

UNIXワークステーション上でのMPEG2デコード機能の実現

3U-6

成田浩志 村山正之

(株)東芝 府中工場 電算機ソフトウェア部

0. はじめに

当社はVOD(ビデオ・オン・デマンド)システムとして、映像の供給を行うVODサーバ、映像の表示を行うVODクライアント、および運用管理ソフトウェアを開発し提供している。この開発の一環として、ATMネットワークを使用したVODシステムのクライアント側で使用するMPEG2デコード機能(MP@MLクラスをサポートする)をUNIXワークステーション(Sun社SPARCstation)上に開発したので報告する。

1. システム構成

本MPEG2デコード機能のハードウェア構成は、UNIXワークステーションに、VODサーバからビデオデータを受信するためのATMアダプタ、MPEG2 PES形式データをデコードするためのS Busカードを実装したものである。(図1 参照)

ビデオデータは、ATMのVC(virtual circuit)を通じてMPEG2 TS(Transport Stream)形式で配信され、unix上のプロセスとして実現されたプログラムが受信する。この受信プロセス内でTSパケットをPESパケットに組み上げた後にMPEG2デコーダカードにPESパケットを入力して動画像をNTSC信号として再生する。

このように汎用プラットフォーム上で実現することにより、他ソフトウェアとの組み合わせが可能になる。たとえば動画をウィンドウシステムの一つのウィンドウとして表示することが必要な場合は、ビデオウィンドウ機能を持ったフレームバッファと組み合わせることで実現できる。

しかし、このような構成にてシステムを稼働させるには、以下に挙げるようなunix (solaris)に依存する問題があった。

2. 課題

(1)カーネルのオーバーヘッド

本システムが想定するMPEG2 TSのデータ転送レートは4Mbit/sec - 8Mbit/secであるが、TSのパケット長は188*2バイトと短いため1秒間に約1400-2800パケットを受信することになる。受信プロセスが各パケットのread待ちでblockしコンテキストスイッチを起こしていたのでは、カーネルのオーバーヘッドが非常に大きくなると考えられる。

他のソフトウェアと協調して動作させることも考えると、本システムが消費するCPU時間をできるだけ少なくする必要がある。

(2)プロセススケジュールの遅れ

従来unix上でのプロセスのスケジューリングは、タイムシェアリングシステムを効率的に動作させることを目的にしているの、長時間cpuを占有したプロセスに対しては、スケジューリングの優先度を下げる。このため、受信プロセスは実行時間が経つとともに優先度が下がる可能性があり、MPEG2デコーダカードへ表示すべきMPEG2データの供給が追いつかなくなり、画像の異常が発生する原因となる。

このような事態の回避策として従来行われているのは、MPEG2デコーダカードへの入力バッファ量を大き

An Implementation of MPEG2 Decoding System on UNIX workstation

Hiroshi Narita, Masayuki Murayama
Computer Software Department, Fuchu works,
Toshiba
Fuchu city Tokyo Japan

unixは、X/Openカンパニーリミテッドが独占的にライセンスしている米国ならびに他の国における商標です。

Sunは米国における米国Sun Microsystems社の商標です。SPARCは米国におけるSPARCInternational, Incの登録商標です。SPARC商標のついた製品は米国Sun Microsystems社が開発したアーキテクチャに基づくものです。

Solarisは米国における米国Sun Microsystems, Incの登録商標です。

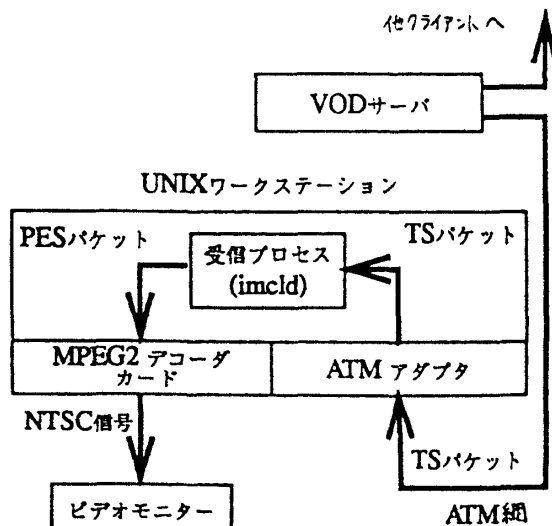


図1 VODクライアントのシステム構成

くして、UNIXのプロセススケジューリングの遅れによるデータの枯渇が発生しないようにする事である。

しかしながら、これを行なうと映像スタート時の遅延が大きくなる。VODシステムは人間が操作するので映像スタート時の遅延を小さくする必要があり、操作に対する高速な応答を要求される。受信バッファを大きくするわけにはいかない。

3. 解決策

(1)カーネルのオーバーヘッド低減化

本システムで使用しているATMアダプタではunixのstreams機構を用いてATMのVCとのread/writeが実現されている。Streams内に受信パケットを溜めることができるので、間欠的に複数のパケットをまとめて読みだせば、ATMからのTSパケット受信待ちによるblock回数を低減させる事ができる。

(2)プロセススケジューリングの遅れの抑制

unixにて実現されているRT(リアルタイム)クラスを用いる。デフォルトのTS(タイムシェア)クラスではプロセスの優先度をカーネルが自動的に調整するのに対し、このRTクラスでは固定優先度である。

したがって受信プロセスを一番高いスケジューリング優先度に固定的に割り当てることにより、他のプロセスの実行の影響を受けて受信プロセスのスケジューリングが不安定になることを回避できる。

4.測定結果のまとめ

(1)測定環境

以下の環境で測定を行なった。

ハードウェア SPARCstation 20

(SuperSPARC 60MHz+SuperCache)

OS solaris2.4

ATMアダプタ

Sun ATM-155/Mfiber Adapter

MPEG2デコーダカード：当社製

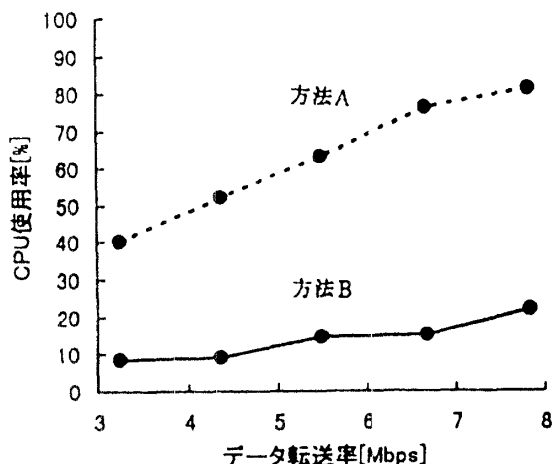


図2 VODクライアントのCPU使用率

(2)カーネルオーバーヘッドの低減

ATMのVCからのTSパケットを単にreadする方法A(read時にパケットがなかったらblockする)と、10ms待ってからカーネル内部に受信されているパケットをすべてreadすることを繰り返す方法Bの2つについて、MPEG2動画再生中のCPU使用率とコンテキストスイッチ回数を測定した。結果を図2と図3に示す。

後者はコンテキストスイッチの回数が抑えられており、CPU使用率が低いことが分かった。

(3)プロセススケジューリングの遅れの抑制

本システム稼働中にCPU時間を消費し続けるTSクラスのダミープロセスを動作させた。

これを一定時間実施し、MPEG2デコーダハードウェアのバッファの枯渇の有無を調べた。受信プロセスをTSクラスで実行した場合にはMPEG2デコーダハードウェアの入力バッファの枯渇が発生したが、RTクラスで実行した場合には発生しなかった。

5. おわりに

MPEG2 TSパケット受信をunix上のプロセスで行う場合のカーネルのオーバーヘッドはパケット受信待ちによるコンテキストスイッチであった。これはパケットがカーネル内部にたまるのを待ってからまとめてreadすることで回避でき、CPU使用率の低減に効果があることが分かった。またsolarisではリアルタイムクラスで実行することによりプロセス実行スケジュールのばらつきを抑制できることが分かった。

以上により、汎用のUNIXワークステーションをプラットフォームとして他のソフトウェア等と組み合わせ使用可能なMPEG2デコード機能を実現することができた。

[参考文献]

- [1] SunATM-155 SBus cards Manual, Sun Microsystems Computer Company, 1995

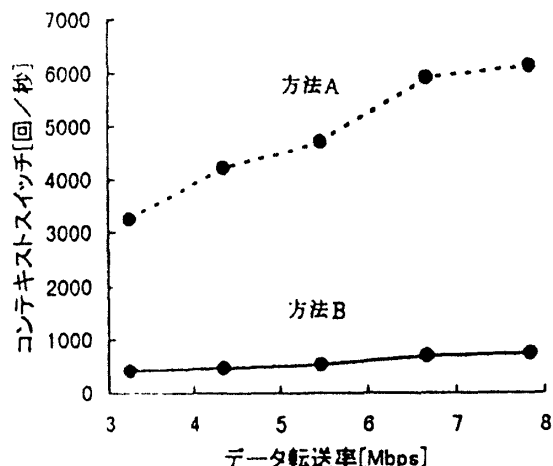


図3 VODクライアントのコンテキストスイッチ数