

6C-2

分散環境における音声映像のレートミスマッチの 検知訂正方式の提案*

河内 清人 田中 賢一郎 米田 健 洪 忠善 松下 温†

慶應義塾大学理工学部‡

1 はじめに

近い将来に、ネットワークの大容量、高速化にともない、通信データの種別は単純なデータファイルに、音声や映像などのコンティニユアスメディア(CM)情報が加わってくる。CM情報の転送には、レートフローコントロールの行なえるプロトコルが必要である。このようなプロトコルは様々なものが提唱されているが、そのレートを送受信間で一意にするための方法は出ていない。本論文では、レートの不一致によって生じる問題を指摘し、その解決法を述べる。

2 クロック訂正の必要性

CMデータはレートフローコントロール方式によって送信されることが考えられる。また、データのストア時にはサンプリングレートが問題になってくる。ここではこれらのレートを守ることを意味を述べていく。

2.1 レートを守ることの必要性

CMデータの通信において、レートが関係しているのは、大きく分けて、通信時のレートと、データ生成時のレートである。まとめると、次のようになる。

- 通信時のレート... 単位時間に、何個のframeを送るか。
- データ生成時のレート... 連続量である元のデータを、どのくらいのサンプリングレートで離散化するか。

通信時のレートが守られないと、再生ホストのバッファ内の蓄積frame量が不安定になってしまう。

データ生成時のレートが守られないと、そのデータを再生した時に、正しい速度で再生することができなく

なってしまう。このことは同期の維持が不可能であることを意味する。

このように、CMデータの通信を行う際にはレートを正確に守る方法が必要である。

しかし、そのためには、ホスト間のクロックのズレの問題を克服しなければならない。なぜなら、ホスト間でクロックがずれていると、それら間で共通のレートというものが決定できなくなるからである。

今回、相手から送られてくるframeの平均到着間隔を測定することで、ホスト間のクロックのずれを効率的に検知、訂正する方法を提案する。

3 レート訂正のアルゴリズム

今回のアルゴリズムでは次のように、レートの検知、訂正が行われる。

アルゴリズムを説明するにあたって、次のように定義をする。

- frame番号は0からの通し番号である。
- $F_i, L_i \dots i$ 番目の訂正を行うために参照したframeの内、最初のもの最後のもののframe番号、ただし、 $F_0 = 0$
- $\text{Time}(i) \dots i$ 番のframeが届いた時刻。ただし、 $\text{Time}(0)$ を基準点0とする。
- $t_i \dots i$ 番目の訂正を行う時刻。ただし、 $t_0 = 0$

このように文字を定義すると、アルゴリズムは次のように簡潔に表現できる。

$$F_i = L_{i-1} + pad \quad (1)$$

$$t_i = \text{Time}(F_i) + \text{Cons}\gamma^{i-1} \quad (2)$$

ここでConsはある定数値、 γ はその巾乗係数である。つまり、観測期間は一回毎に γ 倍されるわけである。これにより誤差の判定精度を保ったまま、大きな誤差の時訂正を行なうことができるようになる。計算で出した最適値は $\gamma = 3$ である。Consの値は、相手のクロックのずれがどの程度のものかによって異ってくる。

*A Study of Detecting and Correcting Rate Mismatches on Audio/Visual Data in Distributed Environment

†Kiyoto KAWAUCHI, Kenichiro TANAKA, Takeshi YONEDA, Choongseon HONG, Yutaka MATSUSHITA

‡Faculty of Science & Technology, Keio University

pad で表されるだけの frame を測定対象にしないのは、レート訂正後しばらくの間は前回のレートの frame が到着するからである。

これらの概略図を figure3.1 に示した。

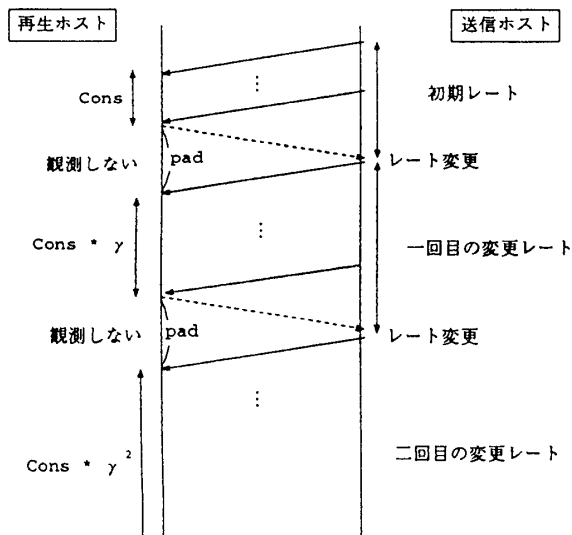


Figure 3.1: レート訂正の概略図

4 シミュレーション

前述のアルゴリズムを用いた場合のレート訂正の様子をシミュレートした。このとき、パラメータは次のように設定した。

- レートの交渉値: 30frame/sec
- 送信ホストの初期レート: 交渉値にその5%の範囲のずれを乱数で加えた。
- ネットワークの最大遅延: 80msec
- 再生ホストの時刻の測定精度: 1msec
- 送信ホストのレート変更精度: 0.001msec
- シミュレーション期間: 4000sec

なお、再生ホストの観測期間、 $Cons \times \gamma^{i-1}$ は、 $Cons = 1(\text{sec})$ 、 $\gamma = 3$ とした。さらに、*pad* は 10frame に設定してある。

5 結果

上記のシミュレーションによって得られた、再生ホスト内のバッファの frame 蓄積量の変化の様子を次の figure.5.1 に示した。この結果より、今回提案したアルゴ

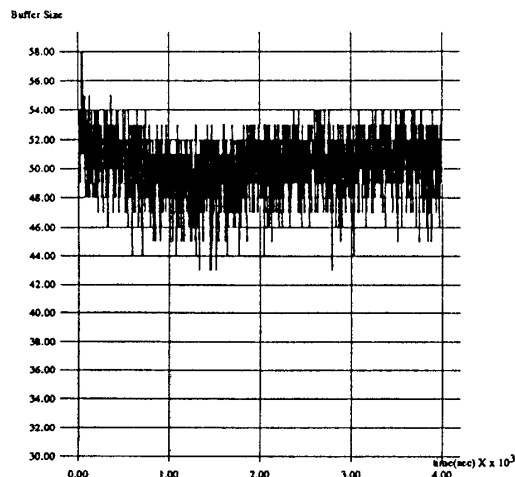


Figure 5.1: バッファ内 frame 蓄積量の変化の様子

リズムは、バッファの値を精密に維持できることが示された。

6 結論

今回、到着間隔を測定することで相手のクロックのずれを検知、訂正する方法を採用することで、十分短い時間で必要な精度を得られることがわかった。ホスト間のクロックを同期する目的のものに NTP[1] があるが、今回の方法は、CM 通信時をその主眼とすることにより、NTP よりも高速に、また、低いコストでクロックの同期を得ることができた。今後は、バッファ内の frame 蓄積量などの情報から、不必要な訂正を減らす方法を研究していく予定である。

参考文献

- [1] David L.Mills. Network Time Protocol(Version 2) Specification and Implementation. Technical Report Request for comments: 1119, Network Working Group, September 1989.