MPEG-4 実時間コンテンツ編集システムの同期処理機構

三浦康之　　勝本道哲
独立行政法人　情報通信研究機構
{miu,katumoto}@nict.go.jp

Synchronization Model of Real-time Contents Editing System for MPEG-4

Yasuyuki Miura,　Michiaki Katsumoto
National Institute of Information and Communications Technology
{miu,katumoto}@nict.go.jp

概要

我々は，複数の画像・音声を入力とし，リアルタイムで MPEG-4 ビデオオブジェクトへの符号化・編集・配信を一体的に行う実時間コンテンツ編集システムを提案している．本システムは，編集・配信・受信の 3 種類のモジュールにより構成され，離れた複数地点からのデジタルビデオによるライブ映像を用いたコンテンツを提供するこが可能である．相互に離れた複数地点からの動画・音声素材を用いて最小限度の遅延によるリアルタイム配信を行うためには，フレームの表示間隔を一定に保つなどの適切な同期処理が不可欠となる．本稿では，実時間コンテンツ編集システムにおける適切な同期アルゴリズムを提案する．実証実験を行った結果，提案するアルゴリズムは遅延を少なく抑えフレームロスの少ない動画像の提供が可能であることが明らかになったので報告する．

1.　は じ め に

近年の広帯域なネットワークの普及により，大容量の動画像コンテンツの配信が可能となっている．今後さらなる高速化が明らかであり，ユーザ同士で大容量の動画像を送受信することが可能になると予想される．それにともない，広帯域ネットワークを通じて画像・音声を配信するサービスやツールが数多く実現している [1) 2) 3) 4)]．これらはいずれも単一の動画を対象としたアプリケーションであり，多数の動画を扱うには不向きである．

我々が提案するリアルタイムコンテンツ編集システム [5) 6) 7)]は，実時間分散環境に散在する複数の画像や音声を入力とし，MPEG-4[8)]のビデオオブジェクトおよび編集用言語の符号化および復号をリアルタイムで行うものである．本システムにおいて，ライブ映像・音声の送信に際して適切な方法によるオブジェクト間同期を行うことは，コンテンツの品質を決める上で重要な要素である．映像は各フレームの表示間隔が一定でなければ不自然な感じを与える．また，音声情報は再生する際に空隙や途切れがあると極端に聞き取りづらくなるため，適切な同期処理が必要になる [9)]．本稿では，MPEG-4 オブジェクトの配信に際して送受信それぞれの同期に伴う時間コストを見積もり，最適なタイミング設定を行う方法を提案する．

2.　実時間コンテンツ編集システム

2.1 MPEG-4 規格

MPEG-4 ビデオ符号化においては，個々の動画を VO（Video Object）と呼んでいる．さらに VO を構成する各フレームを VOP（Video Object Plane）と呼ぶ．これがビデオ符号化の基本となる．VOP には予測符号化の違いにより，I-VOP (Intra VOP) ，P-VOP (Predicted VOP) および B-VOP (Bidirectional Interpolated VOP)が存在する． I-VOP はフレーム内のみで符号化を行う VOP，P-VOP は一つ前の I-VOP または P-VOP を利用してフレーム間予測を行う VOP， B-VOP は前後の I-VOP や P-VOP を利用してフレーム双方向予測を行う VOP である．

2.2 システム条件

インタラクティブコミュニケーションでは，会話のテンポを損なわないように配信に伴う遅延が少ないことが特に求められる．遠隔会議に関する過去の実験において[10) 11) 12)]，約 1 秒前後の遅延を伴う MPEG-2 符号化装置を使用した遠隔会議システムにおいては遅延時間が参加者同士のスムーズな会話の障害になることが報告されている. また, 通常の会話においてはほぼ400ms程度が往復遅延の検地限であることが知られている[12)]. 従って，同期処理に当たっては低遅延なアルゴリズムの構築が必要とされる．また，音声と映像の遅延時間には同期効果があり，双方の遅延が同程度であれば遅延が認識されにくいことが知られている[12)]．そのためしばしば画像・音声情報相互において同期を行う必要が生じる.

2.3 システム構成

図 1 に, 構築するシステムの概念図を示す. 本システムは, 入力装置として複数のビデオカメラと, それらによる入力画像を符号化して配信する複数の配信モジュール, 画像を受信して表示する受信モジュール, および一つの編集モジュールから構成される.

編集モジュールは配信モジュールに対する画像・音声のマルチキャスト要求を実際に行い, シーン記述言語の生成およびマルチキャストを担当する. 配信モジュールは, 入力装置から DV 形式で取り込まれた画像・音声を, DV-MPEG 変換器を用いて MPEG-4 への符号化を行い, 複数の受信モジュールに対しインターネットを介してマルチキャストを行う. 受信モジュールでは, 複数の MPEG-4 画像音声データおよびシーン記述に基づいて動画像を表示する.

配信モジュール, 受信モジュール, 編集モジュールとしては任意の形態が考えられる. 例えば, 汎用のサーバマシン上のプログラムとして配置することもできるし, 家庭用 PC 上で動作させても構わない.

2.4 VOP 選択アルゴリズム

前節のような用途を補助するため, 本システムの配信モジュールにおける DV-MPEG 変換器において, 使用するハードウェアの性能に対応した柔軟な符号化を行うための VOP 選択アルゴリズム[7)]を採用している. VOP 選択アルゴリズムとは, 各 VOP の符号化時間計測の結果を用いて後続の VOP の符号化時間を予測して, 複数個の VOP の集合である VOP ブロックを選択するというものである. アルゴリズム中で, I-VOP と P-VOP それぞれにつき, 後続の VOP によって参照されないため符号化処理の一部を簡略化できる簡略化 VOP と簡略化の不可能な非簡略化 VOP に分け, ①一個の非簡略化 I-VOP の後に複数の非簡略化 P-VOP が続き, 簡略化 P-VOP を配置する IP ブロック ②一個の非簡略化 I-VOP の後に一個の簡略化 P-VOP を配置し, 続いて複数の簡略化 I-VOP が続く PI ブロック ③一個の簡略化 I-VOP のみにより構成される I ブロック の中から適切な VOP ブロックの種類と長さを選択している. これらの VOP 長はそれぞれ,

① IP ブロックならば VOP ブロックの合計符号化時間が目標符号化時間以下になる最長の長さ
② PI ブロックならば VOP ブロックの合計符号化時間が目標符号化時間以下になる最短の長さ
③ I ブロックなら 1

となる.

VOP 選択アルゴリズムの使用により, 本システムはハードウェアの性能にしたがって適切な量の符号量を削減することが可能となる. さらに, MPEG-4 のレートコントロール機能を組み合わせることにより, 一定の符号量の中でハードウェアの性能にしたがって可能な限り高品質な動画像を提供することができる.

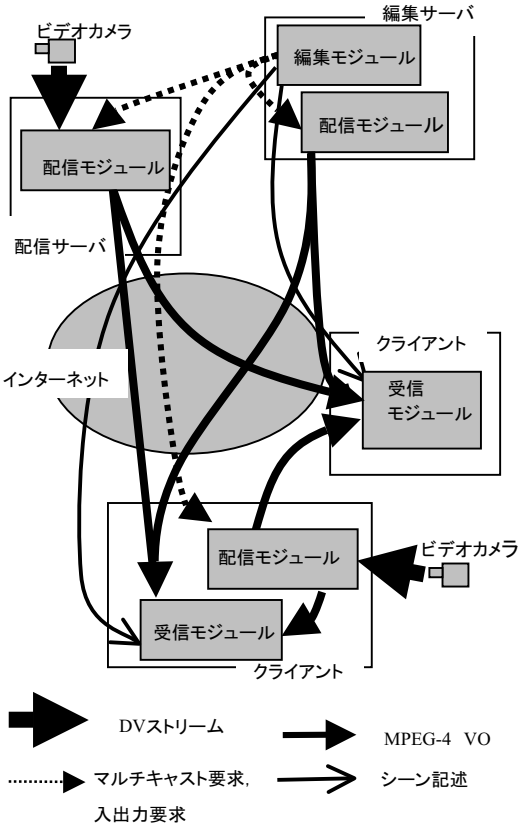


図 1 リアルタイムコンテンツ編集システムの構成例

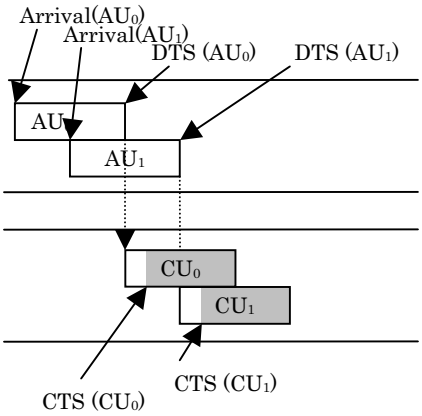


図 2 MPEG-4 システムのタイミングモデル

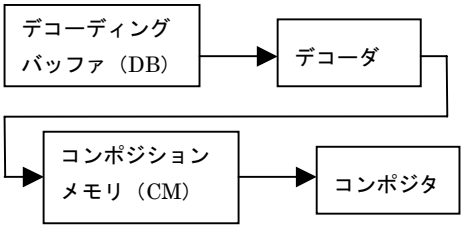


図 3 デコーダの流れ図

3. タイミングモデル

3.1 MPEG-4 システムによるモデル

図2に，MPEG-4 規格のシステムパート [13]において定義されているタイミングモデルを，図3に，タイミングモデルにおけるデコーダのデータフロー図をそれぞれ示す．図2，図3中のおのおののユニットおよびタイムスタンプの詳細を以下に示す．

AU(Access Unit)：復号・合成のための時間管理や同期のための処理単位となる．画像情報については VOP の符号化データ，音声情報については画像 1VOP に相当する符号化データにあたる．

CU(Composition Unit)：1AU の符号化データに相当し，同期のための処理単位となる．

DTS(Decoding Time Stamp)：複合化バッファから AU を取り出し復号を開始する時刻．AU はこの時までにデコーディングバッファに到着し，デコードが開始されなければならない．

CTS(Composition Time Stamp)：CU がコンポジションメモリ内で有効になる時間．同時に，一つ前の CU が破棄される時間でもある．

実際には，VOP の順序関係が入れ替わらない場合 DTS=CTS としても良いことになっている．我々のシステムでは定義されたタイミングモデルに基づき，配信モジュールで各オブジェクトストリームをパケットにして受信モジュールに送り込む．各パケットには，符号化データとともに CTS が含まれており，各パケットの次のパケットに含まれる CTS がそのパケットの寿命となる．

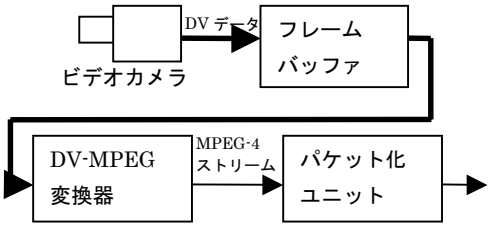


図 4 配信モジュールにおける動画像情報の流れ

3.2 実時間コンテンツ編集システムにおける問題点

前節に示された MPEG-4 の一般的なタイミングモデルに基づき，本システムにおいて VOP の同期を行うためには，下記のような問題点を考慮する必要がある．

① VOP の種類により符号化時間が異なるため，符号化終了のタイミングがまちまちになる．これは，VOP 送信の間隔が一定にならない上にタイミングの予測を困難にする．

② 受信モジュールに符号化した VOP が到着する時刻と CTS=DTS が接近している場合，到着時刻が CTS=DTS の後になる危険があり，フレームスキップや同期の乱れの原因となる．逆に CTS=DTS に余裕を取って VOP が到着する時刻からの間を長めに取ると，遅延の原因になる．そのため，適切な CTS=DTS を決める必要がある．

次章において，上記の二つの問題点について時間モデルを作成して，作成したモデルをもとに適切な同期機構の提案を行う．

4. 同期アルゴリズム

4.1 パラメータの定義

図 4 に，本システムにおける配信モジュールにおける動画像情報の流れ図を示す．配信モジュールでは，ビデオカメラから取得した DV ストリームをいったんフレームバッファに格納する．格納されたデータは順番に DV-MPEG 変換器に送られ符号化される．符号化が完了したビット列はパケット化ユニットに送られ，パケット化した後直ちに受信モジュールに配信される．

本章で述べるモデルにおける各時刻を示すパラメータを以下のように定義する．なお，VOP の通し番号として i を使用する．

$CUP(i)$：VO における i 番目の VOP である VOP_i をビデオカメラから取得した時刻

$E_{start}(i)$: VOP_iの符号化を開始する時刻
$FIN(i)$: VOP_iの符号化を終える時刻
$Depart\_V(i)$: VOにおけるi番目のAUが配信モジュールを出発する時刻
$Arrival\_V(i)$: VOにおけるi番目のAUが受信モジュールに到着する時刻
$CTS\_V(i)$: VOにおけるi番目のCUに設定されたCTS
T_{rIVOP}, T_{IVOP} , T_{rPVOP}, T_{PVOP}: 各VOPにおける平均符号化時間
F: フレームレート

4.2 配信モジュールにおける基本モデル

VO の表示間隔が一定でなければ，見る側に不自然な動きを感じさせる動画になる．また，同時に送られる音声情報と映像情報，あるいは音声情報同士で厳密に同期を取る必要がある場合などの利便性を考慮して，本システムのオブジェクト間の同期処理には，各 VOP の CTS の同期インターバルを一定間隔とするという以下の条件が与えられる．

$$CUP(i) = CUP(0) + i \cdot 1/F \quad (1)$$

また VOP_iの符号化に要する時間を$Encode(i)$とおくと

$$FIN(i) = E_{start}(i) + Encode(i) \quad (2)$$

なお，$Encode(i)$は VOP のタイプによって異なり，通常 P-VOP は I-VOP に比べて長くなる．

直前の VOP の符号化を終えてから次の VOP の符号化を開始することになるので

$$E_{start}(i) \geqq FIN(i\text{-}1) \quad (3)$$

これらの式から

$$\forall j \; E_{start}(i) \geqq CUP(i) \quad (4)$$

となるような$E_{start}(i)$が分かれば，符号化開始時刻をおおよそ見積もることができる．

VOP 選択アルゴリズムを使用した場合，配信モジュールにおいて発生する最悪のケースにおける符号化遅延を以下のように見積もる．

VOP_iの，配信モジュールにおける遅延$Ls(i)$は

$$Ls(i) = FIN(i) - CUP(i) \quad (5)$$

と定義される．

(4)式の条件を満たしつつ効率よく符号化を進めるためには，直前のVOPの符号化終了時点で次のフレームが取得されていなければならないので

$$\forall i \; FIN(i\text{-}1) \geqq CUP(i) \quad (6)$$

を満たす必要がある．上式を満たさないVOP_iがあると，VOP_{j-1}の符号化終了とVOP_iの符号化開始の間にフレーム取得待ちの空き時間が発生する．そのため，画像取得と符号化開始の間には適切なタイムラグを設定する必要がある．このタイムラグT_Aを時間補正値と呼び，

$$T_A = E_{start}(0) - CUP(0) \quad (7)$$

と定義する．適切な時間補正値を設定することにより，フレーム取得待ちが発生しなくなり，(3)式は

$$E_{start}(i) = FIN(i\text{-}1) \quad (8)$$

となる．

最大の遅延が発生するケースは，以下の時間の合計で示される．

① 時間補正値による遅延 L_A

時間補正値 T_A がそのまま「時間補正値による遅延」となる．

② ブロック内遅延 L_i

各 VOP ブロックにおいて各 VOP に発生する遅延の最大値が「ブロック内遅延」となる．すなわち，$Ls(i) - T_A$ がブロック内遅延となる．

以降，各項目の値を算出する．

4.2.1 時間補正値による遅延

時間補正値がそのまま遅延となるため，ここでは必要な時間補正値を見積もる．

VOP 選択アルゴリズムを使用した場合，目標符号化終了時刻よりも若干早くVOPブロック全体の符号化が完了する可能性がある．その際の，目標符号化終了時刻と実際の符号化終了時刻の差の最大値をT_{ad}とすると，目標符号化終了時刻ちょうど符号化　を終了した場合に比べて直後のVOPブロックにおける各VOPの符号化終了時刻が最大でT_{ad}だけ早まる可能性がある．

IPブロックの場合，VOPブロック長Nは目標符号化時間以下になる最長の長さに設定されるため，最もT_{ad}が大きくなるケースは，VOPブロック長$N+1$における符号化終了時刻が，目標符号化時刻よりも若干長くなる場合に起こりうる．この場合における符号化終了時刻は，VOPブロック長$N+1$の場合に比べてT_{PVOP}短くなり，目標符号化終了時刻は$1/F$短くなるため

$$T_{ad} = T_{PVOP} - 1/F \quad (9)$$

となる．

PIブロックでも同様に，VOPブロック長Nは目標符号化時間以下になる最長の長さに設定される．最もT_{ad}が大きくなるケースは，VOPブロック長$N-1$における符号化終了時刻が目標符号化時刻よりも若干長くなり，かつVOPブロック長Nにおける符号化終了時刻が目標符号化時刻以下になる場合に起こりうる．この場合における符号化終了時刻は，VOPブロック長$N-1$の場合に比べてT_{rIVOP}長くなり，目標符号化終了時刻は$1/F$長くなるため

$$T_{ad} = 1/F - T_{rIVOP} \quad (10)$$

となる．

VOPブロックの目標符号化終了時刻と次のVOPブロックにおける最初の VOP のフレーム取得時刻との間に

は T_{ad} の時間差がある. したがって, 実際の符号化終了時刻と次の VOP ブロックにおける最初の VOP のフレーム取得時刻との間には $T_A - T_{ad}$ の時間差がある.

VOP ブロックにおいて符号化待ち時間 $E_{start}(i) - CUP(i)$ の値が最小となるのは, 最初の I-VOP の直後に位置する P-VOP であることから, 最初の VOP の直後に着目する. 次の VOP ブロックにおける最初の VOP の符号化終了時刻までに, 二番目の VOP のフレームを取得しなければならないので

$$T_A - T_{ad} + T_{IVOP} \geqq 1/F \quad (11)$$

である必要がある. したがって, T_A の条件は

$$T_A \geqq 1/F + T_{ad} - T_{IVOP} \quad (12)$$

であるので, 直前のブロックが PI ブロックの場合,

$$T_{A\_pi} \geqq 2/F - T_{rIVOP} - T_{IVOP} \quad (13)$$

IP ブロックの場合,

$$T_{A\_ip} \geqq T_{rPVOP} - T_{IVOP} \quad (14)$$

となる.

IP ブロックが主に使用される環境では $2/F - T_{rIVOP} - T_{IVOP} > T_{rPVOP} - T_{IVOP}$ となり, PI ブロックが主に使用される環境では $T_{rPVOP} - T_{IVOP} > 2/F - T_{rIVOP} - T_{IVOP}$ となることから, 実装上は両者のうち値の低い方を使用し

$$L_A = \min\{2/F - T_{rIVOP} - T_{IVOP},\ T_{rPVOP} - T_{IVOP}\} \quad (15)$$

とする.

4.2.2 ブロック内遅延

(5)式, およびブロック内遅延の定義により, ブロック内遅延は

$$L_i = Ls(j) - T_A = FIN(j) - CUP(j) - T_A \quad (16)$$

となる. ()式により, ブロック内遅延が最大になるケースでは, 直前の VOP ブロックにおける最後の VOP である VOP $_{b0}$ において $FIN(b0)$ が最大になる場合, すなわち VOP ブロックの符号化終了時刻が目標符号化終了時刻と一致する場合であることから,

$$FIN(b0) = CUP(b0+1) + T_A \quad (17)$$

となる. したがって, ブロック内の n 番目の VOP である VOP $_{b0+n}$ におけるブロック内遅延は, (16)(17)式より

$$L_i = FIN(b0+n) - CUP(b0+n) - T_A$$
$$= CUP(b0+1) + \sum_{i=1}^{N} Encode(b0+i) - CUP(b0+n) \quad (18)$$

となる.

PI ブロック内では, 符号化時間の大きい簡略化 P-VOP の直後が最も符号化遅延が大きくなる. 簡略化 P-VOP は PI ブロックの 2 番目の VOP であることから, VOP ブロック C が PI ブロックである場合, ブロック内遅延 L_{i_PI} は

$$L_{i\_PI} = CUP(b0+1) + \sum_{i=1}^{2} Encode(b0+i) - CUP(b0+2)$$
$$= T_{IVOP} + T_{rPVOP} - 1/F \quad (19)$$

となる.

IP ブロックでは, 最初の VOP 以外はすべて P-VOP となる. したがって, 最後の VOP ブロックが最も符号化遅延が大きくなる. VOP ブロックの符号化時間が最も大きな値になるのは, 目標符号化時間通りに符号化が完了した場合であるので, ブロック長 N の VOP ブロックについて,

$\Sigma_{i=1}^{N} Encode(b0+i) = N/F$ となる. したがって, ブロック内遅延 L_{i_IP} は

$$L_{i\_IP} = CUP(b0+1) + \sum_{i=1}^{N} Encode(b0+i) - CUP(b0+N)$$
$$= N/F - (N-1)/F = 1/F \quad (20)$$

となる.

実際は, これらの値のうち大きい方を使用するため,

ブロック内遅延 L_i は

$$L_i = \max\{T_{IVOP} + T_{rPVOP} - 1/F,\ 1/F\} \quad (21)$$

とする.

4.3 受信モジュールにおける基本モデル

受信モジュールについて, 以下のモデルを構築できる. なお, 以降は CTS＝DTS と仮定し, 双方の表現を CTS で統一する.

V における i 番目の AU の伝送レイテンシおよび復号時間の合計を $Trans\_V(i)$ とおくと, AU の出発時刻と到着時刻の関係は

$$Arrival\_V(i) = Depart\_V(i) + Trans\_V(i) \quad (22)$$

配信モジュールにおける VOP の符号化完了時刻がパケットの出発時刻であると見なすと, $Depart\_V(i) = FIN(i)$ なので

$$Arrival\_V(i) = FIN(i) + Trans\_V(i) \quad (23)$$

となる. したがって, システムにおける全体の遅延 L は

$$L = Ls(i) + Trans\_V(i) \quad (24)$$

全体の遅延の最大値 L_{max} は

$$L_{max} = \max_j\{Ls(i) + Trans\_V(i)\} \quad (25)$$

となる. 前節より, VOP 選択アルゴリズムを使用した場合の最大の遅延は(15)式および(21)式の合計で示されることから, $Ls(i)$ と $Trans\_V(i)$ が互いに独立ならば(24)式は

$$L_{max} = \min\{2/F - T_{rIVOP} - T_{IVOP},\ T_{rPVOP} - T_{IVOP}\}$$
$$+ \max\{T_{IVOP} + T_{rPVOP} - 1/F,\ 1/F\}$$
$$+ \max_i\{Trans\_V(i)\} \quad (26)$$

と置き換えることができる. ここで, $\max_i\{Trans\_V\}$ は, 全 VOP における $Trans\_V$ のうちの最大の値に当たる.

4.4 タイムスタンプの決定

実際には符号化時間や伝送レイテンシにばらつきがあることから, 前節により求められた L_{max} に, 遅延の分散を加えた値をタイムラグとして設定して CTS を決める. 分散は L_{smal} や $Trans\_V$ に対して発生する. CTS の制約条件により

$$CTS\_V(i) = CTS\_V(0) + i \cdot 1/F \quad (27)$$

(28)式および(1)式により

$CTS\_V(i) - CUP(i)$

$$= CTS\_V(0) - CUP(0) \quad (28)$$

すべての VOP について $CTS_V(i) - CUP(i)$が遅延時間以上となれば良いことから, CTS を

$$CTS\_V(0) = L_{max} + CUP(0) \quad (29)$$

と設定する.

4.5 タイムスタンプの設定および変更

初期の CTS を下記の手順で決定する.

- 最初に L_{max} ＝3／F と仮定しておき, 非簡略化 I-VOP および簡略化 P-VOP について, n フレームずつ符号化を行い, 平均値および標準偏差を算出.
- 必要ならば伝送による遅延の平均と標準偏差も出しておく.
- 算出されたパラメータおよび(29)式をもとに初期の CTS（$CTS_V(0)$と定義されている値）を決定する.

ただし, この方法でいったん CTS が決定されてしまうと, (27)式の制約により, 以後の VOP の CTS を変更できなくなる. しかし実際には, 配信の途中で画質が変わる, 他のプロセスが動く, ネットワークが混雑するなどといった要因により符号化時間や伝送時間が変化することが考えられる. そこで, ある一定の条件下で(27)式の制約を破り, CTS を設定し直す必要がある. ただし, CTS を再設定した時動画像の動きが一瞬ぎくしゃくしたものになるため, 再設定は最小限に抑えることが望ましい. そこで, すべての VOP ブロックに均一に登場する非簡略化 I-VOP および簡略化 P-VOP に着目し

$$| L_{prev} - L | \geqq R(\sigma_{I\text{-}VOP+rP\text{-}VOP}) \quad (30)$$

または

$$| L_{trans\text{-}prev} - L_{trans} | \geqq R \cdot \sigma_{trans} \quad (31)$$

のいずれかまたは双方の条件を満たしている時に, 符号化時間または伝送・復号時間が変化したと見なして, (30)式の条件を満たす場合には新たに計測された T_{IVOP} と T_{rPVOP}, (31)式の条件を満たす場合には新たな $Trans\_V$, および(26)式に基づいて L_{max} を設定し直し, 新たに設定された値に基づいて(29)式を用いて CTS 変更の処理を行う. ここで, (30)式(31)式の各数値は次の以下の通りである.

L_{prev}: 前回に CTS を再設定（もしくは最初に CTS を決定）した際の $T_{IVOP} + T_{rPVOP}$ の値

L: 現在の $T_{IVOP} + T_{rPVOP}$ の値

$L_{trans\text{-}prev}$: 前回に CTS を再設定（もしくは最初に CTS を決定）した際の $Trans\_V$ の値

L_{trans}: 現在の $Trans\_V$ の値

R: 更新頻度

$\sigma_{I\text{-}VOP+rP\text{-}VOP}$: $T_{IVOP} + T_{rPVOP}$ の標本標準偏差

σ_{trans}: $T_{IVOP} + T_{rPVOP}$ の標本標準偏差

上記のうち, パラメータ R はシステムに許容されるタイムスタンプ更新の頻度により決めるパラメータで, この値が大きいほどタイムスタンプ更新の回数が少なくなるため不規則なフレーム更新や音とびなどの品質劣化が起こりにくくなる一方で, VOP の平均符号化時間などのシステムの状態にタイムスタンプの設定が適合しなくなるおそれがある.

4.6 VOP 選択アルゴリズムを使用しない場合

VOP 選択アルゴリズムを使用しない場合においても, B-VOP の使用によりフレームの順序が入れ替わるため遅延は深刻なものになる. したがって, B-VOP の使用は PC 上での実時間配信には適さない. B-VOPを使用しない場合, I-VOP と P-VOP のみが使われるため VOP の並びは, 一定の長さ N の IP ブロックが並ぶことと等価であると考えることができる. このような場合, 全 VOP の平均符号化時間 T_{ave} と $1/F$ の関係により遅延の大きさは異なるものになる. 以下, それぞれの場合について述べる.

4.6.1 T_{ave}＝$1/F$の場合

VOP ブロックの目標符号化時間と実際の符号化時間が理論的に一致するため, (9)式に代わって T_{ad}＝0 とする. (12)式より, 時間補正値による遅延は

$$L_A = 1/F - T_{IVOP} \quad (32)$$

VOP の並びが, 目標符号化時間通りに符号化を終了する IP ブロックと等価であることから, ブロック内遅延は(20)式に従い

$$L_i = 1/F \quad (33)$$

したがって,

$$L_{max} = 2/F - T_{IVOP} + \max_i\{ Trans\_V(i)\} \quad (34)$$

となる.

4.6.2 T_{ave}＜$1/F$かつ $1/F$＜T_{PVOP}の場合

VOP ブロックの平均符号化時間が $1/F$ よりも若干短くなるため, VOP ブロックの符号化直後に, 符号化を行わない空白の時間が発生する. VOP 長 N の VOP ブロックについて

$$T_{ad} = N/F - T_{block} \quad (35)$$

I-VOP の直後の P-VOP について(4)式を満たす必要があるため, 時間補正値による遅延は

$$L_A = 1/F - T_{IVOP} \quad (36)$$

VOP ブロックの合計符号化時間は目標よりも T_{ad} だけ短いことから, ブロック内遅延も

$$L_i = 1/F - T_{ad} \quad (37)$$

したがって,

$$\begin{aligned} L_{max} &= T_{block} - T_{IVOP} - (N-2)/F + \max_i\{ Trans\_V(i)\} \\ &= (N-1)T_{PVOP} - (N-2)/F \\ &\quad + \max_i\{ Trans\_V(i)\} \end{aligned} \quad (38)$$

となる.

4.6.3 $1/F$＞T_{PVOP}の場合

符号化時間に余裕があるため, フレーム取得直後に符号化が行われ, 符号化終了後, エンコーダは次のフレームの取得を待つことになる. したがって, 時間補正の必要が

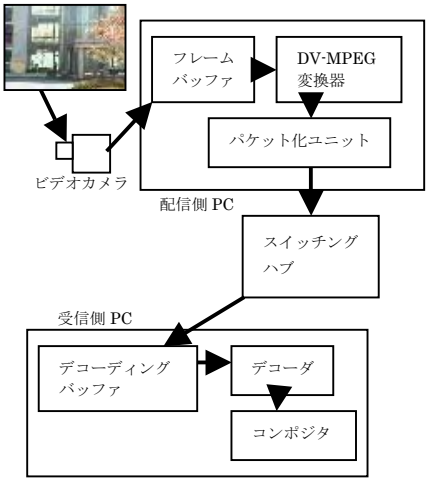
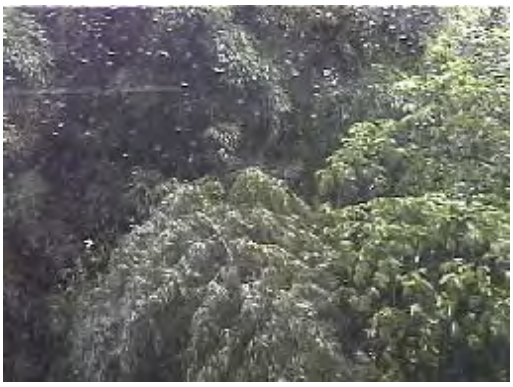


図 5 実験用システムの構成

(a) movie1



(b) movie2

図 6　実験映像

なく，ブロック内遅延も T_{PVOP} と等価になる．したがって

$$L_{max} = T_{PVOP} + \max_i\{Trans\_V(i)\} \quad (39)$$

となる．

4.6.4 $T_{ave} > 1/F$ の場合

この場合フレームロスが発生する可能性がある．

単位時間あたりの符号化フレーム数が N/T_{block}，単位時間あたりの取得フレーム数が F であることから，符号化が実行されるフレームの比率は $N/(F \cdot T_{block})$ となる．したがって，$1-N/(F \cdot T_{block})$ の割合でフレームロスが発生する．

ビデオカメラから取得されたフレームは，符号化開始までフレームバッファに格納され，フレームバッファはいっぱいになると破棄されるため，遅延時間はフレームバッファに格納できるフレームの数に比例して非常に大きなものになるため，このような条件下での実時間環境における動画配信は適切ではない．

5.　動作実験

5.1 実験環境

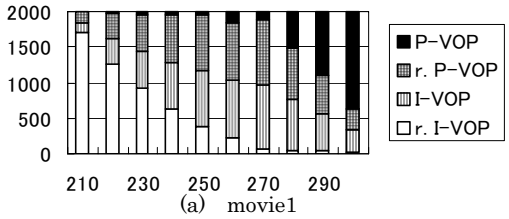
前章に示したアルゴリズムを検証するため，MPEG-4 参照ソフトウェア [14]を用いて 4 章のアルゴリズムを実装した簡単な配信・受信システムを構築し，ライブ映像の符号化・伝送・表示を行った．

図 5 に実験用システムの構成図を示す．システムは，配信モジュールと受信モジュールにより構成されるもので，フレームレート R の条件下において①DV データの取り込み，②符号化，③送受信，④復号，⑤同期，の一連の動作を実行している．配信側のPCは 2.0GHz の PowerPCG5 を搭載したデュアルプロセッサタイプの PC である．受信側のPCは 3.0GHz の PentiumⅣプロセッサを搭載したもので，動画像の復号や表示を行うために十分なスペックを有している．

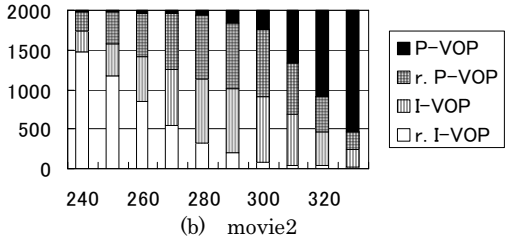
サンプル画像を図 6 に示す．図 6(a) は建物の外側を写した映像で，ほとんど画面に変化は現れない．図 6(b) は樹木が激しく風に揺れている映像で，画面の変化が大きく，符号化時間の分散が図 6(a) より大きい．特に指定がなければ，安全度 K を 0 とし，更新頻度 R を 1 としている．それぞれについて 2000 フレームの符号化を行い，フレームロスの数と更新頻度，および平均遅延を計測している．その際，VOP ブロックの最大長を 10 とし，各 VOP の符号化時間の平均値や分散を求めるサンプル数を，最近符号化が行われた 16 フレームとしている．

5.2 実験結果および考察

図 7 に，$R = 1$ におけるフレーム間隔 $1/F =$ を 230msから 10ms おきに設定した場合の VOP 数を示す．図 7 (a) は movie1 の，図 7 (b)は movie2 の結果である．図 7 のように，$1/F$ が大きくなるにしたがって I-VOP の数が減少して P-VOP が増加している．また，movie2 の符号化時間 movie1 に比べてやや長めのため，若干 P-VOP の数が少ない．Movie1 では，$1/F$ =210-240ms では I-VOP が主に使用され，$1/F$ =280-200ms では P-VOP が主に使用されている．Movie2 では，$1/F$ =240-270ms では I-VOP が主に使用



(a) movie1



(b) movie2

図 7 VOP 数の比較

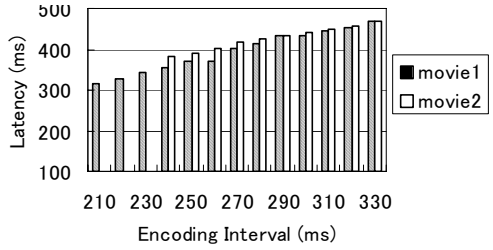


図 8 平均遅延

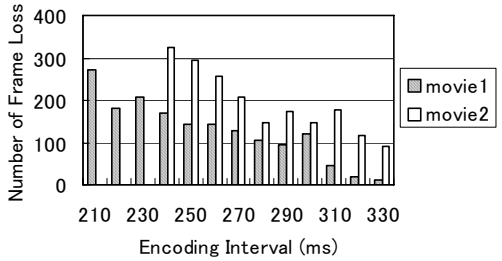


図 9 フレームロス数

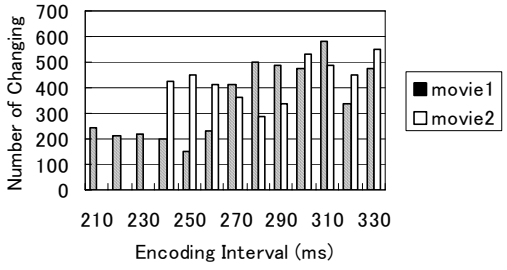


図 10 更新回数

され, 1/*F* =310-330ms では P-VOP が主に使用されている.

双方の画像に対する平均遅延, フレームロスの数, およびと更新頻度を図 8,9,10 に示す. 図 8 に示すように, 1/*F* が大きくなるに従って平均遅延は大きくなっている. これは, ほぼ 4.3 節に示した値通りとなっている.

図 9 において, フレームロス数は 100 から 200 であることが分かる. これは, ほぼ 10～20 フレームに一度の頻度である. これはほぼ VOP ブロック数個に一回の割合である. これらのフレームロスが発生する理由は, 現実の符号化時間がアルゴリズム中で使用される平均時間よりも大きくなる場合があるためであると考えられる. I-VOP の多い領域におけるレームロス数が若干多いのは, 符号化時間計測用に, PI ブロックの多い状態では一定の割合で IP ブロックを挿入しているためである. IP ブロックは符号化時間が長めになるため, 大きな遅延が発生し, フレームロスの原因となる. また, movie1 に比べてmovie2 のフレームロス数が多いことが分かる. movie2 では画像の変化が激しいため, 送信側・受信側双方で符号化や画面表示に要する時間が大きく変わるためである. 図 10 の movie1 において P-VOP が頻発する領域で更新頻度が高くなるのは, 受信側の画面表示が原因であることが分かっている.

図 11 および図 12 に, movie2 における更新頻度 *R* の違いによる更新回数, およびフレームロス数の違いを示す. 図 11 に示すように, *R* を高く設定するに従って更新回数が減少している. また, 図 12 に示すように, *R* を高く設定した場合, 若干ながらフレームロスの数が増加する. これは, 更新回数の減少に伴って movie2 における符号化時間の変化に対応できなくなるためである.

図 13 に, VOP 選択アルゴリズムを使用しない場合のフレームロス数を示す. 実験では, ブロック長 3 の IP ブロックを使用した符号化を行っている. 1 フレームあたりの平均符号化時間は, movie1 が約 280ms, movie2 が約 307ms である. 図 13 に示すように, 符号化インターバルが平均符号化時間を下回るケースではすべてのフレームを処理することができないため遅延が大きくなり, 結果としてほとんどのフレームがロスしてしまう. movie1 と movie2 では平均符号化時間が異なるため, このような状態になる条件が, 異なってくる. 図 13 では, movie1 は 280ms 以下, movie2 では 300ms 以下でフレームロスが多発する. そのため, VOP 選択アルゴリズムを使用しない場合, ハードウェアのスペックを有効利用できる条件の設定が困難となる.

符号化インターバルが平均符号化時間を上回るケースではほぼ 100～200 フレーム程度と, フレームロスの発生回数が安定している. movie1 における P-VOP の平均符号化時間はおよそ 305ms であることから, 符号化インターバルが 310ms 以降の条件下では(39)式に従ったほぼ一定値の遅延時間を設定することになる. この場合, 符号化インタ

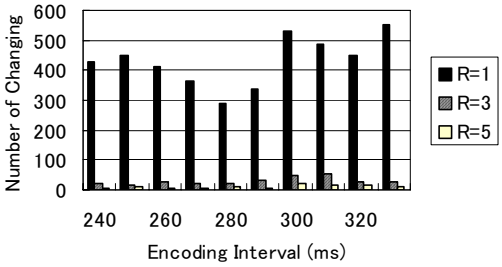


図 11　更新頻度 R に対する更新回数の違い

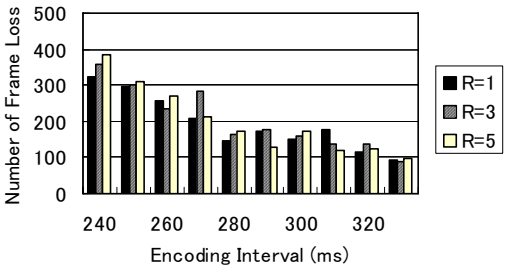


図 12　更新頻度 R に対するフレームロス数の違い

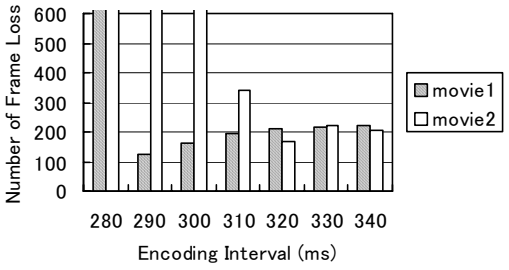


図 13　VOP 選択アルゴリズムを使用しない場合のフレームロス数

ーバルによらずほぼ一定の比率でフレームロスが発生している.

6.　今後の課題と展望

本稿では, 実時間コンテンツ編集システムに求められる技術として適切な同期処理の必要性について議論し, 最適なタイミング設定を行う方法を提案した. 動画・音声情報を実時間配信する場合, 音声情報の同期を厳密に取る必要があり, 各フレームの表示間隔が一定でなければ映像情報の表示に際して不自然な感じを与えるため, フレーム間隔を可能な限り一定に保った同期処理が必要になる. 本稿で提案する同期処理アルゴリズムは, VOP の符号化時間や各 VOP の頻度に合わせて遅延を設定するというもので, 符号化を行うハードウェアのスペックや状態に合わせて遅延を設定できるという特長を有する. 実験の結果, 係数を適切に設定することによってフレームロスやタイムスタンプの再設定の回数を減らすことができることが明らかになった.

今後の課題として, 通信に要する時間を加味した検討, および複数オブジェクト間の同期に関する検討を進めることが挙げられる.

参　考　文　献

1) 勝本道哲, 原田雅博, 中川晋一: D1 over IP による高品位動画像転送・蓄積システムの設計, 情報処理学会・マルチメディア通信と分散処理研究会報告論文集, No.95, pp.85-90(1999)
2) 杉浦一徳, 小川晃通, 中村修, 村井純: 民生用 DV を用いたインターネットビデオ会議システム, 情報処理学会誌, Vol.40, No.7(1999)
3) ohphone: http://www.openh323.org/docs/ohphone_man.html
4) FFmpeg: http://ffmpeg.sourceforge.net/
5) 三浦康之, 勝本道哲: 実時間環境におけるビジュアル符号化のための実証実験, 情報処理学会研究報告, 2002-DPS-100, pp.25-30(2002).
6) Yasuyuki Miura, Michiaki Katsumoto: An Overview of a Real-time Contents Edition System for MPEG-4, *Proc. of 2003 IEEE Pacific Rim Conference on Communications Computers and Signal Processing (PACRIM 2003)*, pp. 81-85(2003)
7) 三浦康之, 勝本道哲: 実時間コンテンツ編集システムの動画像符号化における VOP 選択アルゴリズムの提案, 情報処理学会論文誌, Vol.45, No.2, pp.498-508(2004)
8) ISO/IEC 14496: Final Draft International Standard MPEG-4, 1998.
9) 清末悌之, 湯田佳文: IP ネットワーク上のリアルタイム音声ミキシングに対してバッファサイズが与える影響に関する一考察, 情報処理学会論文誌, Vol.41, No.10, pp.2742-2751(2000)
10) 小峯隆宏, 勝本道哲, 丹康雄: リアルタイム多地点遠隔コミュニケーションにおけるデジタル音声処理機構の提案, 電子情報通信学会技術研究報告, IA2002-36, pp.103-107(2002)
11) Takahiro Komine, et.al.: Development of "high presence" video communication system –Trial experiment of the Next Generation real-time remote lecture–, *Proc. of the 16th International Conference on Information Networking*, Vol.II, pp.4C-4.1-4C-4.8
12) 栗田孝昭, 井合知, 北脇信彦: オーディオビジュアル通信における伝送遅延の影響, 電子情報通信学会論文誌 B-1, Vol.J-76-B-I, No.4, pp.331-339(1993)
13) ISO/IEC 14496-1: Final Draft International Standard MPEG-4: System, 1998.
14) ISO/IEC 14496-5: Final Draft International Standard MPEG-4: Reference Software, 1998.