質が保証された通信を必要とするため RSVP などを用いて資源予約を行なう必要がある。しかし、RSVP と Integrated Service で規定されている品質のパラメータはユーザが動画アプリケーションが要求する品質パラメータと異なるため、動画アプリケーションは RSVP を使用して要求するパラメータとは他にユーザから動画アプリケーションに要求する品質パラメータの値を送信者に送る必要がある。本稿では、ユーザが動画アプリケーションに対して要求するパラメータを考察し、RSVP を用いて資源予約可能な動画アプリケーションを実装するための機能のモデルを提案した。

Study of Movie Application using RSVP

Yukinori Goto[†], Nakaba Nagano[‡] and Keiji Araki[†]

[†] Graduate School of Information Science and Electrical Engineering, Kyushu University,
Institute of Systems & Information Technologies/Kyushu
[‡] Dot Aster Co.,ltd.

abstract

Movie applications on the internet need to use RSVP (Resource Reservation Protocol) which is one of reservation protocol on the internet to guarantee of QoS. However, QoS parameters which are described in drafts of RSVP and Integrated Service by IETF differ with QoS parameters which are required from users to movie application. Therefore it is need the system which send QoS parameters required user to the movie server beside RSVP. In this paper, we discuss QoS parameters which are required by users and proposed the model of system for the implementation of movie application which guarantees QoS using RSVP.

1 はじめに

インターネットの広帯域化に伴い、インターネットを用いた音声や動画の実時間通信の可能性が広がってきた。しかし、音声や動画の実時間通信を実現するためには帯域は遅延時間などの様々な品質が保証されなければならない。が、IP プロトコルはそのプロトコル自身に品質保証の機能がない。そこで、インターネット上で予め資源を予約することにより品質を保証する方法が提案され、資源を予約するためのプロトコルと

して RSVP(Resource ReSerVation Protocol)[1] が開発された。現在 RSVP は RFC(Request For Comments) として標準化が行なわれている [2]。

しかし、実際のアプリケーションは、ネットワークの詳細知識を用いないユーザに対しても容易に操作可能とするため、図 1 のようなアプリケーションとユーザ(人間)との間にインターフェースを介してユーザからの要求を取得し、その要求に従ってアプリケーションが RSVP のインターフェースへネットワーク資源予約するモデルに従って品質の保証を行なう。しかし、

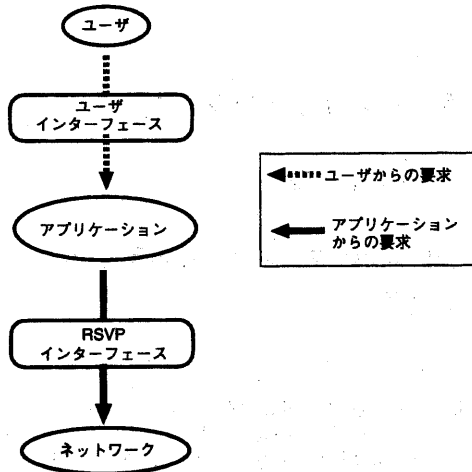


図 1: アプリケーションのためのクライアント側システムモデル

資源予約プロトコル RSVP がネットワーク層で予約するために、アプリケーションから要求する品質パラメータは帯域や遅延時間である。一方アプリケーションがユーザ(人)から要求するパラメータは、圧縮方式や画像の大きさ等であり、ネットワークの状態をユーザに対して意識させないものである。従って RSVP がアプリケーションに求める品質パラメータと異なる場合がある。

本研究では、動画画像アプリケーションに注目し、このアプリケーションに対し、ユーザからの要求とアプリケーションから RSVP に対する要求への変換について述べるとともに、RSVP を用いた動画画像アプリケーションの構築モデルを提案する。

2 資源予約

2.1 RSVP(Resource ReSerVation Protocol)

資源予約プロトコル RSVP は、受信者が送信者に向かう経路上に存在するルータの資源を予約することによって、受信者が要求する品質を保証するためのプロトコルである。まず、送信者は Path_msg[2] の SENDER_Tspec [2] を用い受信者と受信者までの間に存在するルータに対し、送信者が提供するフローの

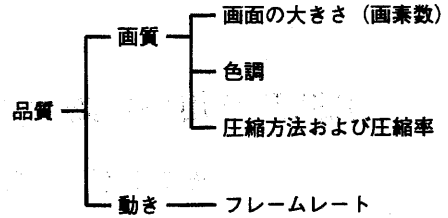


図 2: 動画像のための品質パラメータ

品質を通知する。各ルータは送信者から送られてくる SENDER_Tspec にしたがってネットワーク資源の確認を行なう。そして、受信者は送信者から送られてきた SENDER_Tspec に従うかまたは、受信者が希望するフローの品質を FLOWSPEC[2] に記述し Resv_msg[2] を用いて送信者側に向けて送り、Resv_msg を受信したルータは Resv_msg に記述された FLOWSPEC に従って資源予約を行ない受信者が希望したフローに対する品質を保証する。

このとき SENDER_Tspec や FLOWSPEC に書かれた品質の記述は、参考文献 [3, 4, 5, 6] で規定されており、帯域や送信レートなどである。これは、ユーザとアプリケーションとの間で交換される画面の大きさ、圧縮方式、圧縮率などのパラメータと全く異なる。ユーザとアプリケーションとの間で交換される品質のパラメータについては3節で述べる。

3 動画アプリケーションの品質

この章では、動画アプリケーションがユーザに対して提供する品質パラメータについて考察し、本論文で使用するパラメータを定義する。

3.1 品質パラメータ

動画画像アプリケーションとユーザの間にある品質を求めるパラメータとして帯域や遅延時間、ゆらぎなどを直接指示することも可能である。しかし、この場合ユーザはネットワークの知識を十分に持たなければならぬ。一般にユーザが画像に関して持つ知識としては、経験的に画質と動きの2つである。画質に関して細かく考察すると、画面の大きさ、色調、圧縮方式と圧縮率に分けられ、動きに関してはフレームレートが考えられる(図2)。

そこで、本研究ではユーザから取得可能な品質要求をこの4つに限定する。これら4つの情報から、RSVPの FLOWSPECが必要とするパラメータは簡単な式によって得ることが可能である。例えば、圧縮方式がMPEG1の場合、使用帯域は1Mbps~1.5Mbpsと規定されているので、FLOWSPECにはこのように記述すれば良い。また、ppmの画像やJPEGの圧縮方式を用いた静止画を送ることによって動画とする場合は、
$$\text{帯域 (bps)} = (\text{縦} \times \text{横}) (\text{bit}) \times \text{色調 (bit)} \times \text{フレームレート (f/s)} [\times \text{圧縮率 (\%)}]$$
で求めることが可能である。

3.2 ユーザインターフェース

本節では、前節で述べた品質パラメータをユーザからアプリケーションへ伝えるためのアプリケーション側のインターフェースについて考察する。ここでは、送信者の機能などは考慮せず、純粋に動画アプリケーションの表示機能に対して受信者が品質を求める場合についてのみ考える。

1. 画面の大きさ(画素数)

ユーザが画面のサイズを選択する方法としては2つある。1つは、アプリケーションが既に用意している選択肢から選ぶ方式である。もう一方はユーザが縦と横の大きさを数値で指定するか、または画像が表示されるまたはされているウィンドウ画面の大きさを直接変化させることでアプリケーションへ通知する方法である。

2. 色調

デジタル動画における色調は色の階調をビット数で示している。通常ディスプレイの色調以上に色調を求めると画面上で色落ちがおこる。よって自分のディスプレイが表示できる以上の色調を求めないようにすべきである。ただし、自分のディスプレイが表示可能な色調以下も求めることが可能であってもよい。従って、通常コンピュータにおいて示される基本的な色調である、2bit, 4bit, 8bit, 16bit, 24bitの色調から選択可能な方式があげられる。

3. 圧縮方式および圧縮率

圧縮方式は様々存在する。従って、サービスが提供されている圧縮方式をボタンによって選択できるようにする方式があげられる。圧縮方式には固定的に圧縮率が決められたものも存在するが、

JPEGのように圧縮率をユーザが指定できるものもある。ユーザが圧縮率を指定できる圧縮方法を指定したときは、圧縮率を記述できるインターフェースを用意する。このインターフェースには数値を入れる方法と、レンジを表示して視覚的に調節可能とする方法がある。

4. フレームレート

このインターフェースとしては正数値を入力する方式と、レンジを使用して入力する方式がある。

4 動画通信アプリケーションのモデル

本論文では送信者と受信者間で使用するアプリケーションの形態はサーバ・クライアントとし、使用される動画はカメラから取得した動画を実時間内にネットワークへ放送するテレビ放送の形態について検討する。

RSVPの設計思想に基づく場合、受信者が送信者に対して品質を要求できる。インターネット上の受信者は様々なネットワークを介して接続されているため、それぞれ異なった品質を要求する可能性が少なくない。しかし、送信者はインターネット上の受信者のネットワーク状況を全て把握することは不可能である。そこで、送信者が出来る限り受信者の要望に応じるためには、複数の動画品質、すなわち3章で示した画面の大きさや、色調などの選択肢を増やす必要がある。

これらを考慮して次に受信者側のアプリケーションが送信者から送られてくるサービス内容に従ってユーザへ品質を問い合わせ、ユーザから要求される品質を送信者とRSVPに伝え、送信者と受信者までの通信が行なわれるまでの段階について考察する。

まず、受信者は送信者がどのようなサービスを提供しているか分からなければ品質要求を行なうことが出来ない。送信者が提供するサービスの情報として送信者が使用するフローの性質、つまり帯域や送信レートがあるが、この情報についてはRSVPのSENDER_Tspecによって受信者に伝えることが可能である。しかし、SENDER_Tspecによって通知された情報だけでは、受信者は送信者が提供する画面の大きさや色調、フレームレートについて知ることが出来ない。従って、RSVPのSENDER_Tspecを用いずに他の方法によって送信者が提供する動画に関する品質情報を通知しなければならない。

次に、受信側のアプリケーションはネットワークおよびシステムの状況からユーザに提供可能なものを選択しユーザへ提示する。ユーザは提示された品質情報から適当なものを選択し、送信者へ提示するとともに RSVP 機構に対してネットワーク資源の要求を行なう。RSVP によって資源予約可能な場合は、送信者から動画情報が送られてくる。

もし、ユーザが求める品質が高いために、RSVP による資源予約が失敗した場合は、アプリケーションはユーザに対し、提供可能な品質情報を再度選択しユーザに対し提供する。この繰り返しにより、ユーザは自分が受けられる品質の画像をを限定できるようになる。ユーザは限定された品質からユーザが許容できる品質を選び送信者へ通知し品質保証されや動画情報を受信する。

4.1 モデル化

以上述べたような機能を実現するために次のようなアプリケーションモデル(図3)を提案する。

図の機能の詳細は次の通りである。

送信者側の品質情報管理部

送信者側の品質情報管理部では、送信者が提供する動画の品質を管理し、その情報を受信者へ通知する。また、受信者側から通知された品質情報を受信し、その情報を送出部へ通知する。

受信者側の品質情報管理部

受信者側の品質情報管理部では送信者から送られてくる品質情報を保管し品質制御部に通知するとともに、品質制御部を通してユーザから選択された品質情報を送信者側情報管理部へ通知する。

品質制御部

品質制御部には2つの機能から成り立つ。1つは、品質情報管理部が受けた情報に対し、システムの資源管理部から伝えられる受信可能な資源の情報を考慮し、システムが受信可能な品質情報のみ選定し、選定した品質情報のみユーザインターフェース部に通知する、品質情報選定機能である。もう1つは、情報変換機能であり、ユーザインターフェースから送られてきた情報を RSVP が理解できる情報へ変換する。

ユーザインターフェース部

ユーザインターフェース部は選定された品質情報

を元にユーザへ品質の問い合わせを行なうためのインターフェースを表示する。そして、ユーザから選択された品質情報を品質制御部へ通知する。

画像表示部

画像表示部は、受信した画像情報を品質制御部から通知された情報に従って表示する。

RSVP 部

品質制御部で変換されたユーザからの品質情報を元に資源予約を行なう。

資源管理部

システムが提供可能なシステムの資源状況を管理する。品質制御部からの問い合わせにより、そのときのシステムの資源状況を品質制御部に通知する。

送出部

品質情報管理部から通知された品質と RSVP から通知された品質に従って動画情報を送出する。この部分において、送信側が準備している画像の圧縮方式や画素数などの品質を提供するモジュールが組み込まれており、品質情報管理部へ提供可能な品質情報を通知する。

5 実装と評価

現在、送信者と受信者間をユニキャストで接続するサーバ・クライアントモデルで実装中である。サーバが提供する品質については次のようなものを準備している。画像の圧縮方式については、無圧縮方式の ppm を一定間隔で転送する方式と、Motion JPEG を実装した。画像の大きさに関しては、画像の大きさが横 640 ドット、縦 480 ドットを上限とし、受信者が 1 人の場合は送信者側でユーザが指定する大きさで画像を取得する方式をとるが、1 人以上からはユーザに対し、横×縦の品質情報を 640×480、320×240、160×120 の固定値から選択するように設計した。フレームレートも同様に、受信者が 1 人の場合はユーザが指定するフレームレートを提供したが、2 人目以降は要求が 1 人目の要求より多きときは、送信者側で画像の取り込むフレームレートを大きくし、要求が小さい場合は、送出時に情報を廃棄するように設計した。色調に関しては、固定的に 16bit 階調のみを提供するよう設計した。

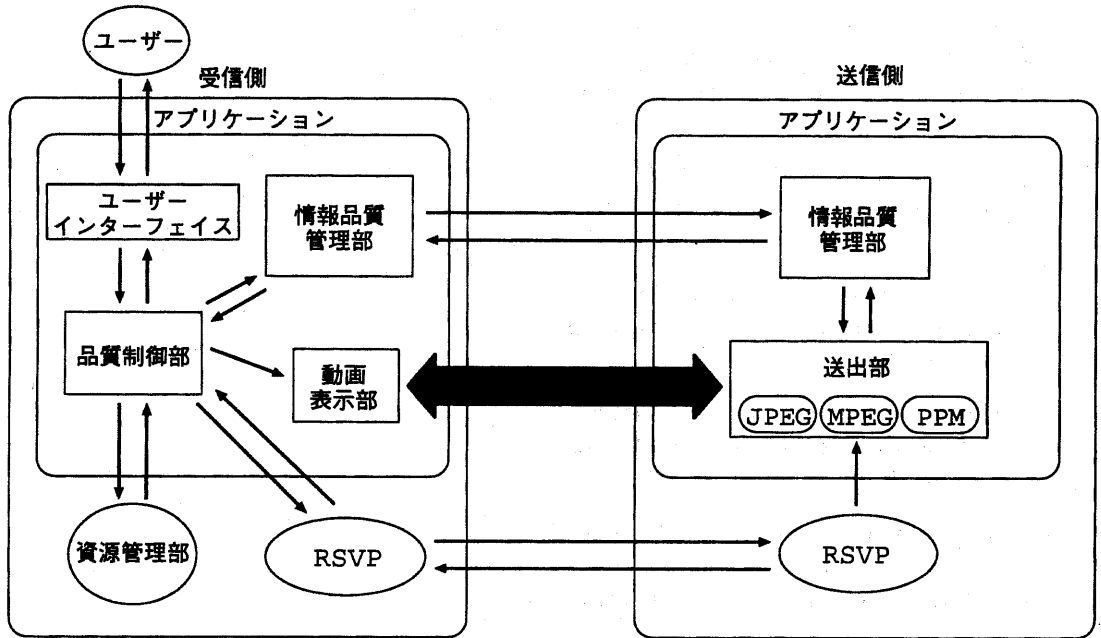


図 3: 提案するモデル

評価に関しては受信数や、異なる要求に対応する頑強性について評価中であるとともに、評価方式について検討中である。

6 まとめ

今回の研究では、RSVP を用いて資源予約可能な動画アプリケーションを設計するための基礎である、ユーザーとアプリケーション間の品質パラメータについて考察し、資源予約プロトコル RSVP が提示する品質パラメータと異なることを示した。そして、RSVP を用いて資源予約可能な動画アプリケーションを実装するための機能をモデル化し提案した。現在、提案したモデルに従ったアプリケーションとシステムを構築中である。

7 今後の課題

今回の実装ではユニキャスト通信によるクライアント・サーバモデルを用いた。そのため受信者が増加した場合、動画を送るためのコネクション数が増加し、多

人数に対して対応できない。従って、今後は IP マルチキャストを使用した実装を考慮する必要がある。しかし、IP マルチキャストを使用した場合は、異なったユーザの要求に対し、どのように送出部で実現するかが問題となる。

8 謝辞

本研究を推めるにあたり、議論、助言をしていただいた京都大学の藤川賢治氏、九州大学工学部の吉村康彦氏、鶴久康治氏、有限会社ドットアスタの田中典翁氏、財団法人九州システム情報技術研究所の皆様様に心より感謝致します。また、本論文の作成にあたり協力していただいた谷川恭子女史に感謝します。

参考文献

- [1] L. Zhang, S. Deering, D. Estrin, S. Shenker, D. Zappala. *RSVP: A New Resource ReSerVation Protocol*, IEEE Network, Vol.7, No.5, September 1993.

- [2] R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin *Resource ReSerVation Protocol (RSVP) Version 1 Functional Specification*, Request for Comments: 2205, Sep.1997
- [3] J. Wroclawski *The Use of RSVP with IETF Integrated Services*, Request for Comments: 2210, Sep.1997
- [4] J. Wroclawski *Specification of the Controlled-Load Network Element Service*, Request for Comments: 2211, Sep.1997
- [5] S. Shenker, C. Partridge, R. Guerin *Specification of Guaranteed Quality of Service*, Request for Comments: 2212, Sep.1997
- [6] S. Shenker, J. Wroclawski *General Characterization Parameters for Integrated Service Network Elements*, Request for Comments: 2215, Sep.1997