システムソフトウェアと
オペレーティング・システム　64－11
マルチメディア通信と分散処理　65－11
（１９９４．　５．　１９）

会話型操作を伴う
マルチメディアシステムにおける同期方式

矢野司　寺西祐人　大野隆一　相田仁　齊藤忠夫

東京大学　工学部

各種のマルチメディアシステムは，複数のメディアを扱うといったその性質上，メディア間の同期制御を必要とする．従来の同期方式の多くは，各メディアのデータを多重化する際に同期制御情報を付加するものであるが，その応用先や状況によってはこの多重化は必ずしも適当ではない．

本稿では，既に存在する動画に対して新たに音声を同期付けする際に，多重化を行うことなくこれを実現する方式を提案する．さらに，こうして同期付けされたマルチメディア情報に対して早送りや巻き戻しといった会話型操作が加えられた場合の同期方式についてもいくつかの案を提示し，その一部に関して試作システムの実装を行いこれを評価した．

A Synchronization Method for a Multimedia System with Interactive Control

Tsukasa YANO, Masato TERANISHI, Ryuichi OHNO, Hitoshi AIDA, and Tadao SAITO

Faculty of Engineering, The University of Tokyo

Most multimedia systems implement synchronization between media streams by multiplexing those streams. However, this is not suitable for some applications.

In this paper, a method to synchronize video and voice data without multiplexing is presented. This method is further extensible for cases in which interactive control such as 'Fast Forward' or 'Reverse' is applied to the media. Part of the system has been implemented on a workstation, and evaluated.

1　はじめに

近年のハードウェアの性能向上やその低価格化を一因とし，パソコンやワークステーション上にマルチメディア機能を追加する傾向が強まっている．こうして実現される種々のマルチメディアシステムによって，複数のメディアの同時利用や，あるいは必要に応じて適当なメディアを選択することが可能となる．これは情報伝達時における理解度を著しく向上させうるものであるが，それと同時に，複数のメディアを扱うことにともなったいくつかの問題も生じさせるものであり，そのひとつとしてメディア間の同期が挙げられる．

現在，各プラットホーム上で採用されている同期方式の多くは，MPEGシステム・パートでの規定（図１）に代表されるように，各メディアの符号化データや同期制御情報を多重化するものである．しかし，例えばネットワークを介して配送されるビデオデータに対して，ユーザが音声による独自の解説を加えるといった場合には，同期のためにこれらを多重化し直すことは煩雑な作業であろうし，そもそも配送されるデータの改変が不可であることもありうる．このように，各メディアの符号化データをそれぞれ異なったデータストリームを介して送受信したり，あるいはこれを蓄積するといった場合や，ユーザが各種メディアの組み合わせを選択的に利用する場合など，その用途や状況によっては多重化が隘路となる．

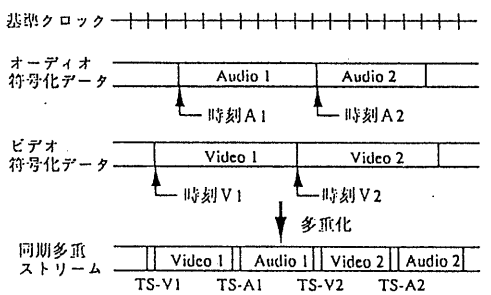

図１　同期多重化方式（ＭＰＥＧシステム・パート）

そこで本稿では，想定する環境を限定した上で，既に存在する動画に対して新たに多重化を行うことなく，音声を同期付けする方式を提案する．さらに本方式によって同期付けられたマルチメディア情報に対して，早送りや巻き戻しといった会話型操作が，その情報提示の際に加えられた場合の同期方式についてもいくつかの案を示し，その一部に関して試作システムの実装を行ったのでこれを報告する．

2　多重化されていないメディア間の同期方式

2．1　システムの概要

１章で述べた通り，マルチメディアシステムのさらなる応用先や，将来的なネットワークの形態を考慮した場合，多重化されていないメディア間の同期制御が必要となる．本稿では，その対象メディアを動画と音声とに限定し，特に，

“既に存在する動画に対して，新たに音声による解説等を同期付けする”

といった状況を想定した．これは，インタラクティブムービーやプレゼンテーションなどの一般的なマルチメディアシステムにおいて，音声が動画に付随的に再生される状況が多いことに起因する．

またこの動画に関しては，そのデータが同一ノードに蓄積されているものであるか，あるいは他のノードから順次転送されてくるものであるかなどといった形態の如何を，本システムは問わない．ただし，動画に対する再生速度の制御やそのデータ内容の改変は不可であるものとする．

2．2　同期の semantics

動画と音声との同期には，両メディアの表現特性の違いからいくつかのレベルが考えられるが，本稿では以下のようにその semanticsを定める．

“動画，音声のグローバルな時間軸上での

再生時の速度は録音時のそれと異なってもよく，さらにメディア間のタイミングのずれもある範囲内で許容する"

ここで，動画像を見ながら音声による解説等を付加し，その入力音声を符号化する時点を録音時，この録音によって得られた音声符号化データを復号化し，動画との同期再生を行う時点を再生時と表記した．

また，上述のずれの許容範囲は，動画と音声との時間的ずれに関して，人間が知覚の上で許容できる最小時間長程度とする．一般的な主観評価実験によれば，この時間長はおよそ 20ms ～ 50ms とされている[3]．

2．3　通常再生時の同期方式

筆者は各メディアに対して会話型操作が加えられていない通常再生時の同期方式に関して，その同期付けの方法も含めたいくつかの案を検討した．本稿ではこれらのうち，試作システムの実装を行ったものについて述べる．

（案1）　入力音声のゲインを監視することによって音声データを有音部分と無音部分とに判別し，有音部分＋無音部分の組をつくる．この組によって音声データを時間的に区切り，各区間の開始時点に同期ポイントを設定する．再生時の同期制御は，この各区間に相当する音声出力のタイミングを順次調整することによって行われる．これを図2に示す．

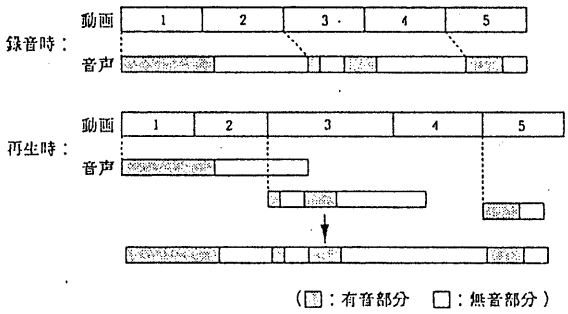

図2　通常再生時の同期方式（案1）

ここで，2．2節の semantics において考慮したメディア間の時間的ずれの許容範囲と，動画（フレームレート30）の1フレーム当たりの時間長とを比較すると，メディア間の時間的対応関係は動画のフレームナンバーのみに頼ることができる．従って，同期制御情報として録音時に取得すべき情報は，同期ポイント設定時に出力されている動画のフレームナンバーのみでよい．再生時にはこの情報に基づいて，各々のフレームナンバーの動画が出力された時点で，該当する音声データを復号化出力する．

また同期ポイント間の時間間隔について，あらかじめその最小時間長 Tmin と最大時間長 Tmax とを設定することによって，同期再生の保証，及び同期制御情報の効率的な生成という二点を同時に考慮した上での制御が可能となる．

この方式は動画に対しては一切の同期制御を加えずに，音声出力のタイミングのみを制御して動画出力に追従させるものである．この制御はアプリケーション層の単純な処理であるためにシステムへの負担が軽い上，動画データの符号化方式やその形態に対して汎用性を保つなどの特徴が挙げられる．

2．4　早送り再生時の同期方式

2．3節で述べた通常再生時の同期制御に関して，音声出力の無音部分の短縮だけでは対処できないような早送り操作が動画に加えられた場合にも同期が維持できるようこれを拡張する．

一般のマルチメディアシステムにおいて，ユーザが動画，音声を問わずなんらかのメディアに対して早送り操作を加えたい場合，以下のいずれかの希望によるものであろう．

（1）目的とするシーンを短時間で探し出し，現在の再生位置をそのシーンに移す

（2）概要をつかむために本来の時間的進行を省略し，本質的要素のみを抽出した情報を得る

この両者ともに，早送り再生の必要性はその操作の最中においてもメディアの内容を理解できるという点に依っている．これらを実現する同期方式各案を以下に示す．

（案1）　同期ポイントによって区切られた各

区間に対して情報としての優先度を予め与えておき，早送り再生時には優先度の低い区間の復号化出力をスキップすることによって，動画との同期を維持する．これを図3に示す．

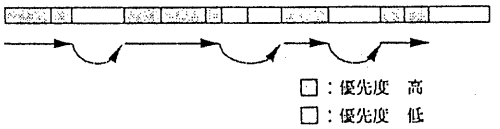

図3　早送り再生時の同期方式（案1）

（案2）　動画と音声の符号化データが多重化されていないことを利用し，通常再生時の音声データに加えて早送り再生時用のデータも別個，用意する．なおこれらの音声データは予め，動画との同期付けをしておく必要がある．これを図4に示す．

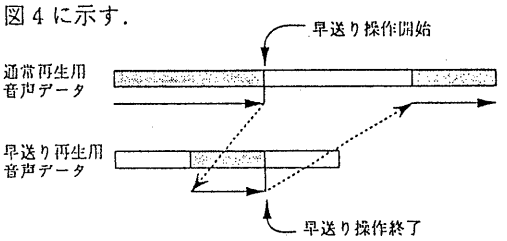

図4　早送り再生時の同期方式（案2）

本稿では（案1）における同期方式を用いて試作システムの実装を行った．

2．5　巻き戻し再生時の同期方式

早送り再生時と同様に，巻き戻し操作の目的を考慮した上で，巻き戻し再生時の同期方式の一案を以下に示す．

（案1'）　2．4節の（案1）と同様に，同期ポイントによって区切られた各区間に対して情報としての優先度を予め与えておき，巻き戻し再生時には優先度の低い区間の復号化出力をスキップすることによって動画との同期を維持する．これを図5に示す．

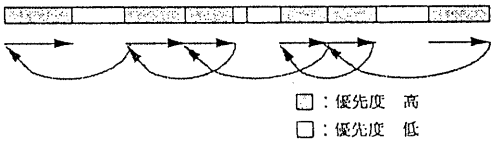

図5　巻き戻し再生時の同期方式（案1'）

この方式では各区間において，動画は逆時間で，音声は順時間でそれぞれ再生されることになる．

2．6　状態遷移時の処理について

通常再生，早送り再生，巻き戻し再生ともに，同期ポイントによって区切られたブロック単位で音声データを扱うため，各操作が開始される時の音声出力は必ず同期ポイントから行われる．多くの場合，ある状態から他の状態へと遷移する時点は同期ポイントではないので，このような時には次の同期ポイント設定時まで音声の出力が行われないことになる．

これを避けるために，状態が遷移する際にはこの時点で出力されている動画のフレームナンバーから音声データブロック内の該当する再生開始位置を推定する処理が必要となる．このアドホックな処理により，同期ポイント以外の位置からも音声出力の開始が可能となるため，状態遷移時の不具合が緩和される．一例として早送り再生から通常再生への移行時の処理を図6に示す．

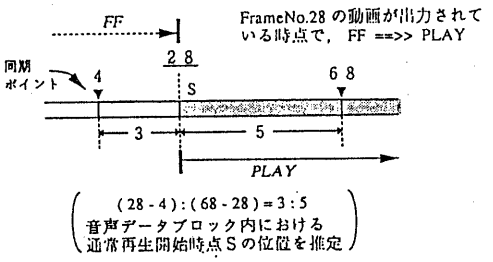

図6　状態遷移時の処理

3　試作システムの実装

3．1　実装範囲

2章の各同期方式案を，ワークステーション（Sun/SPARCStation10）上に実装した．動画データはファイルとして蓄積されており音声の録音を行うノードに既に存在している．また動画，音声各々のメディアは以下のものを使用した．

・動画 … FrameRate : 30 [frame/sec]

FrameSize : 180x120 [pixel]
ColorDepth : 8 [bit/pixel] (256color)
DataSize : 216000 [byte/frame] (無圧縮)

・音声 … Modulation : μ-law
(入力源は通常の音声信号のみ)

なおシステムの実装は，動画を見ながら音声の同期付けを行う録音部と，録音時に生成される同期制御情報を利用して動画と音声との同期再生を行う再生部が中心となる．実際のプログラミングはXViewを用いてOpenWindows上で行った．

3．2　録音部

録音部のシステム構成を図7に示す．

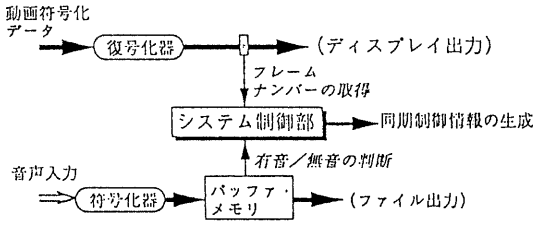

図7　録音部のシステム構成

録音時において，システム制御部は以下の各処理を順次行う．

（1）入力された音声の有音／無音の判断
（2）同期ポイントの設定
（3）同期制御情報の生成

本システムでは，汎用性を持たせるために動画再生に関する処理を別個のプロセスで実現しており，フレームナンバーの取得は共有メモリを介して行っている．また同期制御情報は同期ポイント毎に以下のformatに従って生成され，独立したファイルとして出力される．

[FrameNumber + DataSize + Priority]

この同期制御情報と音声符号化データとの対応関係を図8に示す．

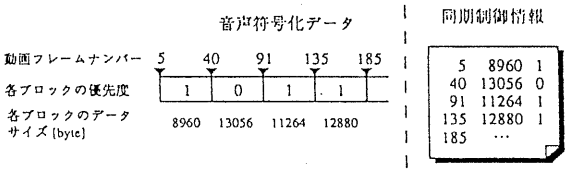

図8　同期制御情報と音声符号化データの対応関係

なお，各データブロックの優先度情報はデフォルトではすべて0(低)となっている．優先度を高くしたいブロックに対してはこの値を0(低)から1(高)に変更する．この作業は録音終了後にユーザが行う．

3．3　再生部

再生部のシステム構成を図9に示す．今回の実装で同期制御の対象としているのは通常再生，早送り（2倍速）再生，巻き戻し（2倍速）再生の各状態である．なお録音部と同様に，動画表示関連の処理は別個のプロセスで実現しており，プロセス間のデータ授受は共有メモリを利用している．

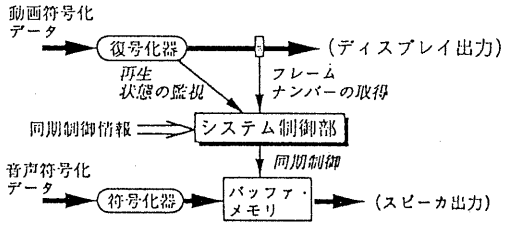

図9　再生部のシステム構成

システム制御部は以下の各処理を順次行う．

（1）同期制御情報の読み込み
（2）動画の再生状態の取得
（3）動画のフレームナンバーの検知と音声出力のタイミング調整

3．4　評価

実装した試作システムの各処理において得られた結果を以下に評価する．

（1）録音時における同期ポイントの設定

入力音声の有音／無音の判断の処理は良好に動作した．また本システムにおいては，同期ポイントの設定は入力音声のブレスの位置や文節毎にされることが望ましいが，無音区間であると判断する時間長を200～300ms程度に設定した場合，目標におおよそ近い結果を得ることができた．ただし，ゲインのみの判定では，適切

な位置に同期ポイントを設定することが難しいと思われる状況もいくつか考えられるので，こういった場合にはユーザがこれを指定できるようにすればより望ましいであろう．

（2）同期制御情報の効率的な生成

1秒毎に同期ポイントが設定されるものとして，1分間の同期録音を行った場合，生成される同期制御情報のサイズは，音声符号化データのそれに対して0.15%である．実際には同期ポイントがこれほど頻繁に設定されることはないので，この値はより小さなものになる．

（3）同期再生

trivialな例ではあるが，1から6までの数字が順次表示される動画を見ながら，これに合わせて「123…」と発音される音声を同期付け録音した場合をとりあげる．このとき，偶数の発音に対応する音声ブロックの優先度を“高”としてある．録音時における動画の出力を毎秒30フレームとした場合，再生時の各状態における音声出力は図10のようになった．

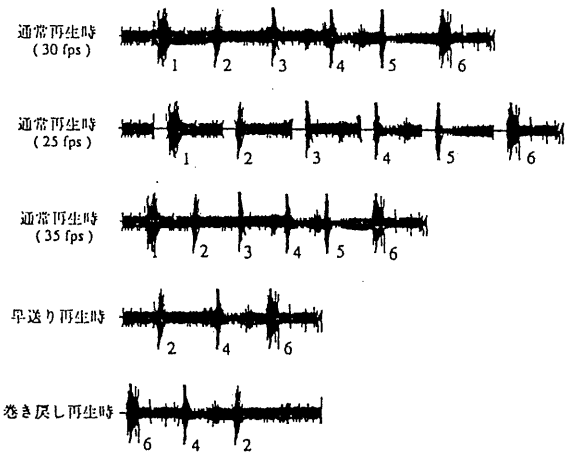

図10　同期再生時の音声出力

今回の実装では定性的な評価のみに終始したが，その同期方式の性質上，同期ポイント以外の再生位置における同期誤差が増大することは避けられない．しかし，動画の再生速度が極端に変化する場合を除けば，適切な同期ポイントの設定によって同期再生時の視聴覚上の違和感は軽減される．簡易な制御であるという点を含めて実用的な同期方式の一つであると思われる．

4　おわりに

本稿では，会話型操作を伴うマルチメディアシステムにおいて，特に動画に音声を同期付けする場合の同期方式案をいくつか提示し，その一部を実装，評価した．この同期方式の特徴は，多重化されていないメディア間の同期を保証するという点にあり，これによって各メディアを個別に処理できるなどの柔軟なシステム構成が可能となる．また，もうひとつの特徴はメディア情報の内容に言及した同期制御を行うという点であり，これは同期再生時の視聴覚上の違和感の軽減と効率的な同期制御とを実現する．本稿で述べた方式は，音声に関して，通常再生時には無音部分を，早送り（巻き戻し）再生時には優先度の低いデータブロックを犠牲にすることによって同期を維持するものであったが，データ内容を時間的に区切ることのできるメディアであれば同様の同期制御が可能であろう．

今後はより高精度な同期の実現や，実装範囲のネットワーク上への拡張を，また同期ポイントの設定法に関しても，メディア間の対応関係をより適切に表現できるよう改良を加えていく予定である．

参考文献

[1] 大野隆一，相田仁，齊藤忠夫：“マルチメディアの会話型操作についての一検討”，電子情報通信学会春季大会93'D-217

[2] 安田浩著：“マルチメディア符号化の国際標準”，丸善，1992-6.

[3] 市野良典，二階堂誠也共著：“オーディオ機器”，コロナ社，1991-8

[4] 藤川和利，下条真司，松浦敏雄，西尾章治郎，宮原秀夫：“分散型ハイパーメディアシステムHramonyにおける情報間同期機構の実現”，電子情報通信学会論文誌D-I，J76-D-I，9，1993-9

[5] Klara and Joathan：“An Integrated Multimedia Architecture for High-Speed Networks”，Mulitimedia'92