

博士学位論文

スケーラブル動画像符号化における
ビットストリーム変換方式に関する研究

Research on Bitstream Transcoding
for Scalable Video Coding

2004年2月

早稲田大学大学院 国際情報通信研究科

永 吉 功

目次

第1章	序論	1
1.1	研究背景	2
1.1.1	スケーラブル映像通信を実現する符号化方式	3
1.1.2	映像符号変換処理(トランスコーディング)によるスケーラブル映像通信	4
1.1.3	動画像符号変換方式の分類体系	4
1.1.4	動画像符号変換方式に関する課題	8
1.2	研究目的	8
1.2.1	階層型伝送のための動画像符号変換方式に関する研究	9
1.2.2	伝送条件に応じた適応的変換制御方式に関する研究	12
1.3	本論文の構成と概要	12
第2章	トランスコーディングと階層符号化の融合によるスケーラビリティ	16
2.1	はじめに	16
2.2	階層符号化方式の概念	16
2.2.1	符号化器/復号器の複雑化と処理量の増大	17
2.2.2	階層化に伴う画質劣化(階層化劣化)	17
2.2.3	提供可能な品質の制限	17
2.3	トランスコーディングと階層符号化の融合によるスケーラブル映像通信	18
2.3.1	ストリーム分離器を用いた非階層から階層符号化情報への変換	18
2.3.2	ストリーム合成器による階層符号から非階層符号化への変換	19
2.4	従来方式との比較	19
2.4.1	SNR スケーラビリティとの比較	20
2.4.2	ビットストリーム更新スケーラビリティとの比較	20
2.4.3	提案方式の特徴	21
2.5	階層型動画像伝送のアプリケーション	21
2.5.1	ニュース映像の速報配信のための階層型伝送方式	21
2.5.2	映像監視システムにおける監視映像の階層伝送	22
2.6	まとめ	22
第3章	ビットストリーム分離・合成によるスケーラブル映像符号化方式	24
3.1	はじめに	24

3.2	ビットストリーム分離・合成処理器の構成	24
3.2.1	ビットストリーム分離器	24
3.2.2	ビットストリーム合成器	25
3.3	再量子化演算における係数値の変化特性の解析	26
3.3.1	再量子化演算処理および量子化制御方法	26
3.3.2	整数化処理に伴う切り上げ演算による影響	27
3.3.3	再量子化入力値のデットゾーン特性	29
3.4	再量子化特性を利用した差分映像情報符号化方式	29
3.4.1	量子化係数変化情報の符号化方法	30
3.4.2	復号方法	32
3.5	符号化性能評価	32
3.5.1	変換差分ビットストリームの総符号量に関する評価実験	33
3.5.2	ストリーム分離器多段接続評価実験	38
3.6	変換差分情報符号化方式の符号化特性解析	40
3.6.1	再量子化による係数変化情報量の導出	40
3.6.2	符号化特性結果	42
3.7	まとめ	44
第4章	ビットストリーム変換による多階層映像符号化方式	47
4.1	はじめに	47
4.2	多階層型高位階層符号による動画像伝送	47
4.2.1	非階層符号の変換による階層符号化情報の生成	47
4.2.2	階層選択性を有する高位階層符号の合成による高位階層映像の取得	48
4.2.3	高位階層符号の再合成による蓄積映像の品質の向上	48
4.3	基本原理	49
4.3.1	階層構造を有する量子化係数情報の構成方法	49
4.3.2	階層数の上限値	51
4.4	多階層対応型高位階層符号化方式の提案	52
4.4.1	階層情報への変換アルゴリズム	52
4.4.2	非階層符号の合成方法	57
4.4.3	部分合成ブロックの生成方法	57
4.5	評価実験	59
4.5.1	階層符号変換時の総符号量に関する評価実験	59
4.5.2	階層数に対する総符号量変動に関する評価実験	62
4.5.3	中間合成映像品質評価実験	63
4.6	まとめ	66

第 5 章	階層間独立性実現のための階層符号変換方式	67
5.1	はじめに	67
5.2	階層間非依存な多階層符号化方式	67
5.3	独立情報構造を実現する量子化係数差分信号の符号化方式	68
5.3.1	階層符号への変換方法	68
5.3.2	非階層符号化信号への合成方法	70
5.4	評価実験	72
5.4.1	階層数に対する総符号量特性	72
5.4.2	合成階層数と画質の関係	73
5.5	提案符号化方式の応用例	74
5.5.1	複数デバイスの利用による耐性の向上	74
5.5.2	高位階層信号成分の分割符号化による伝送路の有効利用	75
5.6	まとめ	76
第 6 章	輻輳制御の影響を利用したトランスコーダ変換レート動的制御方式	77
6.1	はじめに	77
6.2	TCP/IP 輻輳制御による影響を利用した変換レート動的制御	77
6.3	基準時刻情報に基づく帯域推定に関する基礎実験	78
6.3.1	想定する伝送モデル	78
6.3.2	実験条件	79
6.3.3	実験結果	81
6.4	提案方式	84
6.4.1	処理経過時間増加率導出制御 (Step 1)	85
6.4.2	処理経過時間収束目標値制御 (Step 2)	86
6.4.3	目標変換レート更新制御 (Step 3)	87
6.5	シミュレーション実験	88
6.5.1	目標変換レート変動特性	88
6.5.2	処理経過時間増加率の特性	93
6.5.3	復号バッファ内滞留時間に関する評価実験	94
6.6	まとめ	98
第 7 章	再量子化/再符号化処理共有によるマルチレート出力ビデオトランスコーダ	101
7.1	はじめに	101
7.2	マルチレート出力トランスコーダの従来方式	101
7.2.1	シングルレート出力トランスコーダ並列型	102
7.2.2	復号/逆量子化器共用型	102
7.2.3	処理コストの比較	103
7.3	再量子化/再符号化器共用によるマルチレート出力トランスコーダ	103
7.3.1	基本原理	103

7.3.2	処理器の構成	104
7.3.3	MB 符号変換処理手順	105
7.4	再量子化/再符号化処理回数の解析	106
7.4.1	平均再量子化, 再符号化処理回数の導出	106
7.4.2	平均再量子化, 再符号化処理回数の特性解析	108
7.4.3	平均再量子化, 再符号化処理回数の極限值	109
7.4.4	再量子化, 再符号化処理回数の最悪値	111
7.5	評価実験	111
7.5.1	トランスコーダ処理コストの導出	111
7.5.2	トランスコーダ内部処理時間分布	113
7.5.3	総処理量に基づく評価実験	115
7.6	むすび	119
第 8 章	結論	121
8.1	各章の結論	121
8.2	階層型動画像符号変換方式に関する総合評価	126
8.3	本論文の総括および今後の課題	128
	謝辞	130
	参考文献	131
	研究業績	137
	図一覧	146
	表一覧	148

第1章

序論

ブロードバンドネットワークの拡大や計算機環境の高性能化，およびその基盤となる半導体技術の発展により，音声・映像の異なる種類のメディア情報を統括的に扱うマルチメディアサービスの需要が急増している．特に，映像コンテンツはブロードバンドネットワークにおける中心的なコンテンツとして期待されており，動画広告や娯楽系コンテンツ配信，e-Learning に代表される教育分野等でのアプリケーションが期待されている [1]．これらのサービスの普及に中心的な役割を果たした技術の1つとして映像信号の圧縮符号化技術が挙げられる．

映像符号化技術は，膨大な情報量を必要とする映像信号中に含まれる冗長性を排除して，適切な情報量のデジタル符号により効率的に映像信号を表現する技術として発展してきた．映像符号化技術は当初より通信での応用を考慮して検討が進められた経緯もあり，再生機器間の互換性の確保のために技術の標準化の必要性が指摘された．そして，動き補償とDCTをベースとした符号化技術が1990年代初頭より次々と標準化された．1990年にはテレビ会議，テレビ電話を主要アプリケーションとして検討されたH.261が標準化され，続いて1992年にはCD-ROM等の蓄積メディアに1.5[Mbit/sec]の符号化レートで蓄積することを目的としたMPEG-1 [2]が標準化された．1996年には，次世代の通信，放送，蓄積メディアの統合と高画質化を目指してMPEG-1を拡張した汎用的動画像符号化方式としてMPEG-2 [3]が標準化された．MPEG-2はデジタル放送やDVDにおける映像フォーマットとして採用されており，今後のデジタル映像メディアに関連する領域における標準的な符号化技術としての地位を確立しつつある．1998年には，任意形状のオブジェクト単位での符号化を実現し，移動体通信やインターネット上でのビデオストリーミングへの適用を想定したMPEG-4が標準化された [4]．2001年12月にはMPEGとITU-Tとの共同作業で次世代の動画像符号化方式の標準化を行うJVT(Joint Video Team)が設立され，ITU-Tで規格化が進められていたH.26Lをベースとした新たな符号化方式の標準化作業が行われてきた．JVTで規格化される本方式は，MPEGではMPEG-4 Part 10 AVC(Advanced Video Coding)，ITU-TではH.264として2003年に標準化された [5]．MPEG-4 AVC/H.264はMPEG-2の約2倍の符号化性能を有し，標準符号化方式におけるMPEG-2以降最大の技術革新と言われている．

これらの標準技術の確立により，デジタル映像に関連する産業分野では技術の共有による関連機器の大量生産および低コスト化，および標準技術をベースとした映像通信システムの普及が実現した．特に，MPEG-2はデジタル放送における映像信号のデジタル符号

化方式として、MPEG-4は携帯端末やインターネット上の動画配信における標準的な符号化方式としてそれぞれ利用され、マルチメディアシステムの中核を形成する標準技術として位置付けられる。

標準符号化方式は動画圧縮符号化技術の研究開発の起爆剤としての役割を果たし、これらの標準技術に対抗した独自の高效率映像圧縮符号化方式が次々と開発されている。非標準の符号化方式としては、MicrosoftのWindows Media Video 9 [6]、Digital USAのXVD(eXtended Video Disc) [7]、On2TechnologiesのVP6 [8]などが代表的であるが、今後も独自のアルゴリズムを採用した様々な符号化技術が出現するものと考えられる。圧縮符号化技術の性能競争は、標準方式のみならず非標準の符号化方式も加わった形で多種多様な符号化フォーマットが乱立しつつある状況を生み出した。同時に、映像流通のためのプラットフォームの多様化も加速度的に進行しており、映像伝送に利用するネットワークシステム、アクセス伝送路帯域、再生機器の処理能力、記録装置の性能などの周辺環境において、今後様々な特性を持ったデバイスが出現すると考えられる。周辺環境の多様化/複雑化はマルチメディアシステムの適用領域の拡大とともに必然的に進行するものであり、今後もこの状況は継続すると考えられる。このような中で、映像コンテンツの生成者、提供者は、最適な映像情報の符号化フォーマット、配信手段の選択が求められるが、これは利用するハードウェア環境や伝送路特性に依存する上に最適値がユーザごとに異なることも考えられるため、最適解は容易に得られるものではない。今後のマルチメディアシステムの拡大とともに進行するシステムの多様化の流れの中では、コンテンツ提供者、サービス提供者へは多大な負担を要求することになる。そのため、アプリケーションごとの様々な要求に応えるための映像情報の生成手段、変換方式への需要が今後急速に拡大していくものと思われる。

本論文は、動画符号化技術の確立を背景に、動画情報の高機能伝送や次世代の映像配信システムにおいて必要な符号化動画のハンドリング方式についてまとめたものである。

1.1 研究背景

デジタル符号化映像を用いたアプリケーションの実現にあたり直面する課題として、アプリケーションの多様性に起因したシステム構成要素の多様性が挙げられる。すなわち、伝送路の特性、記録装置の性能、受信装置及びディスプレイ等の表示装置の特性に多様性を有するために、1種類の符号化方式であらゆる利用者に満足可能なサービスを提供することが困難な状況にある。そのため、アプリケーション領域の拡大に呼応して、様々な利用環境に適合した映像情報の提供手段の確立が急務となる。

インターネットを利用した映像ストリーミングにおいては、様々な伝送速度を有する利用者からのアクセスが発生する。伝送能力を越えた情報量を有する符号化映像を伝送した場合には、パケットロスや到着の遅れ等が発生して正常な映像再生が困難となる。このようなサービスにおいては、複数の環境を想定して1つの映像ソースを複数種類のパラメータで符号化処理を行い、複数のビットレート、フォーマットのビットストリームを配信源で個々に生成するアプローチが主流となっている [9] [10] [11]。その際、条件ごとに個別に符号化さ

れたビットストリームの中から配信時の条件に適合したストリームを選択することで本問題を解決する。しかし、上記方式は符号化パラメータごとに別々の符号化処理が必要なことから配信サーバにおけるコンテンツ生成のための処理量への要求が高く、さらに1つの映像番組につき複数のビットストリームを蓄積・監理する必要から蓄積媒体の利用効率の低下、コンテンツの一元管理の煩雑性等の問題点を有する。そこで、1つの映像ソースから利用環境に応じた様々な品質を持った符号化情報を動的に生成することで様々なデバイスの特性の違いを吸収し、多様な環境への対応を実現する技術としてスケーラブル映像配信が注目されている [12] [13]。スケーラブル映像配信はネットワークアーキテクチャの設計 [14] [15] と映像、音声メディアを対象とした信号処理と符号化方式 [16] [17] が有機的に融合することで実現可能な映像通信システムであるが、本論文では特に映像符号化方式に焦点を当てて、デジタル映像メディア情報のハンドリングにスケーラブル性(スケーラビリティ)をもたらすための、符号化映像信号に対する信号処理方式を研究対象とする。

1.1.1 スケーラブル映像通信を実現する符号化方式

文献 [18] では、動画像情報のスケーラビリティを”符号化映像情報の一部分を取り出して意味のあるシーンを再生できる機能”と定義し、このような機能を提供する一連の動画像符号化処理方式をスケーラブル映像符号化と呼んでいる。多様な利用環境のもとでは1種類の符号化方式であらゆる利用者の品質要求を満たすのは困難となる。このとき、符号化映像情報の全てを復号できなくても低い品質であるが何らかの映像シーンを再生可能な情報構造を符号化情報が有していれば、最適な品質選択を利用者側へ委ねることができる。利用環境に応じた品質選択機能は様々な品質要求への対応可能性をもたらす、符号化映像コンテンツの制作者への負担を軽減化を可能とする。

スケーラブル映像通信を実現するための映像符号化方式はスケーラビリティ機能の適用領域の観点から以下に大別される [13]。

1. 符号変換方式(トランスコーディング)
2. 階層符号化方式

1. は符号化ビットストリームへの信号処理に着目したものであり、配信サーバやネットワーク中継ノードにおけるルータ/ゲートウェイ等の上に符号化器と復号器の間に位置する処理器を設けて符号化ビットストリームの品質を変換する。すなわち、伝送/再生条件を満たすように符号化情報を変換することでスケーラビリティを実現する方式である。

2. は符号化ビットストリームの構造に着目したものであり、階層構造を有する符号化ビットストリームを符号化器により生成する。伝送系では伝送路の品質、伝送誤り率、割り当てられている伝送帯域等に基づき伝送可能な階層を選択し、再生機器においてはデバイスの処理能力に基づき再生可能な階層数を選択することで、環境ごとに最適な品質を生成する。

本論文では、上記に代表されるスケーラブル映像符号化方式の中から、1. に示す動画像符号変換を基盤とした符号化映像信号処理方式を研究対象とし、符号変換処理器のアーキテクチャ、変換レート制御方式に着目し、各種領域に付随する課題について研究する。

1.1.2 映像符号変換処理(トランスコーディング)によるスケーラブル映像通信

本方式は、符号変換器(トランスコーダ)を用いて、既に符号化された符号化映像情報(ビットストリーム)に対する符号変換処理(トランスコーディング)を施すことで、様々な品質要求に応じてスケーラビリティを実現する方式である。符号化段階で適切な品質が決定不可能な状況に適しており、高品質を提供するパラメータで符号化処理を行い伝送・再生・蓄積デバイスの性能要求に応じて適切な品質に変換する。

本方式では、符号化情報を生産する部分(符号化器)とスケーラビリティを提供する部分(トランスコーダ)が明確に分かれる。そのため、符号化器-復号器から構成される既存のシステムにトランスコーダをアドオンすることによりスケーラビリティが提供可能となる。すなわち、符号化器や復号器自身には特別な機構が不要であり、高い汎用性を持つことが特徴である。さらに、変換時の品質選択には符号化器の制約を受けずに任意のパラメータを与えることが可能であり、品質要求に対する柔軟性を有する。したがって、トランスコーディングによるスケーラビリティは、符号化器/復号器の構造の複雑化および処理量の増大の問題を解決し、高い柔軟性を提供する方式である。このような特徴に起因し、ニュース番組等で利用するための映像素材の放送局への速報配信のための符号化レート削減器、ビデオストリーミングシステムの配信サーバや多地点間のテレビ会議システムにおける中継ノードにおける受信端末の接続速度に応じた品質制御機構、等の様々な利用用途が期待される。一方、エンドユーザでにおけるアプリケーションにおいても、TVキャプチャボードを搭載したPCやHDD/DVDを利用したデジタルビデオレコーダの普及により、数[Mbit/sec]のビットレートを有するMPEG-2で符号化/記録された放送映像を、ほぼ同等の画質を保ちながら500[kbit/sec]~2[Mbit/sec]程度のビットレートで符号化可能なMPEG-4, DivX [19]等への再符号化を目的とした符号化フォーマット変換トランスコーダの実用化も急速に進行している[20][21]。このように、トランスコーディングは既存の符号化器/復号器の変更なしにワンソース・マルチユースを提供する解として、デジタル映像情報の流通の効率化への中心的役割をなす符号化映像情報処理方式としての適用可能性が高い。

1.1.3 動画符号変換方式の分類体系

符号化動画像に対するトランスコーディングは、圧縮符号化情報のデータ構造、符号化形式、コンテンツの表現形態に対する変換処理一般を意味する。本論文では、変換方式の種類の体系を図1-1のように整理する。また、各種トランスコーダが実現する機能およびアーキテクチャを表1-1に示す。トランスコーダの種類は、大きく分けて同一の符号化形式間で符号化パラメータの変換を行うトランスコーダ(Homogeneous型変換)、および異なる符号化形式間への変換を行うトランスコーダ(Heterogeneous型変換)[22]として分類される。また、トランスコーダアーキテクチャは、独立した復号器と符号化器の単純結合による方式(単純結合再符号化型トランスコーダ)、符号化モード、動きベクトルを再利用した空間領域内の再符号化による方式(空間領域トランスコーダ)、DCT係数領域内の再量子化方式(DCT係数領域トランスコーダ)の3方式に分類され、実現可能な変換機能はこれらのアー

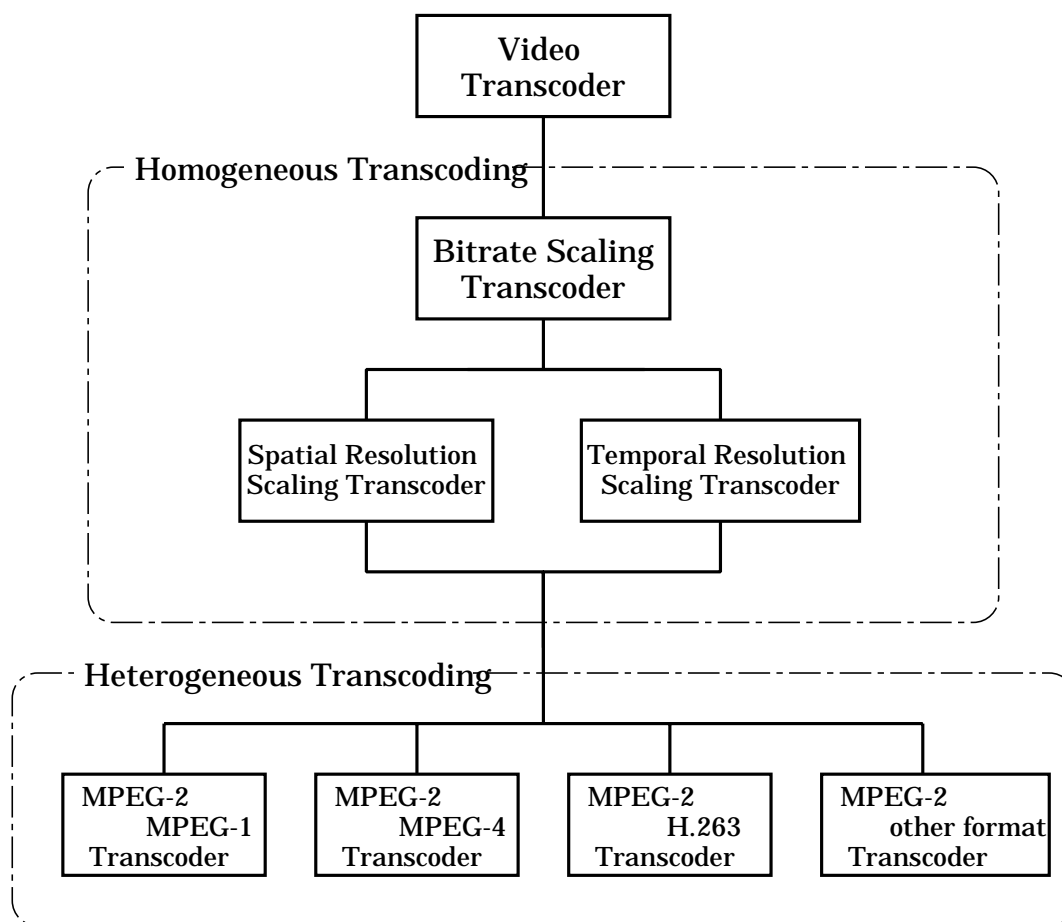


図 1-1: ビデオトランスコーダの分類体系

キテクチャに依存する。以下では、まず変換機能に着目したトランスコーダの種類体系について述べ、次に各種アーキテクチャの特性と実現される変換機能との関係について述べる。

符号化パラメータ変換トランスコーダ

スケーラビリティの実現解としてのトランスコーディングは、アプリケーションに応じて最適な品質が多様であることに起因した、様々な種類の品質要求に応えるための符号化品質の変換である。変換対象となる品質はビットレート、空間解像度、時間解像度の3種類へ分類され、この中でもビットレート変換機能(レートスケーリング)は最も基本的な機能となる。ハイビジョン TV 放送 (HDTV) から標準 TV 放送 (SDTV) [23] , MPEG-2 から MPEG-4 への変換 [24] [25] [20] 等では、空間/時間解像度や符号化形式の変換に伴って出力ビットストリームのビットレートが入力と異なることが考えられるからである。そのために、以降で述べる変換方式は全てこのビットレート変換機能を継承し、ビットレート変換トランスコーダ [26] [27] [28] [29] [30] [31] [32] [33] [34] を基本とする派生物と考えられる。

空間解像度変換トランスコーダ [24] [35] [22] [36] [37] , 時間解像度変換トランスコーダ [38] [39] [40] [41] は、空間、時間それぞれの領域で間引き処理(サブサンプリング)を行うことで大幅に情報量を削減して、表示能力、復号処理能力の制約のある端末での映像再生を可能とするための変換方式である。同一の符号化方式の範囲内での解像度変換としては、

表 1-1: トランスコードの変換機能とアーキテクチャ

トランスコードの種類		変換機能				アーキテクチャ		
		ビット レート	空間 解像度	時間 解像度	符号化 形式	再符号化	空間 領域	DCT 領域
パラメータ変換 トランスコード	ビットレート変換							
	空間解像度変換							
	時間解像度変換							
フォーマット変換 トランスコード	MPEG-2 - MPEG-1 変換							
	MPEG-2 - MPEG-4 変換							
	MPEG-2 - H.263 変換							

MPEG-2 MP@HL で符号化された HDTV サイズのビットストリームを MPEG-2 MP@ML の復号器で再生するための HDTV-SDTV 変換トランスコード [36] が代表的である。また、移動体通信での映像通信等の復号/表示能力が十分でない端末での映像再生を目的とした変換も考えられるが、このようなアプリケーションには、次節で述べる符号化フォーマットの変換が同時に行われることが多い。

フォーマット変換トランスコード

デジタル放送における符号化方式や PC 上での TV 放送の録画に用いるキャプチャボードおよびデジタルビデオレコーダにおける符号化器では MPEG-2 が主に利用されている。MPEG-2 による高レート符号化された放送コンテンツを、TV 受信機や PC 等の据置型の再生端末だけでなく携帯電話や PDA 等の移動端末を含めた多種多様な端末での再生を行うための符号変換としてフォーマット変換トランスコードが検討されている [24] [25] [20] [42] [43]。また、数 M[bit/sec] で符号化された MPEG-2 ビットストリームの再圧縮の目的で MPEG-4 等の高能率符号化方式への変換としても利用される [20]。さらに、MPEG-4 FGS(Fine Granularity Scalability) 等の階層符号化に対応したプロファイルを対象とする非階層符号と階層符号化方式間での変換方式 [44] [45] [46] 等がこれまでに検討されている。

表 1-1 では、MPEG-2 から MC-DCT 型の標準符号化方式 (MPEG-1, MPEG-4, H.263) への変換に限定したが、DV 等の非 MC-DCT 型への変換 [47] [48] や非標準方式への変換もここへ分類される。

トランスコードのアーキテクチャと変換方式への適用性

トランスコードは圧縮符号化されたビットストリームをある段階まで復号した後に再度符号化処理を行った再符号化情報として変換後のビットストリームを出力する。復号処理と再符号化処理の結合度により、以下に示すアーキテクチャへ分類される。

(1) 単純結合再符号化型トランスコード

本方式は、独立した復号器と再符号化器の従属接続により構成される。これは、復号器により画素領域まで完全に復号した後に、復号画像を再符号化器へ入力するものであり、既存

の復号器と既存の符号化器を接続するだけで容易に実現可能な方式である。特に、復号器と再符号化器は完全に独立して動作し、ビットストリームのシンタックスに依存する制御情報等も不要である。そのため、復号器と符号化器の組み合わせによりいかなる変換も実現可能であり、高い自由度を持つことが特徴である。本特徴に起因して、ビットストリーム構造が大幅に変化する異種符号化方式間のフォーマット変換トランスコーダに対して有効なアーキテクチャである。しかし、本方式は復号処理と再符号化処理の両方を同時に行うので膨大な処理量を要求する上に、繰り返し符号化による画質劣化、フレーム並び換えの遅延などの問題も有する。

(2) 空間領域トランスコーダ

単純結合による再符号化方式のトランスコーダにおける問題を解決するために、符号化器からの情報を再利用した復号器依存型の再符号化器を用いたトランスコーダが考えられる。本方式は、入力ビットストリームを空間領域まで復号した後に再符号化を行うものであるが、このときに復号して得られる符号化モード制御情報を再利用する。再符号化時のピクチャタイプを復号画像のピクチャタイプと同期をとることで、繰り返し符号化による画質低下及びピクチャの並び換えに起因した遅延を回避し、動きベクトルの再利用により膨大な演算量を要求する動きベクトル検出処理を省略可能となり、処理量の問題も解決可能となる。

本方式は、符号化モード情報の依存性により、異種フォーマット間の変換へ適用するためには両方式での符号化シンタックスの違いを考慮したモード情報の変換処理を必要とする。また、解像度変換トランスコーダに適用する場合に、解像度の変化に伴う動きベクトル補正演算 [49] [23] [41] を利用することで、再検出よりも少ない演算量で変換後の解像度における動きベクトルを算出する。

(3) DCT 係数領域トランスコーダ

本方式は、復号器と再符号化器との結合度を高めた方式であり、入力ビットストリームを可変長復号、逆量子化して DCT 係数領域まで復号し、得られた DCT 係数信号に対し再量子化処理、可変長符号化処理を行って変換後のビットストリームを出力する。この場合、マクロブロック以上のレイヤに属する符号化情報は一部分を修正するだけでほとんどを再利用可能となり、基本的にはブロック層内の DCT 係数符号のみを変換対象とする。本方式は、DCT/IDCT および動き補償処理が省略されることで少ない処理量で実現可能であり、動き補償参照用のフレームメモリを必要としないことから要求メモリ量の削減も可能となる。本特徴により、TV 会議システムなどにおける映像レート変換器などリアルタイム性が要求されるアプリケーションに対する有効性が高い。しかし、符号化器/復号器間の参照画像の不一致に起因するドリフト誤差による画質劣化が発生する。また、空間画素信号が扱えないのでビットレートの変換しか行えず、結合度が高いことに起因して対応可能な変換機能は制限される。

ここで述べた 3 つのアーキテクチャの特性は表 1-2 のようにまとめられる。

以上に述べた各種トランスコーダは動画像信号のみを対象とした処理方式の体系であるが、単一メディア情報だけでなく、H.323/SIP 等のマルチメディアシステムを対象とした異なるシステム間の相互接続を実現するためのトランスコーダ [50] [51] [52] も検討されている。

表 1-2: トランスコーダアーキテクチャの特性比較

	再符号化	空間領域	DCT 領域
結合度	小 (疎結合)	中	大 (密結合)
画質	×		×
遅延量	×		
自由度			×

1.1.4 動画像符号変換方式に関する課題

これまでに議論されてきたトランスコーダは、伝送路帯域、再生端末の性能が十分でなく符号化器で生成された符号化動画像をハンドリングできない場合に、これらの媒体で処理可能な情報量へスケリングすることを目的としており、使用環境の制約に起因して初めて成立する概念である。変換処理は、高ビットレートから低ビットレートあるいは高空間/時間解像度から低解像度への間引き処理に他ならず、間引きに伴う高周波信号成分の損失、分解能の低下を避けることはできない。低レート化、低解像度化に伴う品質劣化を最小限にすることは共通した重要課題であるが、本質的に情報量を削減している以上は、変換前の符号化時点での映像品質が提供されない点は変わりが無い。また、符号化済の量子化誤差を含む画像信号に対する再度の量子化演算には、非圧縮信号に対する符号化処理とは異なる特有の量子化歪みの特性があり、直接符号化と同等の品質の実現は困難となる [26] [53] [54] [55]。

適応的なスケール変換は利用環境の違いをトランスコーダ側で吸収し、多くの端末へ符号化動画像を提供可能となり動画像情報の流通の活性化が期待される反面、符号化器と復号器との中間に情報変換処理が介在することで符号化時点の品質の保証がなされなくなる問題を有する。すなわち、配信側が意図する品質が利用者へ提供される保証がなく、高画質性を前面に出すコンテンツ配信サービスでは問題となる。

トランスコーダの検討は情報量を削減して信号成分の欠落を許容した単一方向の変換器として終息するものではない。ビットストリーム生成時点の符号化情報の全てを確実に伝送することを前提として映像伝送を支援するための情報構造変換方式が、トランスコーディングが意味する”伝送/再生メディアの特性に応じた符号化情報変換処理”に他ならない。

1.2 研究目的

今後のブロードバンド環境におけるマルチメディアアプリケーションの拡大により、伝送資源の有効利用した高品質動画像通信方式などで映像の配信形態が高機能化が予想される。本論文では、様々な形態を有する映像伝送システムの要求に対して、符号化映像情報の符号変換(トランスコーディング)を中核とした実現解について考察し、その基盤となる変換処理方式および制御方式について研究する。特に、符号化情報成分の損失を伴う単方向の変換方式から脱却し、初回符号化時点と同等の品質の符号化情報を伝送することを前提とした符号変換方式の実現を目指す。

1.2.1 階層型伝送のための動画像符号変換方式に関する研究

1.1.4 で述べた課題に対して本研究では、階層符号化情報を用いた動画像情報の階層的伝送によって、狭帯域伝送路環境で高品質な符号化情報を効率的に伝送するための符号化信号変換方式について検討する。特に、階層符号化方式をトランスコーダに応用した非階層符号から階層符号への変換を行う階層型動画像符号変換による実現解について考察する。

階層符号化方式は、符号化処理の段階で複数品質の復号映像を取得可能な情報構造を持つビットストリームを生成することで、再生端末側での品質選択性を実現する符号化手法である。階層符号化は、基本となる画質を与える基本階層符号と、基本階層画像の品質を向上させるための付加信号となる高位階層符号から構成された階層的な構造を有したビットストリームを生成し、これにより、全ての符号化情報を復号できない場合でも基本階層のみから基本的画質のもとで映像再生が可能となる。基本階層符号は、十分な伝送路帯域や復号能力がない再生端末においても映像再生を可能とするためのものであり、これはトランスコーダでのビットレートのスケールと利用目的が一致する。このとき、レートスケール後のビットストリームを基本階層と見なして、高位階層に相当する画像信号を符号化出力する機能をトランスコーダに設けることで、階層符号化と同等の機能が実現可能となる。

蓄積コンテンツなどの有限時間の符号化動画像を伝送において受信符号化情報の蓄積を前堤とした場合、先頭時刻から終了時刻までを複数回に分けて伝送可能である。このときに伝送動画像情報を階層型の符号化構造へ変換し、初回伝送の段階では動きベクトル情報や低周波 DCT 係数情報などで構成される基本階層を伝送することで、全ての符号化情報を受信しなくても先頭から終了まで意味のあるシーンを再生可能となり、映像内容の早期把握が実現される。また、二次伝送段階で高位階層信号を伝送し、受信側で基本階層信号へ付加することで元の品質の映像の取得が可能である。したがって、階層型動画像符号変換を用いて低品質信号から高品質信号へと段階的に伝送することで、1 回の伝送効率が低い場合でも映像把握に要する時間の削減と高品質符号化情報の取得の両方を同時に満たすことが可能となる。

階層符号化方式は MPEG-2, MPEG-4 の標準化作業当時より検討されてきたスケーラビリティの実現方法であり、両符号化標準では階層符号化機能を持つ専用のプロファイルがサポートされている。しかし、符号化/復号器の汎用性、符号化/復号処理の複雑化等の問題があり広く普及するには至っていない。上記で論じた変換後画像信号に対する付加信号の出力機能を備えた階層型動画像符号変換処理は、ビットレート変換と階層符号化の両手法を融合した新たなスケーラビリティの実現解となり得るものである。

文献 [12] では、DCT 係数領域内再量子化方式によるトランスコーダにおいて画質、処理速度、遅延量の観点から要求性能について考察し、基本となるアルゴリズム及び符号量制御方式を提案した。本研究では、文献 [12] で確立したビットレート変換を実現するトランスコーダを拡張して、レート変換処理で失われる信号成分への補償信号を保存し再利用することで、変換前と同等の符号化情報の再現を可能とする符号変換方式について研究する。

ビットストリーム分離・合成によるスケーラブル映像符号化方式の研究

トランスコーダを用いた符号変換処理において、入力符号化情報の変換処理に伴う映像信号の変化成分情報を利用することが可能である。DCT 係数領域内の再量子化によりレート削減をトランスコーダにおいては、入力符号化情報から復号された量子化係数情報と、出力符号化情報へ再符号化する再量子化係数情報との間の差分信号を出力符号化情報とは別に保存し再利用することで、レート削減後の出力符号化情報を基本階層、差分信号の符号化情報を高位階層とした階層符号化機能がトランスコーダによって生成可能となる。

そこで、本研究ではトランスコーダの変換処理前後間の差分情報が階層符号化情報の生成に利用可能な点に着目し、符号化器/復号器の外部機能によるスケーラビリティの実現を目的とした、映像差分の再利用によるスケーラブルな映像蓄積・配信を実現するための新たな符号化方式を提案する。まず、トランスコーダを利用したスケーラビリティの実現方法を示し、非階層のビットストリームから基本階層、高位階層への分離を行うビットストリーム分離・合成の概念を述べる。次に、トランスコーダ内の再量子化処理により発生する量子化領域での DCT 係数差分に着目し、係数値の変化の性質を考察することで、これに基づく係数差分の符号化方法と、再量子化前の元の係数情報の復元方法を提案する。さらに、符号化効率に関する数値解析実験を行い、係数差分信号の符号化方式の性能を理論的に評価する。

ビットストリーム変換による多階層映像符号化方式の研究

再量子化前後間の差分信号を利用した階層符号への変換方式では基本階層と高位階層の2種類の品質が選択可能となる。ここで、高位階層符号化情報が、その一部分を抽出したものと基本階層を合成して意味のある映像シーンを取得可能な情報構造を有していれば、基本階層と高位階層の中間的な品質を生成可能となる。すなわち、係数情報中の任意の点までを復号して映像信号を再現可能な構造を高位階層符号に持たせることで、MPEG-4 FGS 符号化方式のような任意の品質を取得可能な階層符号化方式が実現される。これにより、高位階層符号の伝送において要求伝送時間と回線速度の制約がある状況において、伝送可能な情報量のみを選択することで、高位階層の伝送にスケーラビリティをもたらすことが可能となる。

本研究では、2階層への階層符号変換方式を拡張し、高位階層符号化情報に部分的に抽出/合成可能な情報構造を持たせることで、多階層/多品質に対応する階層符号への変換方式について検討する。まず、2種類のステップサイズで再量子化された再量子化出力ブロック同士での有意係数の分布形態について考察する。本考察に基づき、係数レベル値により階層化して高位階層符号化情報に階層構造を持たせることで、複数階層/複数品質を取得可能な、MPEG-4 FGS と同種類の機能を実現する符号化方式について検討する。

階層間独立性を有する階層符号化情報変換方式の研究

高位階層が複数の階層で構成される多階層構造をもつ階層符号化方式において、従来より検討されている方式では各階層の符号化情報が直下の階層の差分信号を符号化したものとなっており、各階層が下位側の階層へ依存した情報構造を有している。この場合、 n 番階層

の品質の取得のためには、当該階層より下位側の全ての階層が復号可能である必要がある。ここで、中間に位置する階層が復号不可能な場合、それよりも上位側の階層情報が無効となる。そのため、階層符号化を利用した映像配信方式の検討では、各階層ごとにエラー耐性の強度を制御する必要があることが前提となっている。このとき、基本階層以外の全ての階層が、基本階層を直接参照する情報構造を有していれば、基本階層を除いた任意の階層が復号不可能な場合でも、残りの階層からの復号・再生が可能となる。

本研究では、基本階層との差分信号を互いに独立した情報構造を有する複数のビットストリームとして符号化することで、高位階層間の依存関係を排した階層符号化方式について研究する。特に、基本階層との差分信号を1次元ベクトルで表現した量子化係数列において、各係数のインデックス値に基づき出力先の階層を決定することで、階層間の独立性を実現する多階層符号への変換方式について検討する。

再量子化器/再符号化器共有によるマルチレート変換アーキテクチャに関する研究

上記までに述べた各種階層符号変換方式では、基本アーキテクチャとしてDCT係数領域内再量子化型のトランスコーダを適用している。本研究では、DCT係数領域内の変換処理の拡張性について焦点を当て、特に、ビットレートが異なる複数ストリームへの同時変換機能を実現するためのアーキテクチャについて研究する。異種帯域型ネットワーク上でビデオストリーミングによるマルチキャスト配信を行う場合、個々の受信端末の接続速度が異なるために、各受信端末での再生開始時刻に差が生じて、特に多地点テレビ会議などの通信では問題となる。そのため、受信端末が接続するネットワーク帯域に応じた品質の映像情報の提供が必要とされる。これを実現するために、配信サーバやネットワークの中継ノードのルータ/ゲートウェイ等で、要求帯域に応じてビデオストリームのビットレートを変換するビデオトランスコーダが提案されている [56] [26] [27] [57] [32]。

ビデオトランスコーダは1本の映像ソースから要求品質に応じたビデオストリームを柔軟に生成可能であり、ワンソース・マルチユースを実現する有効な解のひとつである。従来、変換処理における変換映像品質の向上 [54] [53] [33] [58]、変換処理の効率化を実現するアルゴリズム [26] [31] 等の報告があるが、すべて、1入力1出力型のトランスコーダ(シングルレート出力トランスコーダ)による方式である。利用者間の接続速度や再生端末の処理能力の環境の違いを考慮したストリーミングシステムを実現するためには、1本の入力ストリームを品質の異なる複数ストリームへ同時に変換する機能(マルチレート変換機能)を持つトランスコーダ(マルチレート出力トランスコーダ)の実現が課題となる [59]。

マルチレート変換機能の実現方法として、ユーザの品質要求毎にシングルレート出力トランスコーダを適用する方法がある。しかし、変換処理に要する処理量がユーザ数に比例して増大し、トランスコーダを実装する配信サーバや中継ノードの負荷が増大する。これを解決するには、処理量増加の少ない効率的なマルチレート変換アルゴリズムが必要である。そこで本研究では、複数ストリームへの変換を行うマルチレート出力トランスコーダにおいて、複数の出力間で共通した処理を1つの処理器に統合して行い演算リソースを有効利用することで出力レート数増加に対しても処理量を収束可能なアーキテクチャについて検討する。

1.2.2 伝送条件に応じた適応的変換制御方式に関する研究

1.2.1 で述べた階層構造への変換方式は、伝送帯域が制限された環境で映像内容の取得に要する時間的制約がある中で動画像情報の全ての信号成分を効率的に伝送するという要求に対する解を与えるものである。本方式の特徴は、トランスコードの柔軟性に起因して、符号化処理時ではなく伝送処理の段階において利用可能な帯域に応じて動的に階層間のビットレート配分を行いながら階層型伝送を行うことにある。実在する映像伝送ネットワークシステムにおいて本方式の有効性を実証するためには、変換アルゴリズムだけでなく帯域の取得とそれに応じたビットレート制御方式の確立が必要となる。そこで、本研究では変換制御方式に着目し、特に帯域変動を有する QoS 非保証型の伝送路環境において階層型動画像伝送を適用するための動的レート制御方式について研究する。

帯域変動対応型トランスコード動的変換レート制御方式に関する研究

本研究では、IP ネットワークなどの帯域非保証型ネットワーク上でのストリーミングシステムにトランスコードを適用した帯域適応型のスケーラブル映像伝送の実現を目的として、変動する伝送帯域に応じて適切な変換レートの制御を行うトランスコード動的レート制御方式について研究する。伝送プロトコルとして TCP/IP を想定し、輻輳制御を起因としたアプリケーション領域上で観測される間接的影響を制御に利用し、レート制御に用いるパラメータ取得のための特別な構造をプロトコルに対して要求せずに既存の実装のもとで伝送路の状況に応じた変換レートの動的制御を実現する方式について検討する。

TCP/IP 等の輻輳制御を有するネットワークでは、伝送能力を越えた情報量が発生した場合、パケットロスや伝送遅延時間の変動などの輻輳の発生を回避するための制御(輻輳制御)が働く。これにより、伝送路上の混雑状況やルータ/受信端末上の受信バッファの状況に応じてデータの転送速度の制御(フロー制御)およびパケット再送制御(エラー制御)が働き、結果として転送される情報量はその時刻における利用可能帯域による制約を受ける。トランスコードの出力をネットワークインタフェースに接続した場合、上記の輻輳制御に起因した転送速度の低下現象による、出力ビットストリームの送待時間が発生し、トランスコード自身の処理能力(所定の時間区間内における変換処理符号量)へ影響を及ぼすと考えられる。

本研究では、ストリーム上の時刻を表し映像と音声の同期制御の基準時刻を与える SCR(System Clock Reference) 信号を制御に利用することに着目する。MPEG-1 System Stream および MPEG-2 PS に準拠する多重化ストリームを対象として、変換処理中に入力ビットストリームの各パックヘッダ中の SCR を復号するとともにこの時点での実際の時刻を観測する。そして、観測時刻値から経過時間を算出し、これを復号された SCR 値を基準にして評価し、本評価結果を利用して変換ビットレートを導出する制御方式について検討する。

1.3 本論文の構成と概要

本論文では、階層符号化技術を応用し、トランスコーディングを用いた動画像情報の階層的伝送により狭帯域伝送路上で高品質な符号化情報を効率的に伝送するための符号変換方式

について研究する．特に，符号化情報成分の損失を伴う単一方向の変換方式から脱却し，初回符号化時点と同等の品質の符号化情報を伝送することを前提とした符号変換方式の実現を目指して，変換処理アーキテクチャ及び制御方式についての検討を行った．第2章は階層符号変換の概念を述べたものであり，第3章～5章で具体的な符号化処理の方式を検討した．また，第6章は階層間の符号化制御方式を検討し，第7章では複数ストリーム変換への拡張性について検討を行った．各章ごとに概要を述べる．

第2章では，動画像情報の階層型伝送により伝送帯域の有効に利用する映像情報の伝送方式について検討し，非階層符号化情報と階層符号化情報間の相互変換を行う処理器を用いて階層型動画像伝送を実現する方式について述べる．まず，従来の階層型の符号化器/復号器による方式には，符号化/復号処理量の増大，階層化に伴う画質劣化，符号化/復号器の汎用性，の課題があることを述べる．本課題を解決するために，階層符号への変換を行うストリーム分離器および非階層符号への変換を行うストリーム合成器による階層符号変換方式を提案する．また，従来の階層型符号化/復号器による方式との比較を行い，階層化処理を符号化器/復号器とは別に設けることで階層符号化機能に対応していない汎用の符号化器/復号器のもとで階層型動画像伝送を実現可能であり，本機能がビットレート変換トランスコーダを拡張して実現可能であることを述べる．さらに，応用例としてニュース映像の速報配信システム，監視映像の階層伝送方式における本方式の適用例について述べる．

第3章では，DCT 係数領域内再量子化方式による MPEG トランスコーダに着目し，トランスコーダの入力，出力間の映像差分信号の符号化機能を追加することで，第2章で述べた階層符号への変換を実現する方法を提案する．再量子化演算においては，再量子化演算特有のレート歪み特性を考慮して再量子化ステップ値を離散的な値へ制限する再量子化ステップ値禁止領域制御の適用を前提とし，再量子化演算時の DCT 係数値の変動の特性を (i) 整数化に伴う丸め演算処理の影響，(ii) 再量子化入力値のデットゾーン特性，の観点から考察する．その結果，再量子化出力係数ブロック内の非ゼロ係数 (有意係数) は必ず 0 でない変化差分値を伴い各係数に対応した差分信号を持つことを示す．さらに，再量子化出力係数ブロック内の有意係数をジグザグスキャンの順番で参照すれば，各差分信号に対応する係数が明らかとなるために，これらの係数に対する差分信号についてはランレンクス情報などのブロック内位置の特定のための情報が不要となることを示す．すなわち，再量子化処理前後間での DCT 係数ブロックの変化は再量子化出力係数ブロックと再量子化ステップ値により類推可能であり，再量子化出力ブロックにおけるゼロ係数の領域に対してのみ，差分信号を 2 次元ランレンクス符号化して，有意係数に対する差分信号はその値のみを符号化するだけで十分であることを述べる．そして，本方針に基づく係数の変化差分値の係数の変化差分値の分類による変化成分情報の符号化方式と，これを用いた再量子化前の DCT 係数ブロックの復元方法を提案する．また，評価実験から提案方式により基本階層と高位階層の合計符号量が入力符号量以下となり，高い符号化効率を実現可能であることを示す．提案方式の符号化性能の理論的評価として，DCT 係数をラプラス分布でモデル化して基本階層および高位階層の係数情報の符号化に必要な符号量を導出することで符号化効率を評価し，提案方式による階層構造への変換処理は，同一の係数情報を MPEG-2 の 2 次元ランレンクス符号化方式

よりも少ない符号量で符号化可能であり，再圧縮の効果があることを証明する．本特徴により，DCT と量子化をベースにした符号化方式における可変長符号化の新たな方式としての適用可能性を示す．

第4章では，第3章で提案した2階層の階層符号変換方式を拡張し，高位階層符号化情報を部分的に抽出/合成可能な符号化構造を持たせることで，多階層/多品質に対応する階層符号変換方式を提案する．まず，再量子化ステップ値禁止領域制御を適用したときの，隣接するステップ値で再量子化された2種類の再量子化出力ブロック間での有意係数分布の変化について考察する．本考察から，基本階層量子化係数ブロック(基本階層ブロック)と入力量子化係数ブロック(入力ブロック)間の中間係数ブロックは，基本階層ブロックに対して入力ブロック内の所定の係数レベル値を大きい方から降順に印加した量子化係数ブロックと同等の有意係数の分布となることを示す．そしてこれに基づき，再量子化前後間での差分量子化係数信号を入力係数ブロックの係数レベル値ごとに別々にランレングス系列を形成して符号化することで，1つの系列が1階層に対応した階層構造符号化方式を提案する．提案方式は，1つのランレングス系列内に符号化される係数レベル値が1種類であることに着目して，系列の境界符号の検出によりレベル値が自動的に算出することで符号量を削減可能な点を特徴とする．基本階層と高位階層との合計符号量の評価実験より，高位階層の符号化構造において階層数の細分化を行っても2階層方式と同等の総符号量特性となり，多階層化に伴う符号化効率の低下を回避可能であることを示す．また，所定のランレベル系列のみを選択的に合成した中間合成映像の画質評価から，提案方式により基本階層に対して段階的に品質を向上させて，最終的には汎用符号化器より生成される非階層符号化情報と同等の画質及び符号量を達成可能となることを示す．ただし，非階層ビットストリームの再量子化演算処理により同等の画質を実現した場合との比較から，中間合成段階では多くの符号量が発生し，品質向上の中間段階における符号化効率の改善の課題があることを述べる．

第5章では，高位階層間に独立な符号化構造を実現するための階層符号化方式を提案する．3階層以上の階層構造を有する階層化方式において，各高位階層が直接基本階層を参照することで，高位階層同士の情報構造の依存性を排除した階層化符号化方式について述べ，高位階層信号の伝送順番に自由度を持たせた新たな階層符号化情報の伝送方式を示す．このとき，高位階層間の依存関係の排除による総符号量の増大の課題があることから，階層間の依存関係を用いずに高符号化効率性を実現するための符号化方式を提案する．提案方式では，量子化係数の差分信号をジグザグスキャンして1次元系列に変換した係数列において，各係数の位置インデックス値を階層数で除算した余りにより出力先の階層を決定することで高位階層間の独立性を実現する．最後に，符号量と画質に関するシミュレーション実験により提案方式の有効性を示す．

第6章では，提案符号変換方式における変換制御方式に着目し，特に，伝送帯域の時間的変動に対して適応的に変換ビットレートの制御を行う動的制御方式について研究する．伝送プロトコルとしてTCP/IPを想定し，TCP/IPによる輻輳制御を起因としたアプリケーション領域上で観測される間接的影響を利用した制御方式を検討する．まず，輻輳制御発生時の送信待ち時間によりトランスコード処理性能へ影響を及ぼす可能性があることを述べる．次

に、本考察の妥当性の検証として、伝送路帯域と変換レートの差異に起因するトランスコード処理経過時間の影響に関する基礎実験を行う。基礎実験では、経過時間の観測値を入力ストリーム内の同期制御信号である SCR(System Clock Reference) の復号値を基準として評価し、その結果、SCR 復号値と SCR 復号時点での処理経過時間は一定の比率を保ちながら増加し、その増加の割合が伝送路帯域と変換レート間の比率の一次関数として定式化可能であることを示す。次に、基礎実験で定式化された特性関数に基づく変換レートの動的更新制御方式を提案する。本制御方式は、(ステップ 1) 処理経過時間増加率の算出、(ステップ 2) 処理経過時間増加率の収束目標値の算出、(ステップ 3) 特性関数を用いた変換レート更新制御、の 2 つのステップで構成される。ステップ 1 では、トランスコード処理経過時間と入力ストリーム中の SCR 復号値の増加量の比率(変化の割合)として処理経過時間増加率を求めて、伝送帯域の推定に利用する。ステップ 2 では、処理経過時間と SCR 復号値を比較して両者の偏差からステップ 3 における処理経過時間の収束目標値を導出する。ステップ 3 では、処理経過時間増加率と基礎実験で得られた特性関数を用いて目標変換レートを更新する。このとき、伝送帯域と変換レートの均衡点へ指数関数的に収束させるように目標変換レートを制御して、処理経過時間増加率を収束目標値へフィードバックする。最後に、計算機シミュレーションにより目標変換レートと処理経過時間増加率の変化、各ピクチャの復号バッファ内滞留時間の各観点から提案制御方式を評価する。その結果、伝送路帯域が未知であっても自動的に適切なビットレートへの制御が実現可能であり、また伝送帯域の変動に追従可能な方式であることを明らかにする。特に、ステップ 3 における収束のための時定数により提案方式の振る舞いを制御可能であり、本パラメータにより定常状態での安定性と帯域変動時の応答特性に大きな違いが現れ、最適な時定数の導出が今後の課題となることを述べる。

第 7 章では、第 3 章から第 5 章で述べた DCT 係数領域内再量子化による基本アーキテクチャを拡張し、複数ビットストリームへの変換(マルチレート変換)を効率的に実現可能な変換アーキテクチャについて研究する。変換処理がマクロブロック(MB)間で独立であり、再量子化出力 MB は再量子化ステップ値によって一意に求められることを利用して、復号および逆量子化器をそれぞれ 1 つの処理器へ共有するだけでなく、等しい再量子化ステップ値を有する出力間で再量子化/再符号化処理器を共用することでマルチレート変換時の処理量を低減可能なアーキテクチャを提案する。また、MB あたりの再量子化処理回数に関する理論的考察により、出力ビットレート数に対する変換処理量を定式化可能であることを示す。最後に、これに基づく理論的評価と評価実験から、再量子化/再符号化器の共用が出力数に対する処理量を収束させて、単一レート出力時の約 2~4 倍の処理量で任意の出力数に対応可能な方式であることを示す。これにより、第 3 章から第 5 章の各章における基本アーキテクチャは、複数ストリームへの変換に必要な処理量の増加を抑制可能であり高い拡張性を有する方式であることを述べる。

第 8 章は結論であり、本論文で得られた成果を総括する。

第2章

トランスコーディングと階層符号化の融合によるスケーラビリティ

2.1 はじめに

本章では、階層符号化方式の概念をトランスコーダに応用し、動画像情報の階層的伝送によって高能率伝送を実現する符号化情報構造変換方式の概念について述べる。

まず、階層符号化方式の概念と問題点について述べ、次に、変換前と変換後の間での映像信号の差分信号を利用することでトランスコーディングを階層符号化に応用し、階層符号化と同等の機能をトランスコーダの拡張により実現する方法を述べる。これに基づき、本研究で階層符号化とトランスコーディングの両方式を融合した新たなスケーラブル映像通信の実現解を提案する。

2.2 階層符号化方式の概念

階層符号化方式は符号化映像信号の情報構造にスケーラビリティを持たせて、階層型の情報構造を有するビットストリームを符号化器より生成し、伝送/再生系の特性に応じて復号/再生する階層を選択することで、品質選択性を提供するものである。階層符号化方式においては、階層構造を有する専用の符号化器を用いて基本となる品質を提供する符号化情報（基本階層）と基本階層画像の符号化歪みを低減させて品質を向上させるための補助信号の符号化情報（高位階層）に分けて符号化処理を行い、同一ビットストリームから複数品質を提供可能な情報構造を持たせることで品質選択性を実現する。階層符号化により実現される機能として、以下の項目が挙げられる [18] [60] [61]。

- 要求に応じた空間解像度，時間解像度，符号化歪みなどの画質選択機能の実現
- 後方互換性の実現
- 優先制御の実現．基本画像情報を優先情報として高品質画像情報を非優先情報と考え、高画質画像情報が破棄された場合でも、低画質画像を代用画像として再生することで伝送エラーへ対処する [62]。

一方、階層符号化には以下に示す問題点が指摘されている [12]。

1. 符号化器/復号器の構造の複雑化と処理量の増大
2. 階層化に伴う画質低下 (階層化劣化)
3. 提供可能な品質の制限

以下に、これらの問題点について詳述する。

2.2.1 符号化器/復号器の複雑化と処理量の増大

階層符号化方式は、ビットストリームの生成段階で階層化処理を行うため階層構造を有した専用の符号化器/復号器を必要とし、階層ごとに並列に符号化/復号処理を行う必要がある。符号化処理においては、入力画像信号を階層ごとに分岐させて、それぞれを動き補償予測、DCT により DCT 係数信号を生成する。基本階層は DCT 係数信号を粗く量子化した後にエントロピ符号化により符号化処理を行うが、高位階層については、基本階層を逆量子化して得られた逆量子化係数と動き補償予測信号を DCT して得られた DCT 係数との差分信号を、細かいステップサイズで量子化し、可変長符号化処理を経て高位階層符号を生成する。復号処理では基本階層符号と高位階層符号化情報を並列的に可変長復号、逆量子化して両階層の逆量子化係数信号を加算した後に IDCT で画素領域に復元する。このように、符号化器/復号器ともに、階層毎に可変長符号化/復号、量子化/逆量子化、DCT/IDCT の各処理を有するために、階層構造を持たない符号化器/復号器に対して約 2 倍の回路規模となり、各構成要素間の接続が複雑になる。

2.2.2 階層化に伴う画質劣化 (階層化劣化)

階層符号化器により生成されるビットストリームの品質は非階層のものよりも低くなることが知られている。階層化処理に伴う符号化効率の低下は階層化劣化 [63] として知られている。すなわち、複雑な符号化/復号器を用いて、通常より多くの演算量を費やして階層構造を持つビットストリームを生成しても、演算量に見合う品質が得られない問題点を有する。文献 [63] では、階層化劣化の特性解析に基づき、基本階層と高位階層間の符号量配分比率により階層化劣化関数を定式化し、これを最小化する配分比率を動的に制御することで階層化劣化を抑制可能な符号量制御方式を提案している。しかし、符号化処理の段階で非階層時の復号画像を生成しないため非階層方式での映像品質を参照することができず、また、高位階層の復号画像に含まれる符号化歪みには基本階層と高位階層の双方での量子化演算に依存した特性になる点において、階層符号化方式のアーキテクチャ自身が非階層方式と全く同等の信号品質を保証するものとはなっていない。

2.2.3 提供可能な品質の制限

各階層における符号化条件は符号化段階で決定されるために、階層ごとの提供される品質に関しては柔軟性の課題を有する。すなわち、要求条件が符号化時点と異なる場合にはス

ケーラビリティを提供するのが困難であり、提供品質に対する柔軟性の欠如が指摘されている [12]。本問題に関して、MPEG-4 の標準化過程において高位階層信号の情報構造を細かく階層化した符号化方法を適用することで連続的な品質提供を実現した階層符号化方式が検討された。MPEG-4 FGS [64] [65] [66] [67] と呼ばれる本方式は、高位階層の符号化処理において従来の 2 次元ランレングス方式に代わってビットプレーンにより量子化された DCT 係数を符号表現する。これにより、伝送時には回線の特性に応じて任意の場所でストリームの受信を打ち切ることで連続的な品質階層を提供することが可能となり、従来の階層符号化の課題として挙げられていた提供品質の制限に関する問題への解を与えるものである。しかし、本機能を利用するために専用の機構を符号化器と復号器に要求する点は変わらないため、符号化器/復号器の汎用性の問題は解決されない。

2.3 トランスコーディングと階層符号化の融合によるスケーラブル映像通信

本節では、符号化済の映像情報に対する変換処理によってスケーラビリティと同等の機能を実現する方法を検討する。汎用的な符号化器が生成する MPEG ビットストリームを入力とし、基本階層と高位階層の階層構造を有する符号化構造への変換を行うビットストリーム分離器と、逆に、階層符号化情報から非階層符号への変換を行い元のビットストリームを合成するビットストリーム合成器による階層符号化処理をトランスコードの応用により実現する方法について説明する。

2.3.1 ストリーム分離器を用いた非階層から階層符号化情報への変換

本論文で提案するビットストリームの分離・合成によるスケーラビリティの実現方法の概念を図 2-1 に示す。本機能はトランスコードの機能拡張として実現する。すなわち、入力された MPEG 等の標準符号化方式準拠の非階層の符号化情報 (Bitstream A) はトランスコード処理部でビットレートが削減され、入力時よりも低いビットレートに変換された MPEG-2 ビットストリーム (Bitstream B) が出力される。同時に、トランスコードの入力と出力間の差分成分が差分映像情報符号化器 (Differential Data Coder) へ送られ、変換差分ビットストリーム (Bitstream C) が符号化/出力される。このように、トランスコードの入力/出力間の差分情報を利用することで非階層のビットストリーム (Bitstream A) から基本階層 (Bitstream B) と高位階層 (Bitstream C) のビットストリームを生成でき、変換処理により、階層構造を有する符号化器を用いずに階層符号化機能を実現できる。

ビットストリーム分離器はトランスコードの機能を内部に持つため、トランスコード処理部から出力される Bitstream B は標準符号化方式に準拠しており汎用的な復号器で復号/再生可能である。また目標となるビットレートへの制御が可能のため、要求条件等の変化にも柔軟に対応できる。

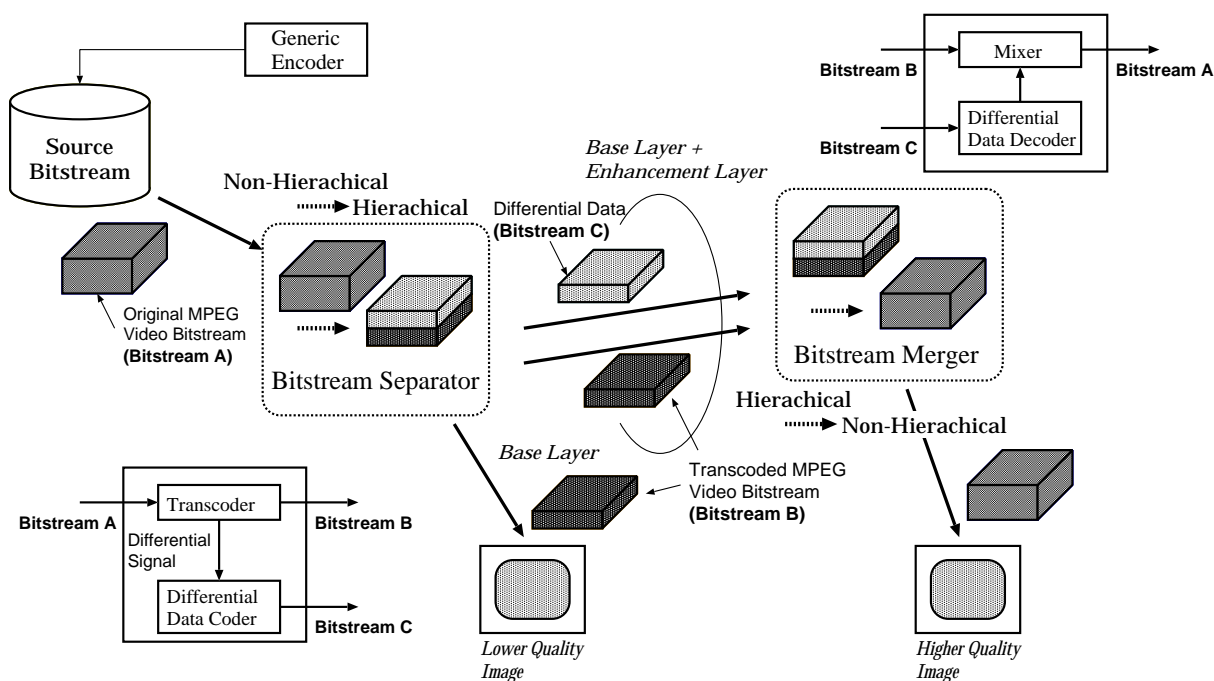


図 2-1: トランスコーダを用いた階層符号化への変換

2.3.2 ストリーム合成器による階層符号から非階層符号化への変換

Bitstream B より提供される基本階層に対して Bitstream C を印加して高品質な映像信号を生成する処理は、復号器の前段に別の処理器 (ビットストリーム合成器) を設けて、本処理器の出力を復号器へ接続することで高画質映像の復号・再生を行う。このとき、Bitstream C の符号化処理において可逆な符号化構造を持たせることで、合成処理後には分離前の Bitstream A と同一のビットストリームを出力する。スケーラビリティ用復号器では、復号器の機能として高品位映像の生成を行うために、階層構造を有する専用の復号器でないと高位階層の復号処理が行えないのに対して、復号器とは別に処理器が高画質層映像の生成を行うために、特別な構造を持たない汎用的な復号器により高画質映像の再生が可能となる。

2.4 従来方式との比較

階層符号化処理を用いた品質選択機能を標準符号化方式の枠組みの中で実現するために、MPEG-2 の標準化過程において様々な階層符号化の方法が議論された [60] [68] [69] [70]。本節では、同一空間/時間解像度のもとで符号化歪みにより階層化を行う階層符号化方式として SNR スケーラビリティ、ビットストリーム更新スケーラビリティ [18] [71] に関して、本章で提案するトランスコーディングの応用による方式と比較する。

2.4.1 SNR スケーラビリティとの比較

SNR スケーラビリティでは、符号化の段階で基本階層と高位階層のビットストリームを生成する。そして、これを実現するためには専用の符号化器/復号器が必要となる。また、SNR スケーラビリティ対応の符号化器/復号器は、基本階層と高位階層の双方で並列的に処理を行う機構を要求するために、符号化/復号処理量の増大、符号化/復号器の構造の複雑化等の問題が指摘されている。

本方式では階層化処理と符号化/復号処理を行う処理器が分離されるので、変換器と符号化/復号器との間でDCT係数と符号化データとの相互変換のためのエントロピ符号化/復号処理に要する処理量が加算される。しかし、DCTや動き補償などの処理を行わずに階層符号への変換を実現すれば階層化に伴う処理量の増加は抑制可能であり、各階層ごとに動き補償予測ループ構造が必要なアーキテクチャに対する処理量の優位性を確保可能である。また、高位階層映像の復号には復号処理の前段に専用の処理器を設ける必要があるが復号器自身への変更は不要であり、既に普及しているMP@ML準拠の復号器の資源の有効利用を図れる。

2.4.2 ビットストリーム更新スケーラビリティとの比較

階層符号化方式における符号化/復号構造の複雑化、処理量の増大などの問題を解決する手段として、復号器の前段で階層構造を有する符号化情報を非階層の符号化情報へ変換する処理器を設けて、階層構造を持たない汎用的な復号器で階層符号化情報を用いた品質選択機能を提供を行うビットストリーム更新スケーラビリティ(以下、更新スケーラビリティと呼ぶ)が提案されている [71]。本方式では、符号化器側では基本階層と高位階層の階層型符号化情報を生成するが、このときに、階層構造から非階層構造への変換が可能な専用の情報構造をもつビットストリームを生成する。復号側では、復号器の前段に専用の処理器(更新処理器)を設けて、更新処理器に両階層の符号化情報を入力して加工・合成を行い、高画質の標準符号化情報を生成し、これを後段の非階層の復号器で復号する。

ここで、更新処理器内での変換処理は提案方式におけるビットストリーム合成処理と全く同様である。一方、符号化側においては、更新スケーラビリティにおいても符号化器自体は階層構造を要求するため、符号化器の構造の複雑化、処理量の問題は依然として残る。更新スケーラビリティでは、量子化係数値の領域で基本階層と高位階層間の量子化ステップ値の違いを補償した階層間差分信号を生成することで、SNR スケーラビリティよりも高い画質を実現するが、基本階層と階層間差分符号量は非階層方式のそれよりも大きく、非階層と同レベルの符号化効率を達成するまでには至っていない。これに対して、提案方式では、符号化側で非階層から階層構造への変換を符号化器とは別の処理器(ストリーム分離器)で行うことで、復号側だけでなく符号化側においても汎用の符号化器で階層符号化機能を提供可能な点を特徴とする。

2.4.3 提案方式の特徴

上記に述べた方式は符号化方法に関する手段であり、符号化済のビットストリームに対してスケーラビリティを提供するものではない。また、符号化段階で階層化を行うために階層化処理機構が実装されている符号化器でしか実現できない問題点があり、既に広範囲で利用される MP@ML レベルのシステムへの適用は不可能となっている。これに対して、トランスコーディングを拡張した方式では、標準方式に準拠した符号化済のビットストリームを入力として、合成器で汎用的な復号器で復号可能なビットストリームを出力することで、符号化器自体に特別な処理構造を必要とせず、通常の標準符号化ビットストリームに対して、スケーラビリティを有する符号化構造を実現することが大きな特徴である。

符号化画像情報に変換処理を行い、その過程で発生する入力符号化情報に対する変化情報成分を保存し別途符号化し、これを合成することでスケーラブル映像蓄積・配信方式を可能にする。特に、トランスコーダとしての機能が基盤にあるために、接続環境に応じた柔軟なビットレート変換機能という特徴を継承する。したがって、符号化/復号器の構造の複雑化と処理量の増大、ビットレートに対する柔軟性、符号化効率の低下などの、従来の符号化/復号器での階層化処理によるスケーラビリティの実現方式における問題を解決できる。

2.5 階層型動画像伝送のアプリケーション

2.5.1 ニュース映像の速報配信のための階層型伝送方式

ニュース映像の編集作業は即時性/速報性が重要であり、素材映像を編集側にできるだけ早く伝送することが求められ伝送時間への要求が高い。一方、編集作業自体は高い品質でなくても作業に影響が出ない程度の品質で十分であり、使用する映像に対する品質要求は高くはない。しかし、編集作業後に放送されることを考慮すると最終的には高い品質が要求されることから、編集段階および放送段階のそれぞれにおける品質要求の違いが発生する。そこで、このようなアプリケーションに提案方式を適用することで、編集、放送のそれぞれの段階で要求される最適な品質の符号化映像を効率的に伝送する動画像情報の階層型伝送方式が実現可能となる。

素材映像の送信側ではストリーム分離処理を行い階層符号へ変換した後に、低レートの基本階層のみを先に伝送して、伝送時間を節約する。一方、編集側では、基本階層映像を用いて編集作業を行い、編集作業が完了後には高位階層符号の中から編集後の基本階層符号に対応する部分のみを再送する。このとき、編集でカットされる部分の高位階層情報は伝送不要であり、転送情報の総容量の削減が可能となる。さらに、配信段階で伝送帯域および要求伝送時間に応じたビットレートへ変換するため、符号化処理器自体には伝送帯域を考慮したビットレートを設定する必要性が無くなり、高レート/高品質の符号化動画像として素材映像の取得が可能となる。

2.5.2 映像監視システムにおける監視映像の階層伝送

遠隔監視システムなどでは、監視カメラからの監視映像を中央の管理局へ伝送する場合、異常事象のない平常の状況に置いては、監視側では大まかな映像内容が把握できれば十分であり、必ずしも高い画質が要求されるわけではない。特に、映像伝送に必要な帯域は常に確保されている必要があり、また監視映像を録画する場合には膨大な記録媒体が必要となるため、監視映像の低レート化が要求される。そこで、監視システムにストリーム分離・合成機能を適用することで、トランスコードによるレート削減機能と高位階層符号化情報を用いた高画質化を組み合わせた新たな監視映像の伝送システムが実現される。

監視カメラにストリーム分離機能を実装し、高画質/高ビットレートで符号化処理された監視映像をストリーム分離処理によって基本階層と高位階層へ変換して、基本階層のみを監理局へ伝送/蓄積し、高位階層を監視カメラ内に蓄積する。これにより平常時での監視映像の伝送に必要な帯域を節約可能となる。異常事象が発生した場合には、その周辺時間帯での高位階層符号を伝送し、監理局側で蓄積基本階層と高位階層を合成して詳細な映像を取得する。これにより、平常時には基本階層伝送により伝送帯域を節約しながら、異常時には高位階層の伝送とストリーム合成処理により異常部分の詳細を取得可能となる。

監視映像の伝送路としてインターネットを利用することで安価なシステムが実現できる [72] が、IP ネットワーク上での映像伝送では、中間ノード/ルータなどで IP パケットを容易に解析でき伝送動画像の盗聴の危険性を有する。そのため、当事者以外からの閲覧を遮断するための映像内容の保護機能が要求される [73]。ところで、高位階層符号は基本階層に対する付加信号であるから高位階層符号単体での復号・再生は不可能である。本特徴を利用して、符号化情報の中から基本階層として再生に重要な部分のみを保護し、残りを保護対象から除外することで、映像内容の保護に要する処理の低減が可能となる。すなわち、情報量を削減した基本階層のみを暗号化することで、暗号化処理に必要な演算量の削減が可能となる。

2.6 まとめ

本章では、階層符号化方式によるスケーラビリティの実現方式について述べ、符号化/復号構造の複雑化、階層化劣化による符号化効率の低下、符号化器/復号器の汎用性の課題があることを示した。また、トランスコードによりレート削減された符号化情報は階層符号化方式における基本階層と同様の目的で利用される点に着目し、レート削減処理における処理前後間での差分映像信号を別途符号化することで、階層符号化処理と同様の機能がトランスコードの拡張により実現可能なことを示した。そして、非階層から階層構造への変換を行うストリーム分離器および階層構造から非階層構造へ変換するストリーム合成器を用いた階層型動画像伝送方式を提案し、符号化/復号処理器内で階層化を行う従来の方式と比較した。

以降の各章において、階層符号への変換処理のための具体的な符号化信号の処理方式を述べる。まず、第3章では DCT 係数領域内での再量子化処理によるトランスコードに差分信号の符号化処理機能を付加することで、基本階層と高位階層の2段階の品質を提供する階層符号化方式を提案する。次にこれを拡張した方式として、第4章にて高位階層の伝送段階で

高位階層符号化情報内の部分情報のみを抽出して合成することで，基本階層と中間的な品質を提供する方式を提案する．さらに，第5章では3階層以上の階層を有する多階層型の階層符号化方式において，最下層（基本階層）以外の階層においては階層間の依存関係を排除した互いに独立な情報構造により符号化情報を構成することで，階層符号化映像の情報構造自体は2階層の依存構造でありながら， $N(N > 2)$ 階層の品質を提供可能な新たな階層符号化方式を提案する．

第3章

ビットストリーム分離・合成によるスケーラブル映像符号化方式

3.1 はじめに

本章ではトランスコードの変換処理前後間の差分情報が階層符号化情報の生成に利用可能な点に着目し、第2章で提案した符号化器/復号器の外部機能によスケラビリティを実現する新たな符号化方式を提案する。特に、符号化効率の向上に焦点を当てて差分情報の効率的な符号化アルゴリズムを提案する。まず、非階層から階層符号化情報への変換を行うビットストリーム分離器・合成器のアーキテクチャについて述べ、本機能はDCT係数領域内の再量子化によるビットレート変換トランスコードを拡張して実現されることを示す。次に、トランスコード内の再量子化処理により発生する量子化領域でのDCT係数差分に着目し、量子化係数値の変化について考察し、これに基づく係数差分の符号化方法と、再量子化前の元の係数情報の復元方法を提案する。最後に、発生符号量の観点から提案方式を評価するとともに、符号化効率に関する数値解析実験を行い、本提案方式の有効性を示す。

3.2 ビットストリーム分離・合成処理器の構成

本節では、符号化済の映像情報に対する変換処理によってスケラビリティと同等の機能を実現する方法として、階層構造を持たない汎用的な符号化器より生成されるMPEG-2ビットストリームを入力とし基本階層と高位階層への分離を行うビットストリーム分離器と、分離されたものから元の分離前のビットストリームを合成するビットストリーム合成器を導入し、ビットストリームの分離・合成によりスケラビリティを実現する方法について述べる。

3.2.1 ビットストリーム分離器

ビットストリーム分離器の構成を図3-1に示す。トランスコード処理部において、Bitstream A を可変長復号 (VLD)、ブロック復元処理 (Re-construct Block Data) を経て量子化係数ブロック B_1 を復元する。次に、 B_1 を逆量子化 (IQ) してDCT係数領域まで復元した後に符号化時よりも粗いステップサイズで再量子化処理 (Q) を行い再量子化出力係数ブロック B_2 を生成する。さらに、 B_2 をジグザグスキャンして二次元ランレングス系列に変換 (Convert

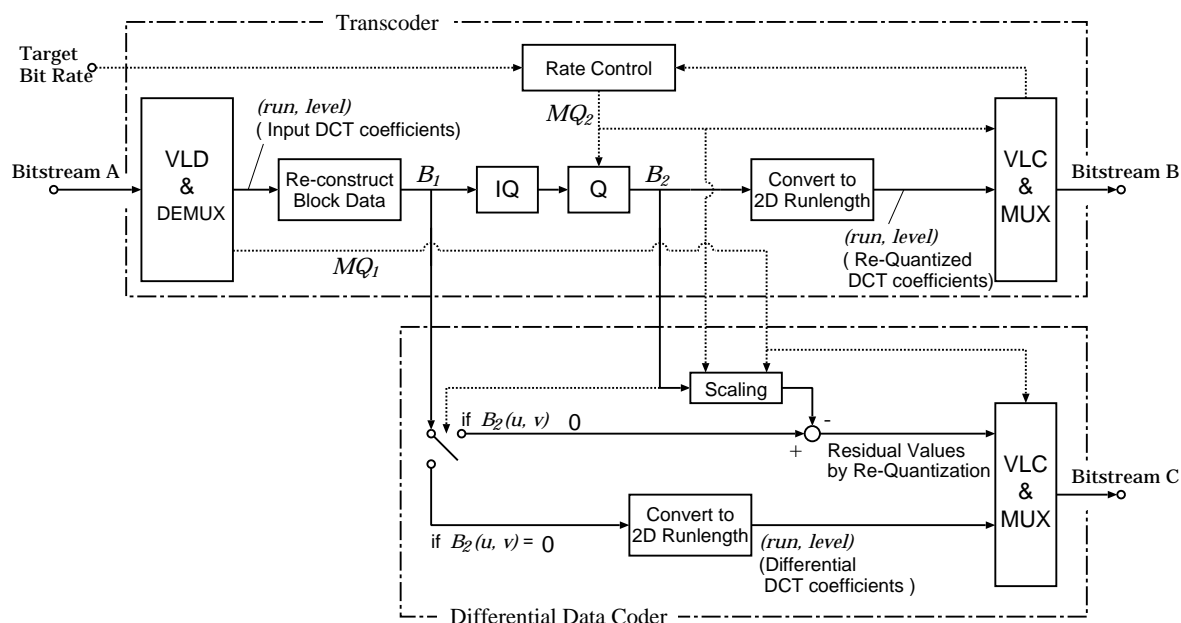


図 3-1: ストリーム分離器ブロック図

to 2D Runlength) した後に可変長符号化処理 (VLC) を経て Bitstream B を出力する。

一方、差分映像情報符号化器では、トランスコード処理部より係数ブロック B_1 、 B_2 及び符号化時/再量子化時それぞれの量子化パラメータ MQ_1 、 MQ_2 を入力して B_1 、 B_2 への係数ブロックの変化成分を符号化する。本変化成分情報は、 B_2 内の係数値 $B_2(u, v)$ が 0 か否かにより分類して符号化される。ここで (u, v) は係数ブロックにおける座標を表し $0 \leq u \leq 7$ 、 $0 \leq v \leq 7$ である。 $B_2(u, v) = 0$ の場合は B_1 内の連続する零係数と $B_1(u, v)$ の値をペアとした二次元ランレングスにより符号化する。一方、 $B_2(u, v) \neq 0$ の場合には、 $B_2(u, v)$ に MQ_1 と MQ_2 の比率 MQ_2/MQ_1 を乗じて再量子化前の値に近づけた後に、 $B_1(u, v)$ との差分を算出することで、再量子化演算により丸められた残差成分を算出して、これを符号化出力する。上記の一連のアルゴリズムの詳細については 3.4 で詳述する。

3.2.2 ビットストリーム合成器

ビットストリーム合成器の構成を図 3-2 に示す。合成器は Bitstream B と Bitstream C を入力として、両者を合成し Bitstream A を出力する。Bitstream B については可変長復号、ブロック復元処理により B_2 を復元する。一方、Bitstream C の符号化情報は、再量子化値 $B_2(u, v)$ が 0 か否かで分類した形で再量子化演算に伴う係数の変化情報が符号化されている。

$B_2(u, v) = 0$ の係数の変化情報については対応する $B_1(u, v)$ の値が二次元ランレングスにより符号化されているので、上記と同様にブロックを再構築する。

$B_2(u, v) \neq 0$ となる係数の変化値については、 B_2 内の 0 でない係数 (有意係数) の位置を検出し、それを元に復号値に対応させる B_2 内の係数を特定する。その後、 $B_2(u, v)$ に MQ_2/MQ_1 を乗じた後に上記復号値を加算して再量子化前の係数値 $B_1(u, v)$ を得る。この

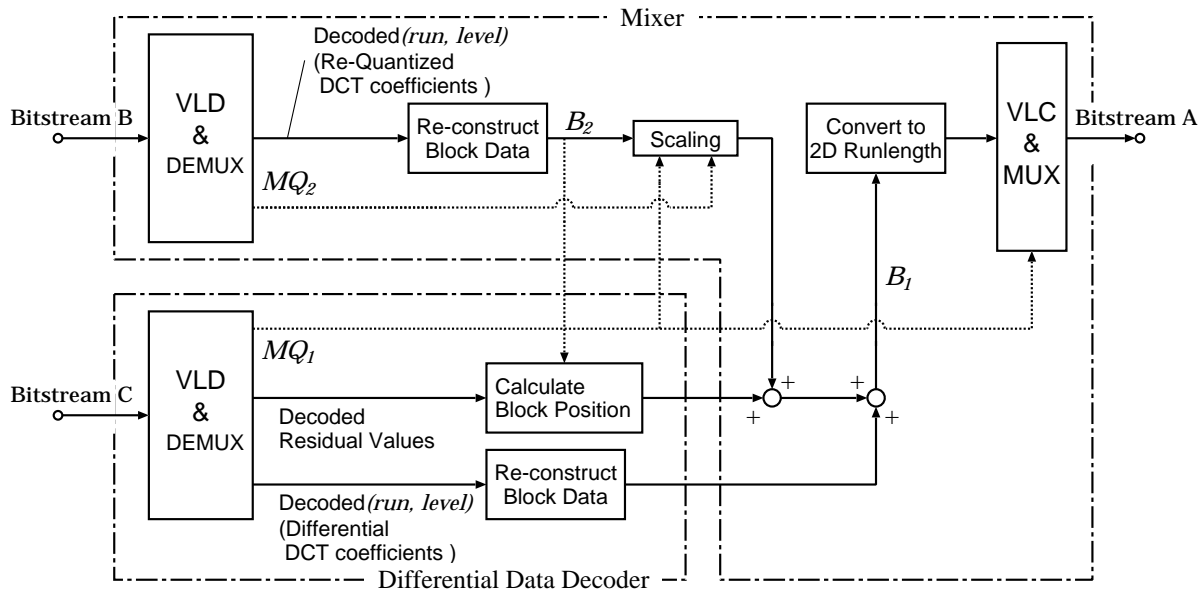


図 3-2: ストリーム合成器ブロック図

とき、 B_2 内の有意係数から座標情報が特定できるので Bitstream C 自身にランレングスのようなブロック内の座標を特定するための情報は不要となる。

以上より、再量子化前の量子化係数が、それぞれ別々のブロックデータとして復元されたので、両ブロック同士を加算して再量子化前の係数ブロック B_1 を復元する。その後、 B_1 をジグザグスキャンして二次元ランレングス系列に変換した後に可変長符号化処理を経て Bitstream A を出力する。

3.3 再量子化演算における係数値の変化特性の解析

ビットストリーム分離器から出力される変換差分ビットストリーム (Bitstream C) は再量子化演算による係数ブロックの変化情報を符号化したものである。本節では、再量子化演算による係数値の変化について考察する。

3.3.1 再量子化演算処理および量子化制御方法

ストリーム分離・合成処理はマクロブロック (以下、MB と呼ぶ) 単位に行われ、トランスコード処理後の符号量は再量子化時のステップサイズ (再量子化パラメータ) によって制御される。再量子化前の MB に対してステップサイズを粗くとるほど、多くの係数情報が削減されそれに伴い再量子化前後間での係数ブロックの変化が大きくなるので、係数変化情報は再量子化前後の量子化パラメータに依存する。

逆量子化、再量子化処理は、逆量子化演算と再量子化演算を結合させた演算を用いて、再量子化入力量子化係数値 $k_1 (= B_1(u, v))$ から直接再量子化出力係数値 $k_2 (= B_2(u, v))$ を算出する [74]。本処理は画面内符号化モード (イントラフレーム符号化モード、以下イントラモードと呼ぶ) と画面間符号化モード (インターフレーム符号化モード、以下インターモー

ドと呼ぶ) で別々の演算式により行い、イントラモード時の演算式を式 (3.1) に、インターモード時の演算式を式 (3.2) にそれぞれ示す.

$$k_2 = \left\lfloor k_1 \times \frac{MQ_1}{MQ_2} + \frac{1}{2} \times \text{sign}(k_1) \right\rfloor \quad (3.1)$$

$$k_2 = \left\lfloor \left(k_1 + \frac{1}{2} \times \text{sign}(k_1) \right) \times \frac{MQ_1}{MQ_2} \right\rfloor \quad (3.2)$$

式 (3.1), 式 (3.2) において $\text{sign}(x)$ は x の符号を表し式 (3.3) で与えられる.

$$\text{sign}(x) = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0 \end{cases} \quad (3.3)$$

ここで, $\lfloor x \rfloor$ は x を越えない最大の整数を表す. すなわち, x が任意の係数値 k に対して $k \leq x < k+1$ を満たすときに, $\lfloor x \rfloor = k$ のように算出される.

再量子化パラメータ MQ_2 の算出に関して, 再量子化処理特有のレート歪み特性 [75] を考慮して, 符号化効率が劣化する領域を避けるように再量子化パラメータを選択する量子化パラメータ禁止領域制御 [76] を適用する. 本制御は, MQ_2 を, 再量子化前の量子化パラメータ MQ_1 と整数値 m により算出し, MQ_2 の値域を特定の値に制限する制御を行う. これもまたイントラモード, インターモードで分けて行われ, イントラモードの場合は式 (3.4) に, インターモードの場合は式 (3.5) のようにして MQ_2 を算出する.

$$MQ_2 = \begin{cases} MQ_1 & \text{if } (m = 0) \\ 2m \times MQ_1 + 1 & \text{if } (m \neq 0) \end{cases} \quad (3.4)$$

$$MQ_2 = (m + 1) \times MQ_1 \quad (3.5)$$

そこで, 以下では式 (3.1), 式 (3.4) もしくは式 (3.2), 式 (3.5) を用いて再量子化演算が行われるときの, i) 整数化処理に伴う切り上げ演算による影響を 3.3.2 で, ii) 再量子化出力値が 0 となるような再量子化入力値の範囲 (デットゾーン) について 3.3.3 で考察する. ただし, 本考察では正の係数値についてのみ議論する.

3.3.2 整数化処理に伴う切り上げ演算による影響

式 (3.1) および式 (3.2) の演算式において, MQ_2 が MQ_1 に近い場合には, 丸め演算の結果 $k_2 = k_1$ となって再量子化を行っても係数値が変化せず符号量の削減がなされない場合がある. このとき量子化パラメータは MQ_1 から MQ_2 に変化しているため, 量子化誤差は増加する [75]. 本節では式 (3.4), 式 (3.5) のようにして MQ_2 を決めることによってこのような現象を回避し, 必ず係数の変化を生じさせる制御が実現されることを示す.

1. イントラモードの場合

式 (3.1), 式 (3.4) より再量子化出力係数値 k_2 は m を用いて式 (3.6) のように表される.

$$k_2 = \begin{cases} k_1 & \text{if } (m = 0) \\ \left\lfloor \frac{k_1}{2m + MQ_1^{-1}} + \frac{1}{2} \right\rfloor & \text{if } (m \geq 1) \end{cases} \quad (3.6)$$

式 (3.6) の演算において $m = 1$ のときには式 (3.7) に示す不等式が成立する .

$$k_2 \leq \frac{k_1}{2 + MQ_1^{-1}} + \frac{1}{2} < k_2 + 1 \quad (3.7)$$

MQ_1 の値域は $1 \leq MQ_1 \leq 31$ なので $MQ_1^{-1} \geq 0$ である . そのため式 (3.8) に示す不等式が成立する .

$$\frac{k_1}{2 + MQ_1^{-1}} + \frac{1}{2} < \frac{k_1}{2} + \frac{1}{2} \quad (3.8)$$

式 (3.7), 式 (3.8) より式 (3.9) を得る .

$$k_2 \leq \frac{k_1}{2 + MQ_1^{-1}} + \frac{1}{2} < \frac{k_1}{2} + \frac{1}{2} \quad (3.9)$$

ここで, $k_1 \geq 1$ の範囲において $k_1/2 + 1/2 \leq k_1$ であるため k_1 と k_2 の間には式 (3.10) が成立する .

$$k_2 \leq \frac{k_1}{2 + MQ_1^{-1}} + \frac{1}{2} < k_1 \quad (3.10)$$

従って, 式 (3.10) より $m = 1$ のときには, 入力量子化係数は必ず再量子化前よりも小さな値へ変換される .

一方, $m > 0$ より式 (3.11) が成立する .

$$\left\lfloor \frac{k_1}{2m + MQ_1^{-1}} + \frac{1}{2} \right\rfloor \geq \left\lfloor \frac{k_1}{2(m+1) + MQ_1^{-1}} + \frac{1}{2} \right\rfloor \quad (3.11)$$

以上から, 式 (3.10), 式 (3.11) より $m > 1$ に対しても $k_2 < k_1$ の関係は保たれるので, 整数化演算を考慮しても, 零でないあらゆる係数は再量子化前よりも小さな値へ変換される .

2. インターモードの場合

式 (3.2), 式 (3.5) よりインターモード符号化時の再量子化出力係数値は式 (3.12) のように表せる .

$$k_2 = \left\lfloor \frac{k_1 + 1/2}{m + 1} \right\rfloor \quad (3.12)$$

式 (3.12) において, $m = 1$ の時には式 (3.13) に示す不等式が成立する

$$k_2 < \frac{k_1 + 1/2}{2} < k_2 + 1 \quad (3.13)$$

$k \geq 1$ の領域においては $(k_1 + 1/2)/2 < k_1$ であるため, これと式 (3.13) より, $m = 1$ のときには $k_2 < k_1$ が成立する .

また, $m > 0$ から式 (3.14) が成立する .

$$\left\lfloor \frac{k_1 + 1/2}{m + 1} \right\rfloor \geq \left\lfloor \frac{k_1 + 1/2}{(m+1) + 1} \right\rfloor \quad (3.14)$$

式 (3.13), 式 (3.14) より $m > 1$ に対しても $k_2 < k_1$ の関係は保たれ, インターモードの再量子化演算においてもイントラモード時と同様の関係が成り立つ .

3.3.3 再量子化入力値のデットゾーン特性

再量子化処理により有意係数 (非零係数) が非有意係数 (零係数) に変換されてその個数が減少すると符号量は大きく削減される。そこで、本節では $k_2 = 0$ を満たす入力量子化係数値 k_1 の条件を導出する。

1. イントラモードの場合

式 (3-6) において $k_2 = 0$ のときには式 (3-15) が成立する。

$$0 \leq \frac{k_1}{2m + MQ_1^{-1}} + \frac{1}{2} < 1 \quad (3-15)$$

式 (3-15) を整理して、式 (3-15) が成立するための m と k_1 の関係式 (3-16) を導く。

$$k_1 < m + \frac{1}{2}MQ_1^{-1} \quad (3-16)$$

ここで、 k と m はともに整数値であることを考慮すると、式 (3-16) から $k_1 \leq m$ が導かれる。以上から、 $k_2 = 0$ になるための k_1 の条件は $k_1 \leq m$ となる。

2. インターモード場合

式 (3-12) において $k_2 = 0$ のときには式 (3-17) に示す不等式が成立する。

$$0 < \frac{k_1 + 1/2}{m + 1} < 1 \quad (3-17)$$

次に、式 (3-17) が成立するための m と k_1 の関係として式 (3-18) が導かれる

$$k_1 < m + \frac{1}{2} \quad (3-18)$$

ここで、 k_1 , m は整数値であるので、 $k_2 = 0$ となるための k_1 の条件は、イントラモードと同様に $k_1 \leq m$ である。

3.4 再量子化特性を利用した差分映像情報符号化方式

3.3 より再量子化演算における演算前後での量子化係数値の変化について考察した。本節では、3.3 より導かれる特性を利用した再量子化前後間差分量子化係数の符号化方法について述べる。

3.3.2 および 3.3.3 の考察より本再量子化演算処理において式 (3-19)、式 (3-20) に示す性質が導かれる。

$$\text{I: } k_2 = 0 \quad (0 \leq k_1 \leq m) \quad (3-19)$$

$$\text{II: } k_2 < k_1 \quad (k_1 > m) \quad (3-20)$$

式 (3-19)、式 (3-20) を用いて再量子化前の係数ブロック B_1 の復元に必要な変化成分情報について考察する。まず、性質 I より m 以下の係数値は再量子化後には零係数としてまと

められるため、 B_2 内の零係数は元からの零係数と再量子化後により変換されたものとが混在する。そのため、係数値とともに B_2 内のどのゼロ係数に対応するかを特定するための位置情報が必要となる。そこで、式 (3-19) を満たす係数については B_1 内の連続する 0 の個数と再量子化前係数値 $k_1 (= B_1(u, v))$ をペアとした二次元ランレングスにより符号化する。

一方、性質 II は式 (3-6)、式 (3-12) の整数化演算において、再量子化後の係数値は必ず再量子化前よりも値が減少することを保証する。すなわち、 B_2 内の有意係数は全て再量子化演算により値が変化したものであり、これらの係数と同一位置に必ず変化成分が存在する。そのため、 B_2 内の有意係数を決められた順番 (ジグザグスキャン順) に従って参照すれば対応する係数が分かるため、ブロック内位置の特定のための情報が不要となる。さらに、 k_1 の算出に必要な情報としては、 k_1 の値そのものが再量子化前後間の係数差分値 $k_1 - k_2$ を符号化する方法が最も簡単である。復号時には、復号済の k_2 にこの差分値を加算することで k_1 を算出できる。しかしながら、量子化係数の値域は ± 2047 まで定義されており、十分に大きな値を有する係数値に対しては差分値も大きくなる。一方、式 (3-1)、式 (3-2) に示す演算は、 k_1 に比率 MQ_1/MQ_2 を乗じた後に小数点以下を丸めて整数化して k_2 を算出している。それならば、この丸められた成分のみがあれば k_1 を得るには十分である。以上より、式 (3-20) を満たす係数については、係数値に関する情報のみを符号化対象とし、 MQ_1 、 MQ_2 を用いて再量子化で丸められた剰余成分を算出して、これを符号化出力する。

以上の考察から、再量子化演算に伴う係数ブロックの変化情報の符号化アルゴリズムの方針を以下のように定める。

方針 1 $B_2(u, v)$ が 0 か否かで B_1 内の有意係数を分類

方針 2 $B_2(u, v) = 0$ の場合は B_1 内の連続する 0 の個数と係数値で二次元ランレングス符号化

方針 3 $B_2(u, v) \neq 0$ の場合は $B_2(u, v)$ 算出時に丸められた剰余成分を符号化

3.4.1 量子化係数変化情報の符号化方法

再量子化処理に伴う B_1 から B_2 への係数ブロックの変化の符号化処理は、 B_1 と B_2 をジグザグスキャンして、まず方針 2 に該当する係数を符号化した後に、残りの方針 3 に該当する係数を符号化する。

(1) $B_2(u, v) = 0$ に対する変化情報の符号化

1. B_1 、 B_2 をジグザグスキャンの順番で読みだす。 $B_1(u, v)$ 、 $B_2(u, v)$ により 2-1、2-2、2-3 の 3 通りの処理に分かれる。

2-1. $B_1(u, v) = 0$ の場合

零係数カウンタ c を 1 だけ増加し、3. へ進む。

2-2. $B_1(u, v) \neq 0$ かつ $B_2(u, v) \neq 0$ の場合

3. へ進む。

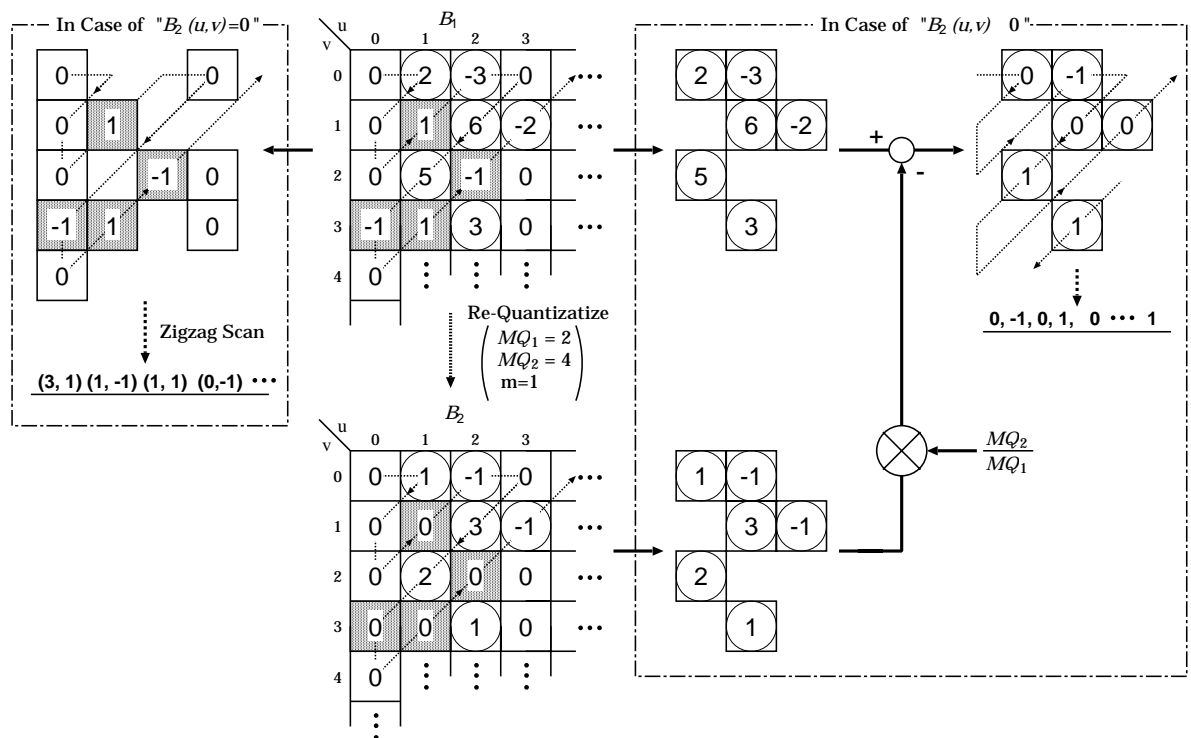


図 3-3: 係数ブロックの変化成分情報の符号化方法

2-3. $B_1(u, v) \neq 0$ かつ $B_2(u, v) = 0$ の場合

カウンタ c の値を run として, $B_1(u, v)$ を $level$ とした二次元ランレングスイベント $(run, level)$ を生成し, 可変長符号化. カウンタ c を 0 にリセット. run は B_1 内の連続する 0 の個数を表す.

3. ブロック末尾ならばランレングスの終りを示す符号 End_of_Runlength を符号化し終了. そうでなければ 1. へ戻る.

図 3-3 から具体例を述べる.

B_1, B_2 をスキャンして $B_1(1, 0) = 2$ を検出したら, $B_2(1, 0)$ を参照する. このとき $B_2(1, 0) = 1$ なので上記手順の 2-2. に該当し, ここはスキップされる. 次に, $B_1(1, 1)$ まで進んだときに $B_2(1, 1) = 0$ となる. そこで, ここまでにカウントされた 0 の個数は 3 個となるから $run = 3$, $level = B_1(1, 1) = 1$ とした $(3, 1)$ を可変長符号化する.

次に, $B_1(2, 0) = -3$ を検出するが, このとき $B_2(2, 0) = -1$ なのでここをスキップし, 同様に $B_1(2, 1), B_1(1, 2)$ もまたスキップされる. 次の $B_1(0, 3) = -1$ にて $B_2(0, 3) = 0$ となり, $B_1(2, 0)$ からここまでにカウントされた 0 の個数は $B_1(3, 0) = 0$ の 1 個より $run = 1$ として, $level = B_1(3, 0) = 1$ とした $(1, -1)$ を可変長符号化する. 以下, 同様に B_1 をスキャンして $B_1(u, v) \neq 0$ を検出したら $B_2(u, v)$ を参照し, $B_2(u, v) = 0$ のときに二次元ランレベルイベントを符号化する.

(2) $B_2(u, v) \neq 0$ に対する変化情報の符号化

次に，方針 3 に該当する係数の符号方法を述べる．以下にその手順を示す．

1. B_1, B_2 をジグザグスキャンの順番で読みだす．

2. $B_1(u, v) \neq 0$ かつ $B_2(u, v) \neq 0$ ならば， $B_2(u, v)$ に式 (3-1)，式 (3-2) における量子化パラメータ比率の逆数 MQ_2/MQ_1 を乗じた値と $B_1(u, v)$ との差分をとり，再量子化演算で丸められた成分 $B_{err}(u, v)$ 算出する．すなわち，式 (3-21) により $B_{err}(u, v)$ を算出して，これを符号化する．

$$B_{err}(u, v) = B_1(u, v) - \left\lfloor B_2(u, v) \times \frac{MQ_2}{MQ_1} + \frac{1}{2} \right\rfloor \quad (3-21)$$

3. ブロック末尾ならば終了．そうでなければ 1. へ戻る．

以上の手順により， $B_2(u, v) \neq 0$ に対応する係数の再量子化前後間での値の変化の情報はジグザグスキャンの順番で符号化出力される．

図 3-3 の例では，ジグザグスキャン時に $B_1(1, 0) = 2$ を検出したら，このとき $B_2(1, 0) = 1$ なので式 (3-21) より $B_{err}(1, 0) = 0$ を算出し，これを符号化する．以下，同様にして， $B_1(u, v) \neq 0, B_2(u, v) \neq 0$ を検出したら式 (3-21) より $B_{err}(u, v)$ を算出し符号化する．

3.4.2 復号方法

復号処理では再量子化後の係数情報と変化成分情報を組み合わせて，再量子化前の係数ブロック B_1 を再構成する．本処理は B_2 の復号処理直後に行われる．

まず，Bitstream C よりランレングス符号を復号し， B_1 において run の値からカウンタアップし特定されるジグザグスキャン位置に $level$ の値を挿入して $B_2(u, v) = 0$ となる係数の再量子化前の値が復元される．ここで， run の値に $B_2(u, v) \neq 0$ の係数はカウントされていないので，復号済の B_2 を参照して，これらの係数位置をスキップする．

End_of_Runlength 符号を検出して全てのランレングス情報を復号し終わったら， $B_{err}(u, v)$ 符号を復号する． $B_{err}(u, v)$ は，ジグザグスキャンの順番に従い符号化されているので， B_2 中の非零係数をジグザグスキャン順に参照して対応する係数を特定した後に， $B_{err}(u, v)$ と $B_2(u, v)$ から式 (3-21) を用いて $B(u, v)$ を逆算して求めて， B_1 内の本非零係数と同一位置に挿入する．

ここで， $B_{err}(u, v)$ は B_2 内の非零係数にジグザグスキャン順に対応しているので，ブロック内位置を特定する情報がなくても再量子化前の値を復元できる．

3.5 符号化性能評価

本論文で提案するビットストリーム分離・合成器に関して，変換差分ビットストリームの提案符号化アルゴリズムについて評価する．

表 3-1: 比較対象方式

方式	スキャン方式	可変長符号化テーブル
1	MPEG-2 方式	MPEG-2 VLC
2	提案方式	MPEG-2 VLC
3	提案方式	専用 VLC テーブル

3.5.1 変換差分ビットストリームの総符号量に関する評価実験

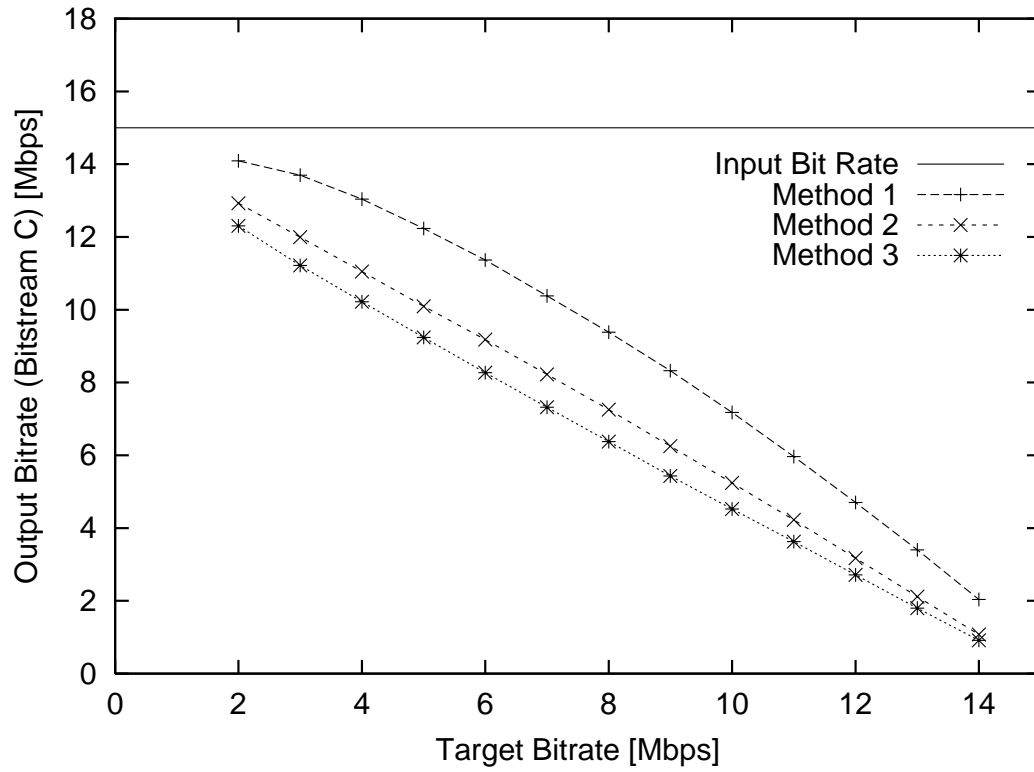
ストリーム分離器への入力 MPEG-2 ビットストリーム (Bitstream A), 出力 MPEG-2 ビットストリーム (Bitstream B), 変換差分ビットストリーム (Bitstream C) のビットレートをそれぞれ $R_A, R_B, R_C [Mbps]$ とする. Bitstream B の目標ビットレートを $2 [Mbps]$ から $(R_A - 1) [Mbps]$ まで $1 [Mbps]$ きざみでとり, このときの R_C および分離出力後の合計ビットレート $R_B + R_C$ について評価する. Bitstream A の符号化条件として表 3-2 に示す評価用 MPEG-2 ビットストリームを使用する. Bitstream C の符号化方法に関して比較対象方式を表 3-1 に示す.

方式 1 は B_1 と B_2 の係数ブロック同士で差分をとった量子化係数差分ブロック $\Delta B (= B_1 - B_2)$ を算出し, ΔB を MPEG-2 と同様の符号化アルゴリズムにより符号化した方式である. 方式 2 は, 3.4 で述べた符号化方式を適用し, 二次元ランレングス ($run, level$) は MPEG-2 標準の符号化表を用いて可変長符号化し, $B_{err}(u, v)$ はその値域に基づき固定長符号化する. 方式 3 は, 3.4 に示した方式からさらに符号化テーブルを再設計したものである. すなわち, ($run, level$), $B_{err}(u, v)$ にその確率分布をもとにして設計された可変長符号化表を適用する.

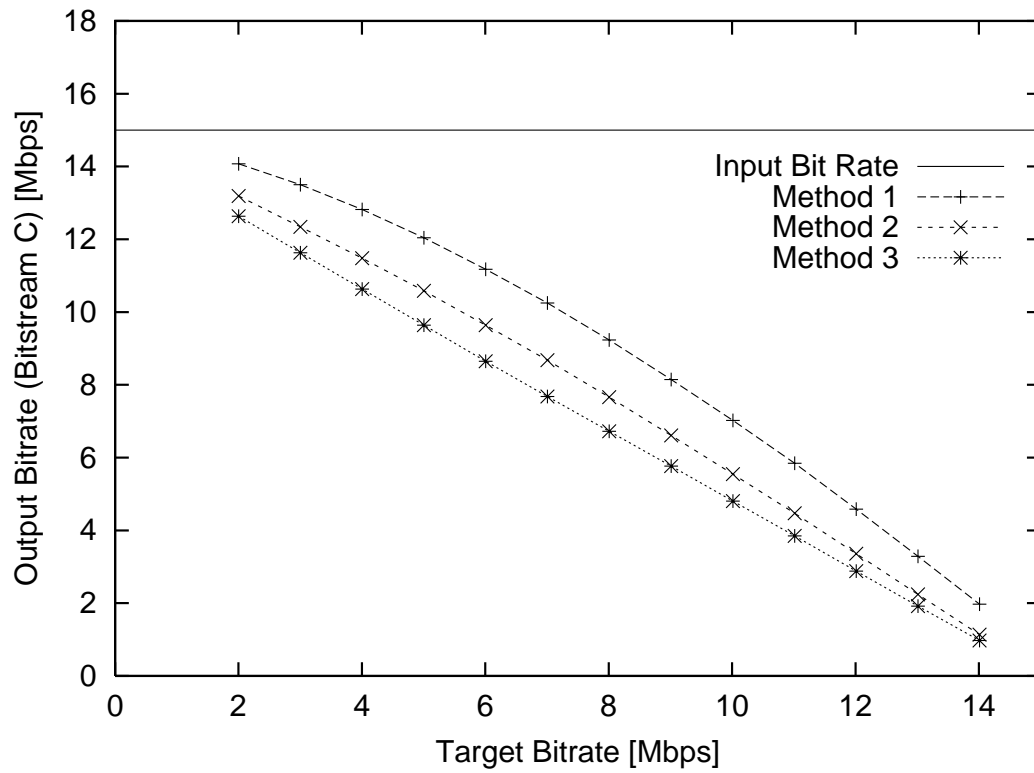
MPEG-2 では二次元ランレングスの係数レベル値 $level$ の範囲は ± 1 から ± 2047 までが定義されている. 一方, 3.4.1 で得られるランレングスイベントに関しては式 (3-19) より $level$ の値域として ± 1 から $\pm m$ までを定義すれば十分なので, MPEG-2 で規定される範囲は必要ないことが分かる. また式 (3-21) で算出される $B_{err}(u, v)$ も一様な確率分布である保証はなく, 分布に応じた符号化表の設計により符号化効率の向上が期待できる. そこで, 表 3-2 に示す条件で符号化された 7 種類のテストシーケンス (Bus, Table Tennis, Football, Flower Garden, Cheer Leader, Bicycle, Mobile & Calendar) を用いて, 目標ビットレートを本実験と同様に $2 [Mbps]$ から $(R_A - 1) [Mbps]$ までの範囲にとりストリーム分離処理を行い, そのときの ($run, level$) および $B_{err}(u, v)$ の確率分布の測定を行い, 本測定結果に対してハフマン符号化アルゴリズムを適用して符号語を導出することで, 可変長符号化テーブルの設計を行い, これを方式 3 に適用する.

変換差分ビットストリーム出力符号量特性

$R_A = 15 [Mbps]$ における Bitstream B の目標レートに対する R_C の変化を図 3-4 に示す. Bitstream C はトランスコード処理過程での変化情報なので, レート削減量が少ない (目標



(a) Bus



(a) Table Tennis

図 3-4: Bitstream C のビットレート R_C の変化 (入力符号化レート 15[Mbps])

表 3-2: 実験用ビットストリーム符号化条件

符号化方式	MPEG-2 Video MP@ML
符号化レート	15, 10 [Mbps]
符号量制御方式	TM5 方式
GOP 構造	N=15, M=3 (固定)
ピクチャ構造	フレーム構造
動き補償方式	フレーム/フィールド MC
テストシーケンス	Bus, Table Tennis
画像フォーマット	ITU-R BT.601 4:2:0 Format 輝度信号 704pel × 480 line 色差信号 352pel × 240 line
符号化フレーム数	150 [Frame]

レートに近い) ときにはビットストリーム構成情報は大きく変化しないために R_C は小さく、レート削減量が大きいく程、再量子化ステップサイズが粗くとられて DCT 係数の変化が大きくなることから、 R_C は目標レートに対して減少傾向を示す。

再量子化対象の MB や再量子化ステップサイズはトランスコード処理に依存するため各方式で同一である。しかし、方式間で R_C は大きく異なり、方式 1 のように単純に再量子化前後間で係数ブロック同士の差分をとって MPEG-2 と同様の方式によりランレングス符号化すると、他の方式に比べて非常に多くの符号量を要することが分かる。

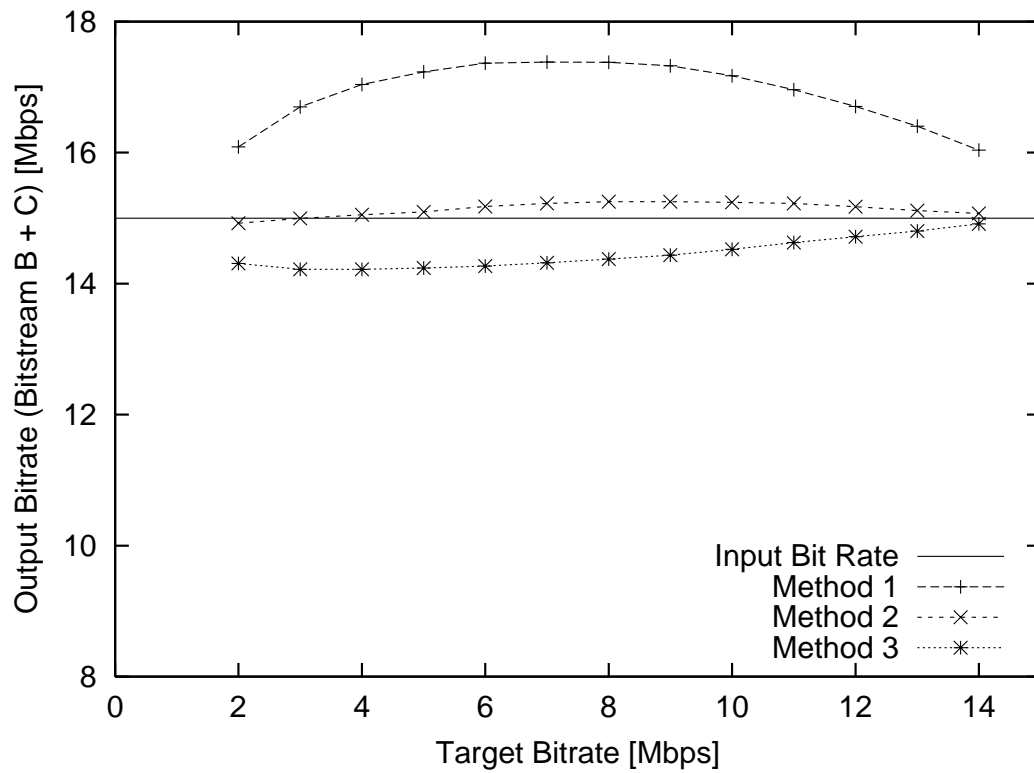
出力合計符号量特性

$R_A=15, 10$ [Mbps] における Bitstream B の目標レートに対する出力ビットレート合計値 $R_B + R_C$ の変化を図 3-5, 図 3-6 それぞれ示す。 R_A と $R_B + R_C$ を比較し、分離処理前後間での総符号量の変化について考察する。

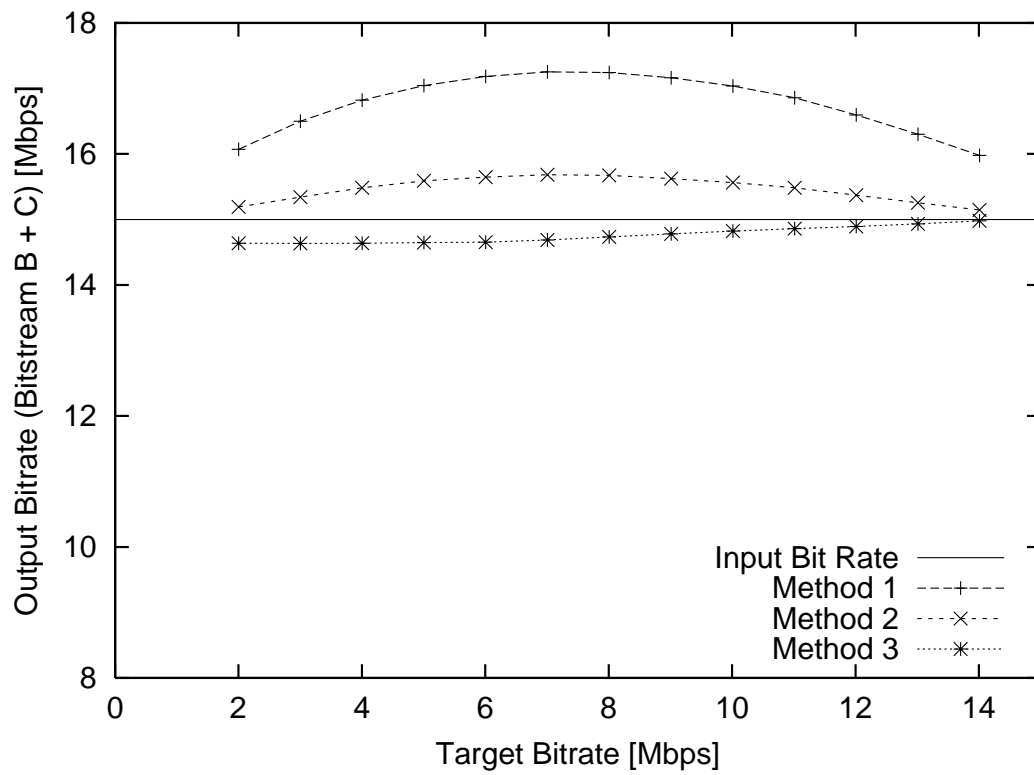
方式 1 は、分離後のビットレート合計値が入力を大きく上回っており符号化効率の大幅な低下が見られる。特に、中間の出力レートにおいてピークとなり、分離による符号量の増加量が入力符号量の約 10% に達する。 ΔB 内の有意係数は $B_2(u, v)$ の有意/非有意に関係なくすべての入力非零係数に対して発生する。したがって、 ΔB 中のランレングスイベント数は B_1 と変わらない。ここで、 $B_2(u, v) \neq 0$ の係数は復号時に B_2 を参照すれば特定可能なので、これらの係数に対するラン情報は不要である。方式 1 ではこれに関係なく ΔB をスキャンしているため、余計な符号が発生して符号化効率の低下をもたらしていると考えられる。

方式 2 は、3.4.1 で示した符号化方式により、このような冗長な符号を排除し、入力時とほぼ等しい出力符号量を達成可能となる。方式 1 にあった符号量の増大を抑制可能なことから、提案アルゴリズムが符号化効率の点で有効であることが分かる。

方式 3 では、可変長符号化テーブルを係数変化情報の符号化用に再設計することで入力よりも少ない符号量を達成できる。MPEG-2 の符号化表では *level* の値として $\pm 1 \sim \pm 2047$ ま

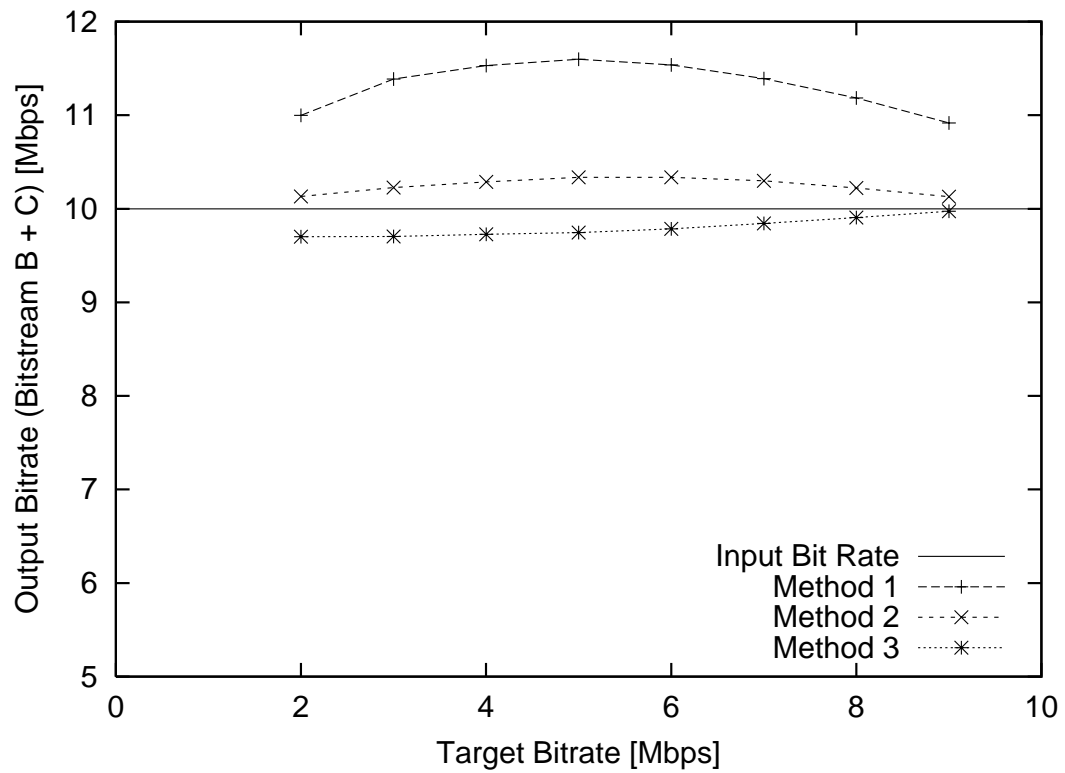


(a) Bus

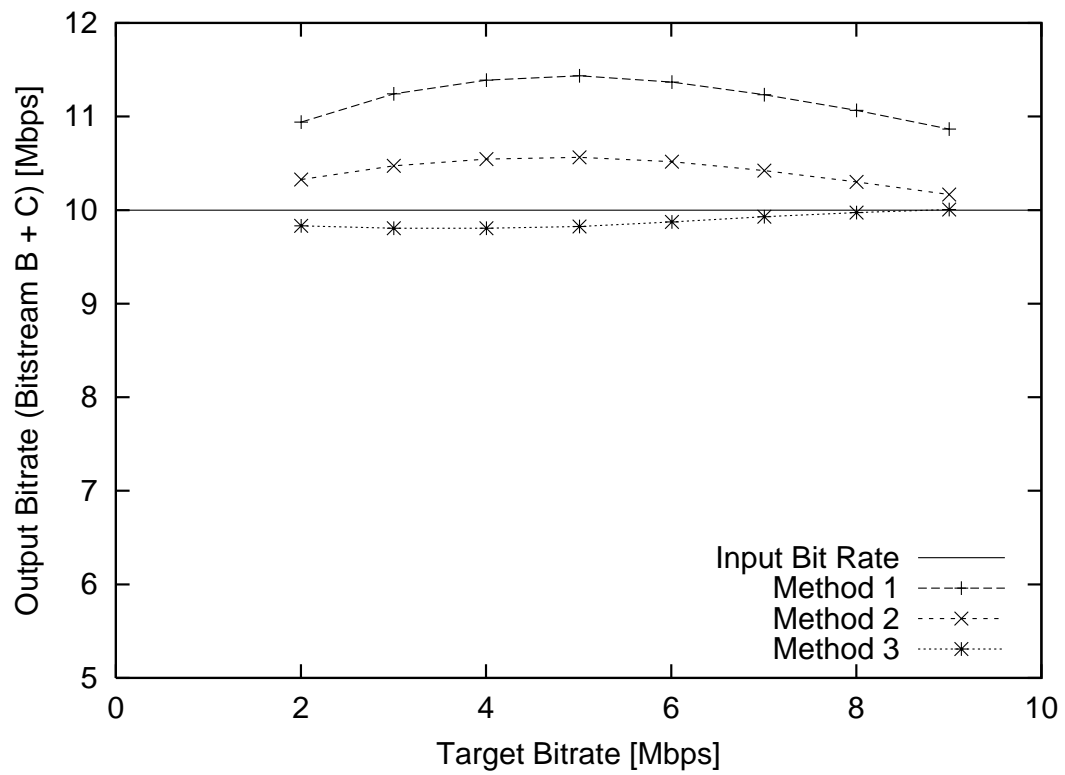


(a) Table Tennis

図 3-5: ビットレート合計値 $R_B + R_C$ の変化 (入力符号化レート 15[Mbps])



(a) Bus



(a) Table Tennis

図 3-6: ビットレート合計値 $R_B + R_C$ の変化 (入力符号化レート 10[Mbps])

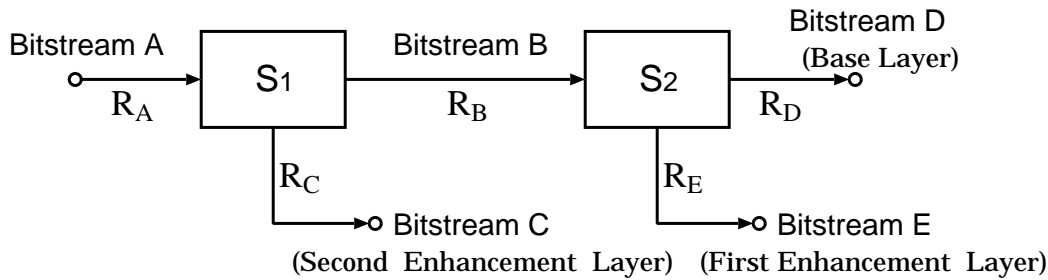


図 3-7: ストリーム分離器二段接続実験図

で定義されているのに対し、式 (3-19) より変化情報のランレングスについては $level$ の値域が $1 \leq level \leq m$ に制限されるために、MPEG-2 のそれに比べてはるかに小さな符号化表となる。したがって、適切な符号設計がなされていれば、多くのイベントに十分短い符号が割り当てられることが見込める。したがって、出力情報源 $((run, level), B_{err}(u, v))$ の特性に合わせた符号割当てにより、更なる符号化効率の改善が可能となる。

3.5.2 ストリーム分離器多段接続評価実験

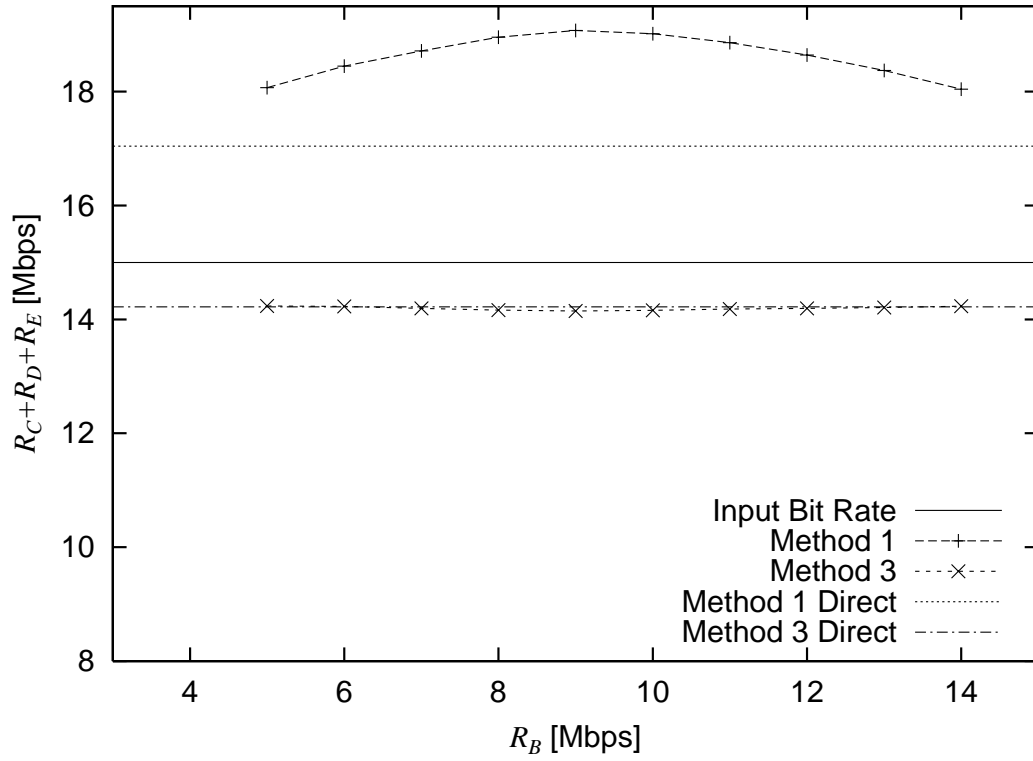
本実験ではストリーム分離器を二段に接続した場合における総符号量の変化を評価する。図 3-7 に示すように Bitstream A を第一段ストリーム分離器 S_1 へ入力し、このときの第一段出力 MPEG-2 ビットストリーム (Bitstream B) を第二段ストリーム分離器 S_2 へ入力して、第二段変換出力 MPEG-2 ビットストリーム (Bitstream D) と第二段変換差分ビットストリーム (Bitstream E) を出力する。Bitstream D と Bitstream E のビットレートをそれぞれ R_D, R_E とする。

実験では、表 3-2 に示される符号化条件における $15[Mbps]$ のビットストリームを Bitstream A として入力し、Bitstream D の目標ビットレートを $4[Mbps]$ に固定し、Bitstream B の目標ビットレートを $5 \sim (R_A - 1)[Mbps]$ までとったときの、 $R_C + R_D + R_E$ を方式 1 と方式 3 で比較する。実験結果を図 3-8 に示す。図 3-8 には直接 $4[Mbps]$ へ変換したときの結果を合わせて示す。

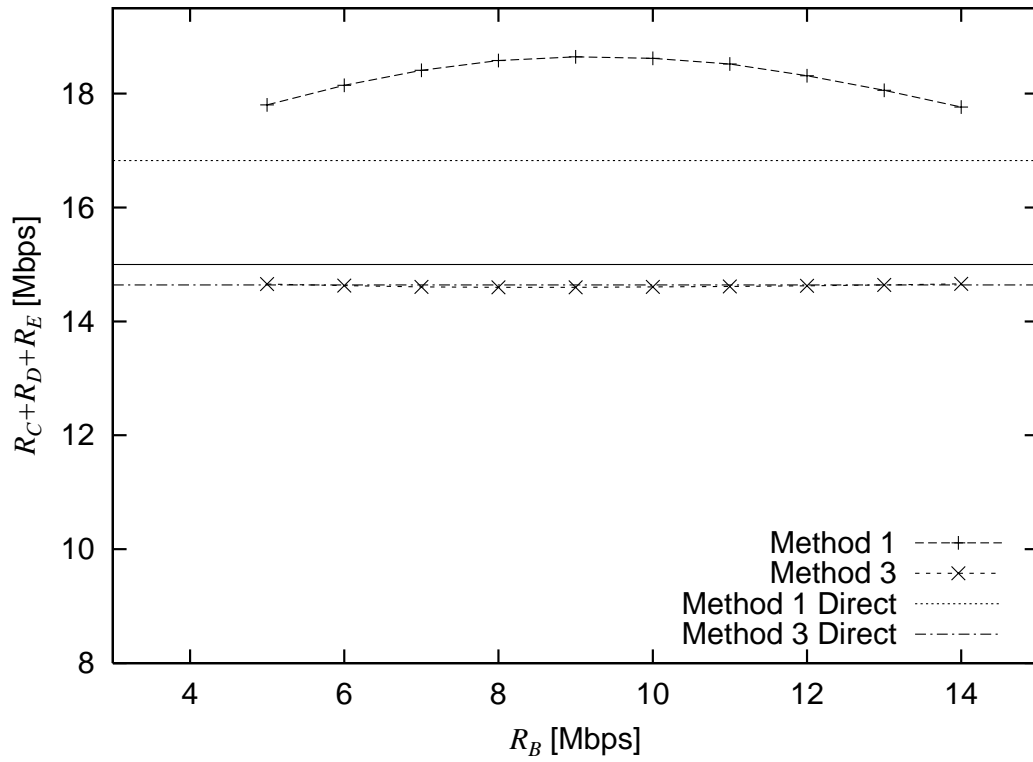
図 3-8 より方式 1 について見てみると、二段に接続して段階的にレートを削減した場合には、直接 $4[Mbps]$ へトランスコードするのに比べてビットレート合計値が大きく増大することが分かる。3.5.1 より、方式 1 は分離処理による符号量の増加が大きいため、多段接続構成により各段で発生する増加符号量が蓄積し、段数を重ねるに従い総符号量を増大させる結果を招く。

方式 3 は、二段に接続したときのビットレート合計値 $R_C + R_D + R_E$ は、直接変換時よりも若干増加するものの依然として R_A を下回っており、多段接続構成をとっても増加符号量の累積による問題は発生しないことが分かる。

スケーラビリティ機能を有する符号化器は階層数に対して加速度的に複雑化するので、現実的に利用可能な階層数は制限される。しかし、本方式では汎用的な符号化器が利用可能なだけでなく、ストリーム分離器の多段接続構成により容易に階層数を増やすことができる。さらに、ビットレートの調節も自由に行うことができ、柔軟性の問題も解決可能となる。



(a) Bus



(a) Table Tennis

図 3-8: ストリーム分離器二段接続構成におけるビットレート合計値 $R_C + R_D + R_E$ の変化

3.6 変換差分情報符号化方式の符号化特性解析

本節では 3.4 に示す提案符号化方式の性能を理論的に考察し，圧縮符号化の一手法としての評価を行う．DCT 係数をラプラス分布でモデル化した上で 3.5.1 に示す各方式における係数変化情報の符号量を導出し，分離出力の合計符号量と入力符号量を比較して評価を行う．

3.6.1 再量子化による係数変化情報量の導出

原画像信号の DCT 係数 x が符号化器においてステップサイズ Q_1 で量子化されているものとする． x の分布を，平均が 0，分散が σ^2 のラプラス分布と仮定すると， x 量子化値（量子化係数値） k_1 の発生確率 $P_1(k_1)$ は次の式 (3.22) により表される．

$$P_1(k_1) = \int_{(k_1 - \frac{1}{2})Q_1}^{(k_1 + \frac{1}{2})Q_1} p(x) dx \quad (3.22)$$

ただし，確率密度関数 $p(x)$ は式 (3.23) により与えられる．

$$p(x) = \frac{1}{\sqrt{2}\sigma} e^{-\frac{\sqrt{2}}{\sigma}|x|} \quad (3.23)$$

ここで，密度関数 $p(x)$ は $x = 0$ に対して対称であり，また量子化係数の絶対値に 1 ビットの符号ビットを追加することで正負が表現可能なため，本考察では $x \geq 0$ の領域についてのみ議論する．そこで，符号化された信号の 1 個の DCT 係数あたりの平均符号量 R_1 [bit/sample] は式 (3.24) のように表される．

$$R_1(Q_1) = \sum_{k_1=0}^{\infty} P_1(k_1) L(k_1) \quad (3.24)$$

ここで， $L(k)$ は量子化係数値 k を可変長符号化したときの符号量である．MPEG-2 では，量子化係数はジグザグスキャンされて一次元系列に並べられたあと，連続する 0 の個数（ラン長） run とそれに続く量子化値 k を組み合わせさせた事象 (run, k) に対する可変長符号により符号化する．すなわち，符号長 $L(k)$ は一意には定まらないので，本考察では各 run に対して平均値をとり $L(k)$ を表現する．そこで， (run, k) の符号長を $len(run, k)$ と表したときに符号長 $L(k)$ は次の式 (3.25) により表される [77]．

$$L(k) = \frac{\sum_{run=0}^{63} \{P(0)\}^{run} \times len(run, k)}{\sum_{run=0}^{63} \{P(0)\}^{run}} \quad (3.25)$$

ただし， $k_1 = 0$ の関してはラン長 run の値に含まれて符号化されているので $L(k) = 0$ とする．

次に，上記量子化係数 k_1 を再量子化ステップサイズ Q_2 で再量子化したときの符号量について考察する．3.3 で示したように Q_2 は整数値 m と Q_1 により式 (3.4) あるいは式 (3.5)

により算出され、再量子化値 k_2 は式 (3-1) および式 (3-2) により算出される。このとき、式 (3-19)、式 (3-20) から整数値 m により再量子化値 k_2 の有意/非有意の境界が与えられるため、再量子化によって減少する符号量 R_{red} を式 (3-26) のように表現する。

$$R_{red}(Q_1, m) = \sum_{k_1=0}^m P_1(k_1) \times \{L(k_1) - L(0)\} + \sum_{k_1=m+1}^{\infty} P_1(k_1) \times \{L(k_1) - L(k_2)\} \quad (3-26)$$

式 (3-24)、式 (3-26) より再量子化出力後の 1 個の係数あたりの平均符号量 $R_2(= R_1 - R_{red})[bit/sample]$ は式 (3-27) により表される。

$$R_2(Q_1, m) = \sum_{k_1=m+1}^{\infty} P_1(k_1) \times L(k_2) \quad (3-27)$$

次に、各方式における再量子化前後間の係数変化情報量 $R_{diff}[bit/sample]$ を導出する。

1. 方式 1

方式 1 では係数情報の変化成分として、再量子化前後の差分を算出した後に MPEG-2 の符号化方式と同様のジグザグスキャンと二次元ランレングスにより符号化処理を行う。したがって、方式 1 での係数差分情報の符号量 R_{diff} は式 (3-28) により表される。

$$R_{diff}(Q_1, m) = \sum_{k_1=0}^{\infty} P_1(k_1) \times L(k_1 - k_2) \quad (3-28)$$

2. 方式 2, 方式 3

方式 2, 方式 3 では、係数の変化成分に対し入力係数値を参照して m を境界にして別々に符号化される。まず、 $k_1 \leq m$ である係数は再量子化前の係数ブロック中の連続するゼロ係数をラン長とし、これに k_1 の値を組み合わせた二次元ランレイベントを用いてランレングス符号化する。

$$R_{rl}(Q_1, m) = \sum_{k=0}^m P_1(k_1) \times L_{rl}(k_1) \quad (3-29)$$

式 (3-29) において、 $L_{rl}(k_1)$ はランレングス符号化される係数の係数値 k_1 の符号量でありラン長分布が再量子化前に対してほとんど変わらないと仮定すると、式 (3-25) と同様にして、ランレングスイベントの符号語長 $len_{rl}(run, k)$ を用いて式 (3-30) に表現される。

$$L_{rl}(k_1) = \frac{\sum_{run=0}^{63} \{P(0)\}^{run} \times len_{rl}(run, k)}{\sum_{run=0}^{63} \{P(0)\}^{run}} \quad (3-30)$$

式 (3.30) においても式 (3.25) と同様に $L_{rl}(0) = 0$ とする。

一方、残りの $k_1 > m$ の係数に関しては、式 (3.31) に従い Δk を算出し、 Δk を可変長符号化して符号化される。 Δk に対する符号語長を $len_{res}\{\Delta k(k_1, k_2)\}$ とすると、符号量 $R_{res}[\text{bit/sample}]$ は式 (3.32) により算出される。

$$\Delta k = k_1 - \left\lfloor k_2 \times \frac{Q_2}{Q_1} + \frac{1}{2} \right\rfloor \quad (3.31)$$

$$R_{res}(Q_1, m) = \sum_{k_1=m+1}^{\infty} P_1(k_1) \times len_{res}\{\Delta k(k_1, k_2)\} \quad (3.32)$$

したがって、係数差分値の符号量 R_{diff} を式 (3.29) と式 (3.32) の和として式 (3.33) に導出する。

$$R_{diff}(Q_1, m) = R_{rl}(Q_1, m) + R_{res}(Q_1, m) \quad (3.33)$$

方式2では、ランレングスイベントに対して MPEG-2 と同一の符号語を割り当てるので、符号語長 $len_{rl}^{(2)} = len(run, k)$ となる。また $len_{res}^{(2)}(\Delta k)$ については、 Δk の値域に応じた固定長符号とする。

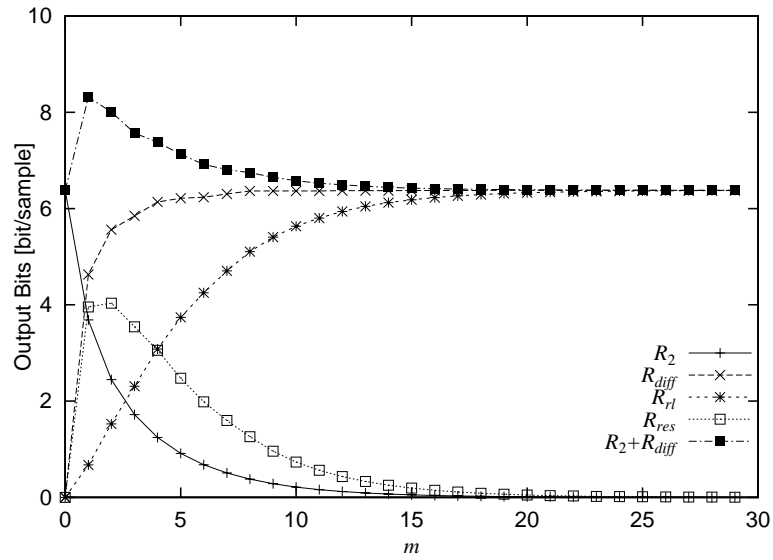
方式3では、 $len_{rl}^{(3)}(run, k)$ と $len_{res}^{(3)}(\Delta k)$ とともに発生確率の分布に基づくハフマン符号により導出する。

3.6.2 符号化特性結果

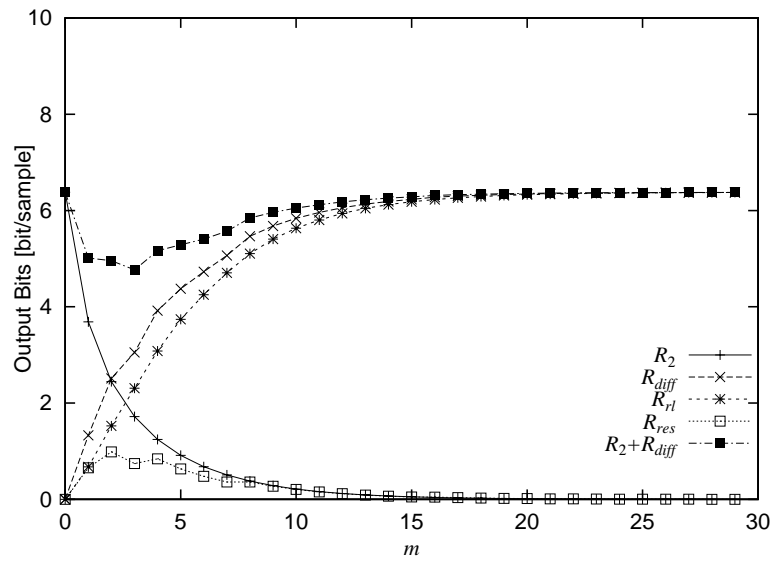
3.6.1 より導出される各符号量を算出し、提案方式の符号化特性を検証する。 $\sigma = 10.0$ 、 $Q_1 = 2$ としたときの m に対する $R_2(Q_1, m)$ 、 $R_{diff}(Q_1, m)$ 、 $R_{rl}(Q_1, m)$ 、 $R_{res}(Q_1, m)$ の変化を図 3-9 に示す。 $m = 0$ においては再量子化を行わないので、このときの符号量は入力符号量 R_1 に相当する。ただし、方式1については R_{diff} のうち、 $k_1 \leq m$ の入力係数に対する変化差分情報量を R_{rl} として、 $k_1 > m$ に対する変化差分情報量を R_{res} としている。

図 3-9 より、方式1は $m = 1$ から 10 の範囲で分離出力総符号量 $R_2 + R_{diff}$ が入力符号量を上回っていることが分かる。 R_{rl} は、 R_2 の指数関数的な減少とほぼ対象的な増加傾向を示しており m が低い領域では R_{rl} は少ない。しかし、 R_{res} が占める符号量が特に多いために、結果的に分離後の総符号量は入力時を上回り、符号化効率の低下を招く。したがって、符号化効率の改善には R_{res} に要する符号量を削減することが重要であることが分かる。

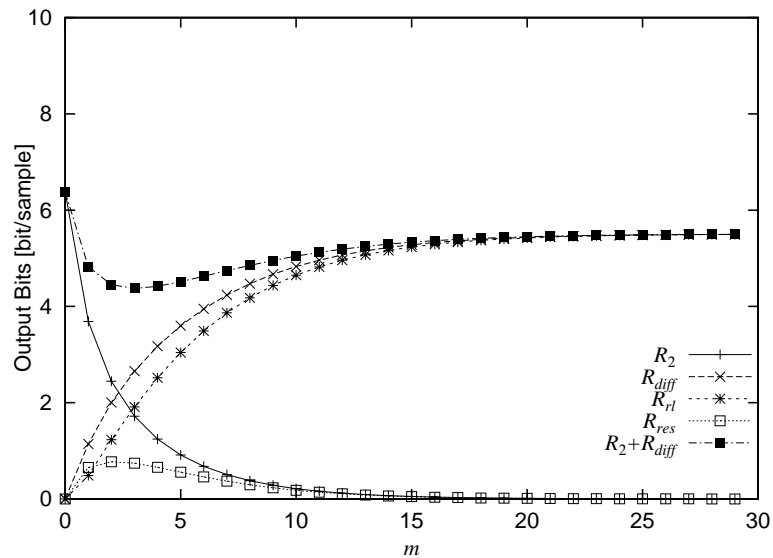
方式2、方式3では $k_1 > m$ の係数の変化情報に対して、ラン情報を符号化せず係数値のみを符号化対象とすることで、必要な符号量を削減している。これにより R_{res} の符号量が大きく改善されることが図 3-9 に示されている。また、再量子化前後間の予測符号化によって、値の大きな入力係数に対しても必要な符号量を少なくできる。このように $k_1 > m (k_2 \neq 0)$ の係数に対する変化値は、再量子化値を利用して不要な符号を排除し、再量子化値のみからでは再現できない情報成分のみを符号化対象とすることで、通常の二次元ランレングス符号化方式に比べて 1/4 程度の符号量で表現可能となる。



(a) Method 1



(b) Method 2



(c) Method 3

図 3-9: m に対する各種符号量の変化 ($Q_1 = 2$)

さらに、 $k_1 \leq m$ の入力値に対する変化値については、ランレングス符号化表を再設計することで、ランレングスの符号量 R_{rl} を大幅に削減可能であることが方式3の結果よりわかる。 m が大きい領域でも $R_2 + R_{diff}$ が R_1 を下回っており、二次元ランレングスの値域の縮小による符号語長の削減が再圧縮に効果的であることが分かる。

次に、 m とともに Q_1 を変化させ、入力符号量 R_1 と m に対する分離出力総符号量 $R_2 + R_{diff}$ の特性を図3-10に示す。図3-10より、 R_1 が大きいほど各方式の違いが顕著に現れ、提案方式は、入力係数の個数が多い高レートで符号化されたビットストリームに対して特に高い効果を発揮することがわかる。

提案方式は、係数の変化差分値に対して、量子化等の非可逆処理は行わないので、復号処理後には再量子化前と全く同一の係数情報が再現される。また、可変長符号の割り当てに、再量子化ステップ算出時に用いた整数値 m を利用することで、割り当て対象の値域を制限し、導出される可変長符号の長さを削減可能である。その結果、全く同一の係数情報が、MPEG-2 で用いられている二次元ランレングス符号化方式よりも少ない符号量で表現可能となる。

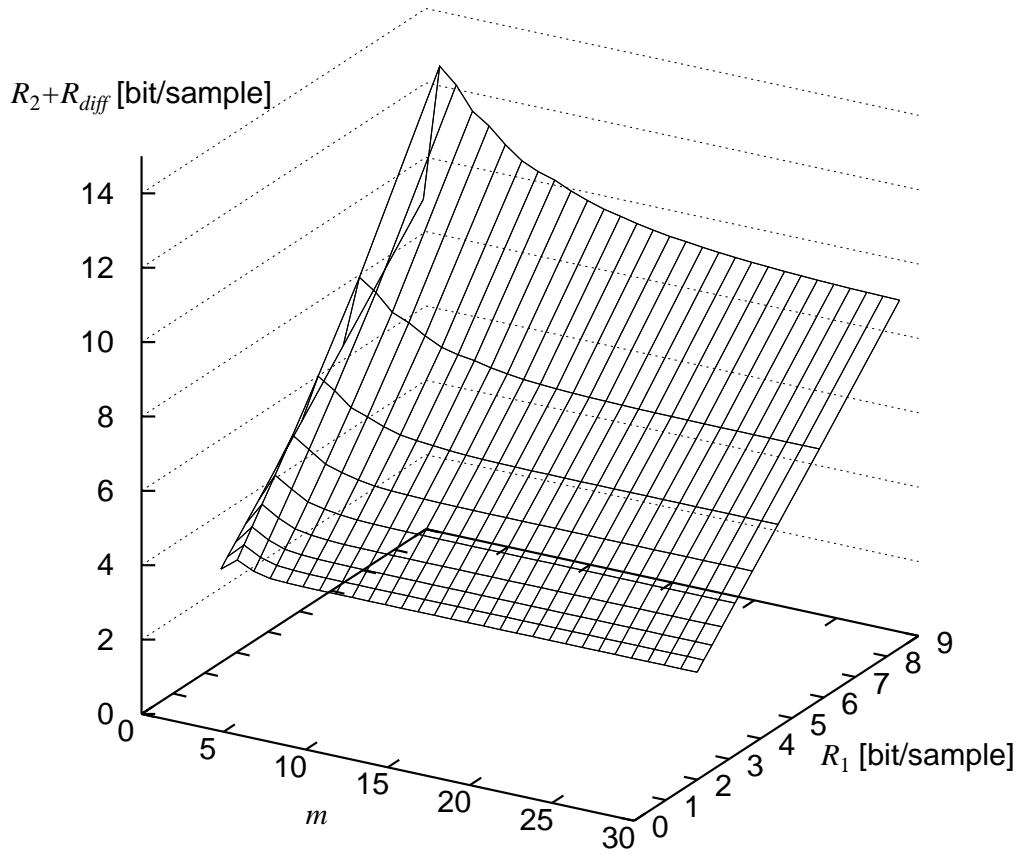
このように、本論文で述べた整数値 m による可変長符号割り当ての効率化手法は、単にトランスコードによるスケーラビリティの実現手段としてだけでなく、DCT と量子化をベースにした各種符号化方式における可変長符号化方式の新たな手法としても有用である。

3.7 まとめ

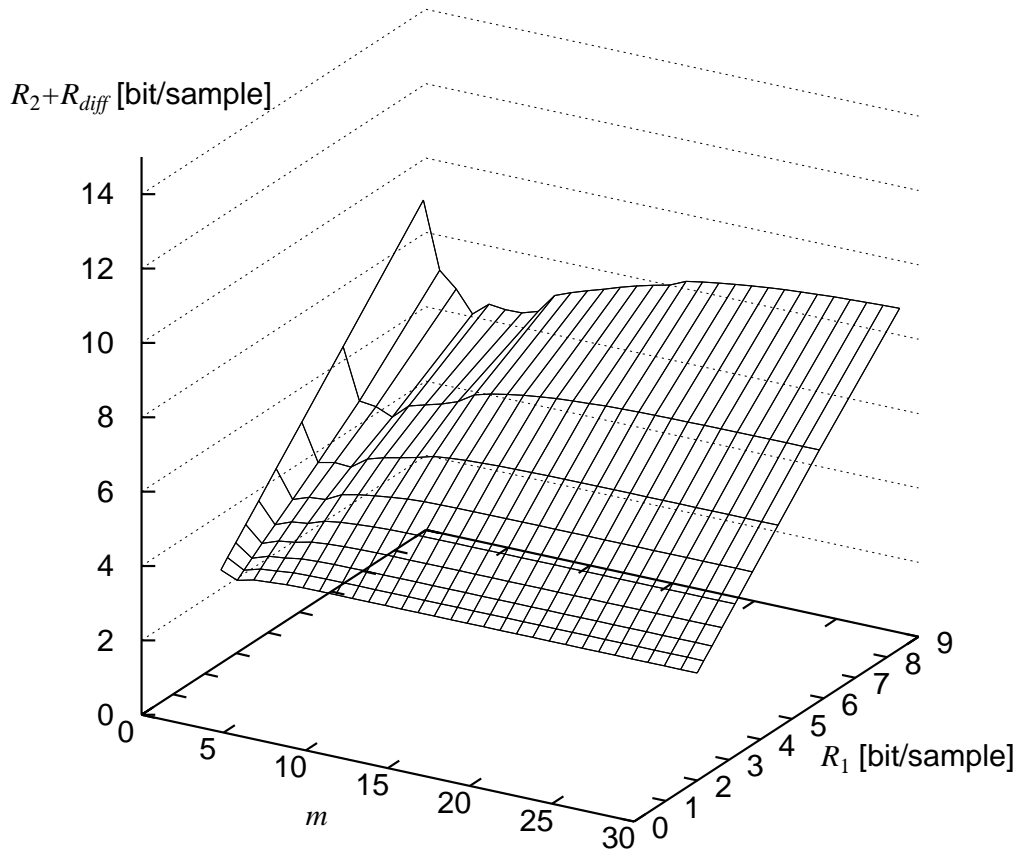
本論文では、送信/受信装置に特別な機能を追加せずに、符号化画像情報に対する変換処理により接続環境に応じた柔軟なスケーラビリティを実現可能なトランスコードに着目し、トランスコードを用いて階層符号化機能を提供することで新たなスケーラブル映像蓄積・配信方式を実現するための符号化方式を提案した。

まず、符号化器/復号器によるスケーラビリティの実現方式の問題点について整理し、トランスコードの入力と出力間の差分映像信号の符号化処理による階層符号化機能を実現方法を述べた。次に、差分映像信号を再量子化処理による DCT 係数の構成情報の変化成分として符号化するための符号化アルゴリズム及び元の係数情報の再現方法の詳細について述べた。また、評価実験から、変化成分を単純な係数ブロック間差分として表現して、これに MPEG-2 同等の符号化処理を施した場合、総符号量が大きく増大し符号化効率の低下が見られた。これに対して提案方式では、係数のブロック内の位置に関する情報が再量子化後の係数ブロックから取得可能な点に着目し変化情報符号に含めないことで大幅な符号化効率の向上を実現したところに特徴がある。また、符号化性能を理論的に考察したところ、再圧縮の効果が得られることを確認し、符号化効率の有効性を明らかにした。

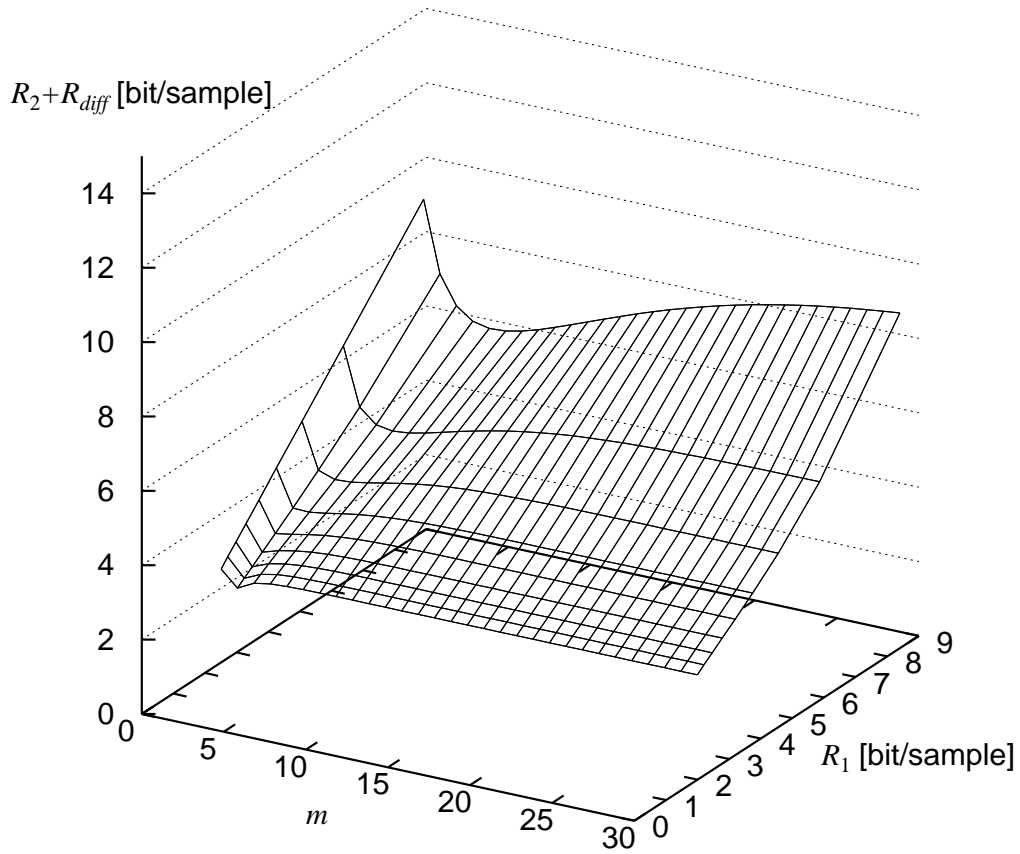
本論文で提案するビットストリーム分離・合成機能は、符号化器/復号器自体には特別な機能を要求せずに階層符号化によるスケーラビリティを実現可能という特徴を持つ。また、多段接続によって多階層のスケーラビリティを容易に実現可能とし、従来の階層符号化処理における符号化器/復号器の構造の複雑化、柔軟性等の問題を解決可能である。このように、



(a) Method 1



(b) Method 2



(c) Method 3

図 3-10: (R_1, m) に対する出力総符号量の変化

ストリーム分離・合成機能は、トランスコーディングにより階層符号化処理を実現することで、これらの二つの機能を融合する技術としてデジタル映像通信の分野で有用なものとなると考えられる。今後の課題としては、ビデオ以外の音声や同期情報などを含めた多重化ストリームを対象とする提案映像通信システムの実現があげられる。

第4章

ビットストリーム変換による多階層映像符号化方式

4.1 はじめに

第3章では、符号化済のビットストリームの情報構造の変換により階層符号化と同等の機能を実現する符号化処理アーキテクチャを提案した。本方式は基本階層と高位階層の2階層から構成されるの階層符号への変換方式であり、提供可能な品質は2通りに制限される。一方、高位階層符号の一部分を合成して意味のある映像シーンが取得可能であれば、基本階層と高位階層の中間的な品質を持ったビットストリームを取得可能となる。高位階層伝送時に要求時間と回線速度の制約がある環境において伝送可能な情報成分を選択的に伝送することで、高位階層符号の伝送にスケラビリティをもたらすことができる。

本章では、第3章で提案した2階層の階層化方式を拡張して、高位階層符号化情報の内部構造を階層化して、品質選択性の自由度を向上させた新たな階層符号変換方式を提案する。

4.2 多階層型高位階層符号による動画像伝送

本節では、図4-1に示す内部構造が複数の階層により階層化された多階層構造を有する高位階層符号を用いた動画像伝送の概念について述べる。

4.2.1 非階層符号の変換による階層符号化情報の生成

汎用符号化器より生成された非階層入力ビットストリーム (Bitstream A) を、ビットレートが削減された基本階層ビットストリーム (Bitstream B) と削減処理に伴い発生した差分信号を符号化した高位階層ビットストリーム (Bitstream C) に変換する。本処理を $\mathcal{F}[\cdot]$ と定義して式 (4.1) により表現する。

$$\mathcal{F}[\text{Bitstream A}] \longrightarrow \text{Bitstream B} + \text{Bitstream C} \quad (4.1)$$

Bitstream C は E_1 から E_n までの n 階層により階層化された多階層の情報構造を有するものとする。 E_i を部分階層符号と定義し、Bitstream C の構造を式 (4.2) に表現する。

$$\text{Bitstream C} = E_1 \oplus E_2 \oplus \cdots \oplus E_{n-1} \oplus E_n \quad (4.2)$$

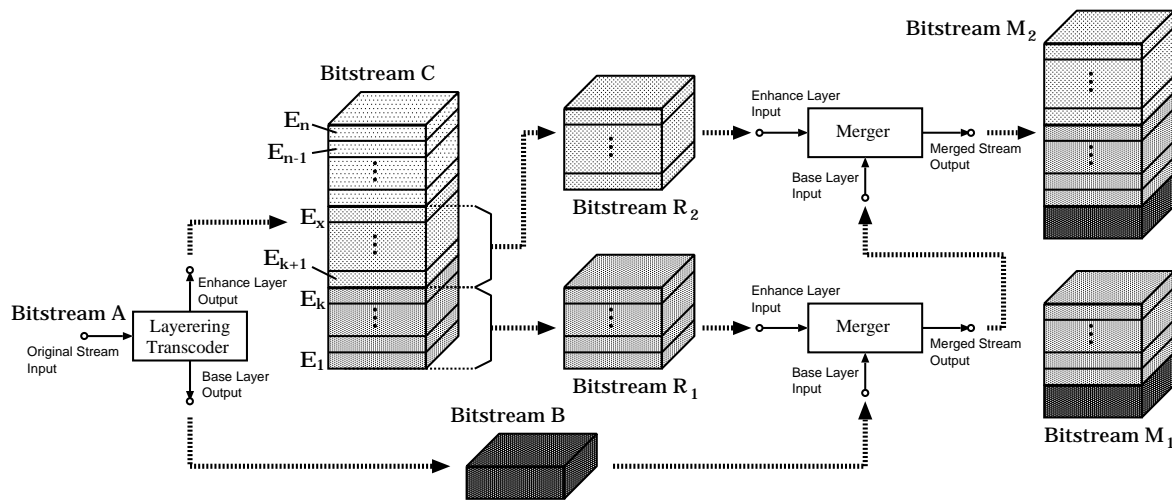


図 4-1: 多階層構造を有する高位階層符号による階層型動画伝送

4.2.2 階層選択性を有する高位階層符号の合成による高位階層映像の取得

Bitstream C の伝送において十分な伝送帯域が確保されておらず伝送時間に制約がある場合, $E_1, E_2 \dots E_n$ の中から伝送可能な階層を下位階層から順番に選択して伝送する. E_1 から $E_k (1 \leq k \leq n)$ までを選択して生成した部分的な高位階層符号を Bitstream R_1 とする. 階層選択処理を $S[\cdot]$ と定義して, Bitstream R_1 を式 (4.3) に表現する.

$$\begin{aligned} \text{Bitstream } R_1 &= S[\text{Bitstream } C, 1, k] \\ &= E_1 \oplus \dots \oplus E_{k-1} \oplus E_k \end{aligned} \quad (4.3)$$

式 (4.2), 式 (4.3) より Bitstream C の階層選択処理は式 (4.4) のように表現できる.

$$\underbrace{E_1 \oplus E_2 \oplus \dots \oplus E_{n-1} \oplus E_n}_{\text{Bitstream } C} \longrightarrow \underbrace{E_1 \oplus \dots \oplus E_k}_{\text{Bitstream } R_1} \quad (4.4)$$

Bitstream B と Bitstream R_1 を合成することで Bitstream A と Bitstream B の中間的な品質を有するビットストリーム Bitstream M_1 が生成可能となる. 本処理は式 (4.1) とは逆の処理であることを考慮して $\mathcal{F}^{-1}[\cdot]$ と定義し, 式 (4.5) に表現する.

$$\begin{aligned} \text{Bitstream } M_1 &= \mathcal{F}^{-1}[\text{Bitstream } B, \text{Bitstream } R_1] \\ &= \mathcal{F}^{-1}[\text{Bitstream } B, E_1 \oplus E_2 \oplus \dots \oplus E_{k-1} \oplus E_k] \end{aligned} \quad (4.5)$$

4.2.3 高位階層符号の再合成による蓄積映像の品質の向上

Bitstream C 中の未伝送の部分階層符号を伝送して式 (4.5) で合成された Bitstream M_1 を再合成することで上位階層の符号化情報を取得する過程を説明する.

Bitstream C 中から, E_k 以降の E_{k+1} から E_x ($k < x \leq n$) までの部分階層を抽出したビットストリームを Bitstream R_2 と定義し, 式 (4.6) に表す.

$$\begin{aligned} \text{Bitstream } R_2 &= \mathcal{S} \left[\text{Bitstream } C, k+1, x \right] \\ &= E_{k+1} \oplus E_{k+2} \oplus \cdots \oplus E_x \end{aligned} \quad (4.6)$$

第 2 章で述べたように合成出力ビットストリーム Bitstream M_2 は汎用復号器で復号, 再生されるので, 標準符号化方式に準拠した Bitstream B と同等の符号化構造となる. そこで, Bitstream M_1 を基本階層, Bitstream R_1 を高位階層として合成処理を行い, 再合成ビットストリーム Bitstream M_2 を生成する. 本過程を式 (4.7) に示す.

$$\begin{aligned} \text{Bitstream } M_2 &= \mathcal{F}^{-1} \left[\text{Bitstream } M_1, \text{Bitstream } R_2 \right] \\ &= \mathcal{F}^{-1} \left[\mathcal{F}^{-1} \left[\text{Bitstream } B, E_1 \oplus \cdots \oplus E_k \right], E_{k+1} \oplus \cdots \oplus E_x \right] \end{aligned} \quad (4.7)$$

Bitstream M_2 を構成する信号成分について考察する. Bitstream M_2 は E_1 から E_x までの部分階層に含まれる信号成分で構成される. これは, Bitstream B に対して E_1 から E_x 階層を合成した場合と等価な構成であり, 式 (4.7) は式 (4.8) として表現可能である.

$$\text{Bitstream } M_2 = \mathcal{F}^{-1} \left[\text{Bitstream } B, E_1 \oplus \cdots \oplus E_k \oplus E_{k+1} \oplus \cdots \oplus E_{x-1} \oplus E_x \right] \quad (4.8)$$

上記で述べた階層符号化映像の伝送方式は, 伝送時間と高品質性の相反する要求が求められたときの有効な解となる. まず, 基本的な品質を持つ符号化情報 (Bitstream B) を先行配送して大まかな映像内容を早期に取得する. その後, 階層化された高位階層符号 (Bitstream C) を用いて, 伝送時間, 帯域に応じて適切な階層を選択的に伝送し, 受信側では取得済の合成符号化情報と再合成する. このように, Bitstream C を構成する部分階層符号 E_i の再送・再合成を繰り返し行うことで詳細部分を表現する信号を順次伝送することにより, 受信側で高画質映像を取得する. これにより, JPEG, JPEG2000 等で実現されている静止画像のプログレッシブ伝送 [78] と同種の機能を動画像情報の伝送に実現する.

4.3 基本原理

本節では, 高位階層符号を構成する量子化係数差分信号の階層構造化のための基本原理について述べる.

4.3.1 階層構造を有する量子化係数情報の構成方法

再量子化出力係数ブロック B_2 に適用する再量子化ステップサイズ MQ_2 は初回符号化時のステップサイズ MQ_1 と整数値 m を用いて, イントラ MB の場合には式 (4.9), インター MB の場合には式 (4.10) としてそれぞれ算出される.

$$MQ_2(m) = \begin{cases} MQ_1 & \text{if } (m = 0) \\ 2m \times MQ_1 + 1 & \text{if } (m \neq 0) \end{cases} \quad (4.9)$$

$$MQ_2(m) = (m + 1) \times MQ_1 \quad (4.10)$$

$m \neq 0$ の場合に $MQ_2(m)$ および $MQ_2(m-1)$ なるステップサイズで再量子化した再量子化出力係数ブロックをそれぞれ $B_2^{(m)}, B_2^{(m-1)}$ と定義し, これらの係数ブロック中の非ゼロ係数 (有意係数) の分布について考察する.

3.3 と同様の議論により, $B_1(u, v) \leq m$ である入力係数値は再量子化演算によって 0 へ変換され, $B_1(u, v) > m$ なる係数値は $B_1(u, v)$ より小さい 0 でない値へ変換される. これより, $B_2^{(m)}$ において全ての (u, v) に対して式 (4.11) が成立する.

$$B_2^{(m)}(u, v) = \begin{cases} 0 & \text{if } B_1(u, v) \leq m \\ Q[B_1(u, v), m] & \text{if } B_1(u, v) > m \end{cases} \quad (4.11)$$

同様にして, $B_2^{(m-1)}$ 内の全ての (u, v) に対しては式 (4.12) が成立する.

$$B_2^{(m-1)}(u, v) = \begin{cases} 0 & \text{if } B_1(u, v) \leq m-1 \\ Q[B_1(u, v), m-1] & \text{if } B_1(u, v) > m-1 \end{cases} \quad (4.12)$$

ただし, $Q[\cdot]$ は再量子化演算であり, 3.3 と同様に式 (4.13) のように表される. 式 (4.13) において, 上式がイントラ MB の場合, 下式がインター MB の場合を表す.

$$Q[k, m-1] = \begin{cases} \left\lfloor \frac{k}{2m + MQ_1^{-1}} + \frac{1}{2} \right\rfloor & \text{for Intra MB} \\ \left\lfloor \frac{k + 1/2}{m1} \right\rfloor & \text{for Inter MB} \end{cases} \quad (4.13)$$

ここで, 各係数値が 0 か否かのみに着目することで, $B_2^{(m)}, B_2^{(m-1)}$ 内の非ゼロ係数の分布について考察する. まず, 演算子 $\mathcal{B}[\cdot]$ を式 (4.14) のように定義する.

$$\mathcal{B}[x] = \begin{cases} 1 & \text{if } x \neq 0 \\ 0 & \text{if } x = 0 \end{cases} \quad (4.14)$$

また, 行列 X に対しては式 (4.15) に定義する.

$$\mathcal{B}[X] = \begin{bmatrix} \mathcal{B}[X(0,0)] & \cdots & \mathcal{B}[X(l,0)] \\ \vdots & \ddots & \vdots \\ \mathcal{B}[X(0,n)] & \cdots & \mathcal{B}[X(l,n)] \end{bmatrix} = \begin{cases} 1 & \text{if } X(u, v) \neq 0 \\ 0 & \text{if } X(u, v) = 0 \end{cases} \quad (4.15)$$

B_2^m, B_2^{m-1} に $\mathcal{B}[\cdot]$ を適用することで, 式 (4.16), 式 (4.17) を得る.

$$\mathcal{B}[B_2^{(m)}(u, v)] = \begin{cases} 0 & \text{if } B_1(u, v) \leq m \\ 1 & \text{if } B_1(u, v) > m \end{cases} \quad (4.16)$$

$$\mathcal{B}[B_2^{(m-1)}(u, v)] = \begin{cases} 0 & \text{if } B_1(u, v) \leq m-1 \\ 1 & \text{if } B_1(u, v) > m-1 \end{cases} \quad (4.17)$$

式 (4.17), 式 (4.16) より式 (4.18) を得る.

$$\mathcal{B}[B_2^{(m-1)}(u, v)] - \mathcal{B}[B_2^{(m)}(u, v)] = \begin{cases} 1 & \text{if } B_1(u, v) = m \\ 0 & \text{if } B_1(u, v) \neq m \end{cases} \quad (4.18)$$

一方, B_1 において m なる係数値のみを抽出したブロック $B_*^{(m)}$ を式 (4-19) に定義する.

$$B_*^{(m)}(u, v) = \begin{cases} B_1(u, v) & \text{if } B_1(u, v) = m \\ 0 & \text{if } B_1(u, v) \neq m \end{cases} \quad (4-19)$$

式 (4-18), 式 (4-19) より式 (4-20) が成立する.

$$\mathcal{B} [B_2^{(m-1)}] - \mathcal{B} [B_2^{(m)}] = \mathcal{B} [B_*^{(m)}] \quad (4-20)$$

式 (4-20) から, $B_2^{(m-1)}$ に対して $B_1(u, v) = m$ なる値をもつ係数値のみを抽出したブロック $B_*^{(m)}$ を加えたブロックにおける有意係数の分布は, $B_2^{(m)}$ における有意係数分布と全く一致する. 各 m に対する再量子化出力ブロック内の有意係数分布に着目すると, m を 1 つ加算するごとに, B_1 中の m なる値を持つ係数値が加わった形となる. また, 式 (4-20) の漸化式を解くことで式 (4-21) を得る.

$$\mathcal{B} [B_2^{(m-l)}] = \mathcal{B} [B_2^{(m)}] + \sum_{i=0}^{l-1} \mathcal{B} [B_*^{(m-i)}] \quad (4-21)$$

ここで, $m = 0$ のときは $MQ_2 = MQ_1$ であり再量子化演算を行わない. したがって, 再量子化出力ブロックは入力係数ブロックと等しく $B_2^{(0)} = B_1$ である. 式 (4-21) において $m = l$ とおくと入力係数ブロック B_1 の有意係数の分布が式 (4-22) に表される.

$$\mathcal{B} [B_1] = \mathcal{B} [B_2^{(0)}] = \mathcal{B} [B_2^{(m)}] + \sum_{i=0}^{m-1} \mathcal{B} [B_*^{(m-i)}] \quad (4-22)$$

$1 \leq l < m$ である l における再量子化出力ブロック $B_2^{(m-l)}$ は入力係数ブロック B_1 と再量子化出力ブロック $B_2^{(m)}$ の中間ブロックである. 式 (4-21) より, $B_2^{(m-l)}$ の有意係数分布は $B_2^{(m)}$ に対して $m-1, m-2, \dots, 2, 1$ なる係数値を逐次追加することで生成することが可能となる. これより, $B_2^{(m)}$ を基本階層係数ブロック, $B_*^{(m-1)}, B_*^{(m-1)}, \dots, B_*^{(2)}, B_*^{(1)}$ を基本ブロック $B_2^{(m)}$ に対する付加ブロック (第 l 番拡張係数ブロック) とすることで, m 階層の係数ブロックが構成される.

そこで, 再量子化前後間の差分係数情報を係数値 (レベル値) により分類してランレベル系列を構成することで, 複数階層をもつ情報構造を有した高位階層の符号化方式を提案する.

4.3.2 階層数の上限値

基本階層符号化情報中の各 MB に適用する再量子化ステップ値はトランスコーダの符号制御機構により MB ごとに算出されるものであるから, 本方式により作成される階層数は MB ごとに異なる値をとる.

再量子化ステップ値 MQ_2 は階層数 m と式 (4-9), 式 (4-10) により関係づけられるため, 階層数の最大値 m_{max} は MQ_2 の値域によって制限される. 標準方式で規定される再量子化ステップ値の最大値を MQ_{max} と定義し, $MQ_1 \leq MQ_2(m) \leq MQ_{max}$ を満足する m の範

圏を求めることで m_{max} を導出する．式 (4.9) , 式 (4.10) を本不等式に代入し , イントラ MB で $m = 0$ の場合には $MQ_2 = MQ_1$ であることに注意して m について整理することで式 (4.23) を得る．ただし , 式 (4.23) において上式がイントラ MB の場合 , 下式がインター MB の場合をそれぞれ表す .

$$\begin{aligned} 0 \leq m &\leq \frac{MQ_{max} - 1}{2MQ_1} && \text{for Intra MB} \\ 0 \leq m &\leq \frac{MQ_{max}}{MQ_1} - 1 && \text{for Inter MB} \end{aligned} \quad (4.23)$$

したがって , m_{max} は式 (4.24) により求められる .

$$m_{max} = \begin{cases} \left\lfloor \frac{MQ_{max} - 1}{2MQ_1} \right\rfloor & \text{for Intra MB} \\ \left\lfloor \frac{MQ_{max}}{MQ_1} \right\rfloor - 1 & \text{for Inter MB} \end{cases} \quad (4.24)$$

MPEG-2 では $MQ_{max} = 31$ として規定されているため , m_{max} は式 (4.25) となる .

$$m_{max} = \begin{cases} \left\lfloor \frac{15}{MQ_1} \right\rfloor & \text{for Intra MB} \\ \left\lfloor \frac{31}{MQ_1} \right\rfloor - 1 & \text{for Inter MB} \end{cases} \quad (4.25)$$

4.4 多階層対応型高位階層符号化方式の提案

階層的な情報構造を有する高位階層情報の符号化処理方法を提案する . 提案方式は 第3章で述べた方式を基本にして Bitstream C の符号化方法を拡張したものであり , 基本階層の符号化方法および量子化制御方式は第3章と同様である

4.4.1 階層情報への変換アルゴリズム

提案方式の概念を図 4-2 に示す . 提案方式は , 3.4 に示す符号化方式において , $B_2(u, v) = 0$ に対応した $B_1(u, v)$ のランレングス符号化方法を以下に述べる方式へ置き換えたものであり , $B_2(u, v) \neq 0$ に対する $B_1(u, v)$ との変化差分値の符号化方法は 3.4 と同様である .

$B_2(u, v) = 0$ に対応する $B_1(u, v)$ のランレングス情報を , $B_1(u, v)$ の値ごとに別々にジグザグスキャンを行い生成する . すなわち , $B_1(u, v) \leq m$ において $B_2(u, v) = 0$ であるとすると , $B_1(u, v) = 1, 2, \dots, m$ のについて m 通りのランレベル系列を生成する . また , スキャン対象の係数レベル値を $level$ とすると , ランレングス値 run には , B_1 内のゼロ係数の他に $B_1(u, v) < level$ を満たす非零係数も含めてカウントする . すなわち , $level = 2$ の場合には , B_1 内の連続する 0 または 1 の個数を run として表現する . 以下に本アルゴリズムのフローを示す .

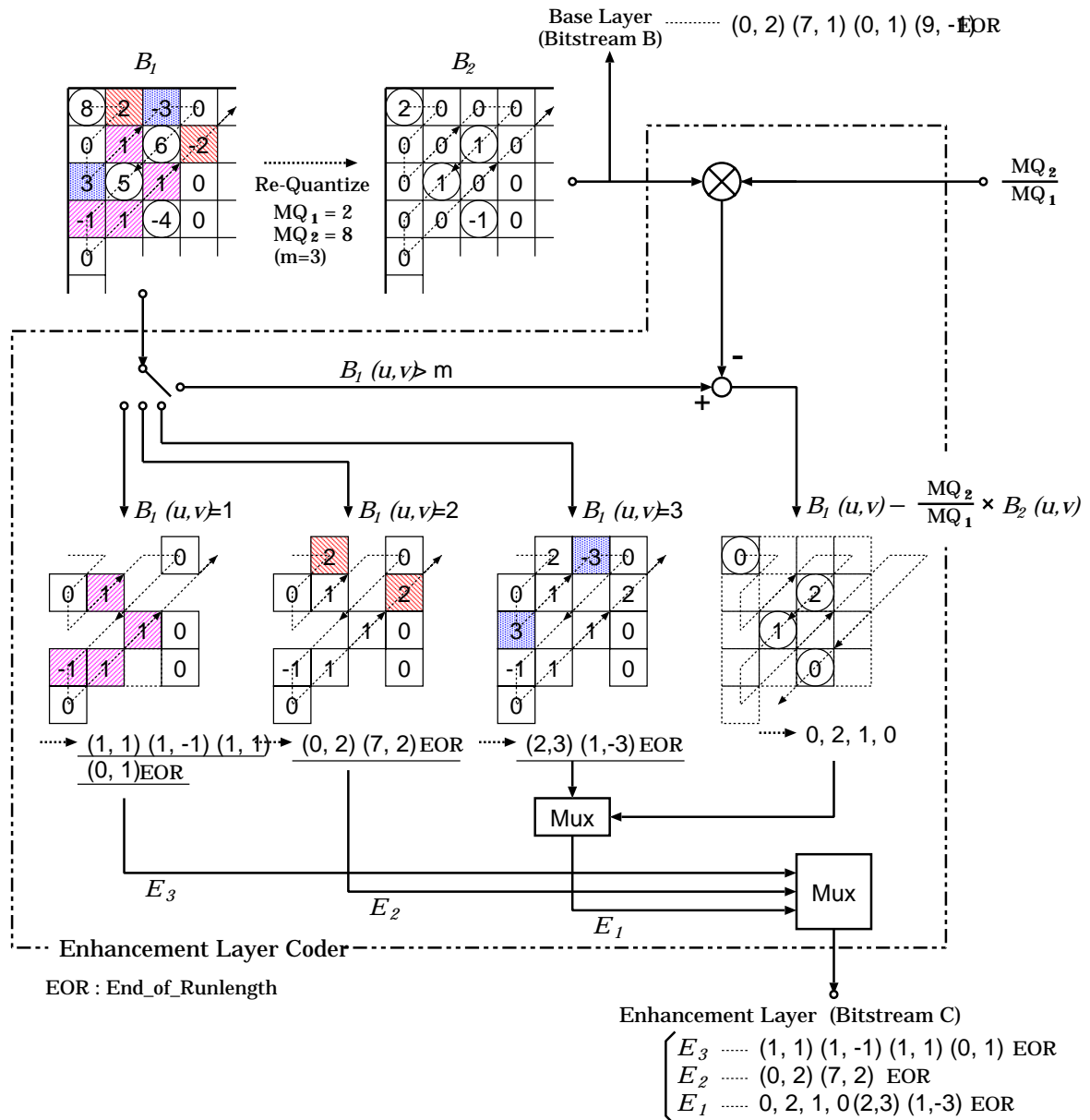


図 4-2: 係数差分符号化情報の生成

1. B_1, B_2 の係数をジグザグスキャンオーダーで読みだす。当該スキャン対象レベル値 $level$ と読み出した係数値 $B_1(u, v)$ により以下の 3 とおりの処理に分かれる。
 - 2-1. $B_1(u, v) < level$ の場合
ランレングスカウンタ c を 1 つだけ増加し, 3. の処理へ進む。
 - 2-2. $B_1(u, v) > level$ の場合
3. へ進む。
 - 2-3. $B_1(u, v) = level$ の場合
カウンタ c の値を run としたランレベルイベント ($run, level$) を生成し可変長符号化する。カウンタ c を 0 にリセット。 run は $B_1(u, v)$ 内の連続する $B_1(u, v) < level$ となる係数の個数を表す。3. へ進む。

3. ブロック末尾ならば、当該スキャン対象レベル値に関するランレベル系列の末尾を表す符号 `End_of_Runlength` を符号化する。
4. $level = \max Level$ ならば終了。そうでなければ $level$ に 1 を加算して、スキャン位置を $(u, v) = (0, 0)$ にリセットして 1. へ戻る。

図 4-2 から具体例を述べる。図 4-2 では、 $B_1(u, v) \leq 3$ において $B_2(u, v) = 0$ となる場合を表している。このときには、 $level = 1, 2, 3$ の 3 つのランレベル系列を生成する。まず、スキャン対象レベル値 $level$ を 1 にセットし、 $B_1(u, v) = 1$ である係数に関するランレベル系列を生成する。 $(u, v) = (0, 0)$ および $(1, 0)$ においては $B_1(u, v) > 1$ なので 2-2. へ該当し、このときは何もせずに次のスキャン位置へ進めるのみである、次に、 $(u, v) = (0, 1)$ において $B_1(0, 1) = 0$ となるからカウンタ c を 1 つ加算する。 $(u, v) = (1, 1)$ において $B_1(1, 1) = 1$ となるから 2-3. の処理が行われる。ここまでカウントされた $B_1(1, 1) < level$ である係数の個数は 1 個なので $(run, level) = (1, 1)$ をランレベル情報として出力する。以下同様の手順により、 $level = 1$ に関するランレベル系列を生成する。

$level = 1$ のランレベル系列の生成が終了したら、スキャン位置を $(0, 0)$ へ戻し、 $level = 2$ として同様の処理を行う。このときには、 $(u, v) = (1, 1), (0, 3), (1, 3), (2, 2)$ の $B_1(u, v) = 1$ となる係数もランレングス値に含まれるカウントされる。

ランレベルイベントの可変長符号化方法

上記の方式により生成されたランレベルイベントは係数レベル値ごとにグループ分けされて、境界符号 `End_of_Runlength` で区切られる。そのため、復号時にはランレベル系列の符号化順序と `End_of_Runlength` 符号の出現回数により $level$ の値を算出可能である。そこで、ランレベルイベントの符号化処理では、 run および $B_1(u, v)$ のサインビットのみを可変長符号化して、 $level$ を符号化しないことで差分情報の符号量を削減する。

階層情報の構成

入力量子化係数ブロック B_1 は、基本階層と m 通りの高位階層による、 $m + 1$ 階層で構成される多階層符号化情報へ変換される。式 (4.21) より基本階層係数ブロック $B_2^{(m)}$ に対して、 $m, m - 1, m - 2, \dots$ なる順番で B_1 内の非零係数を降順に印加して中間ブロック $B_2^{(m-l)}$ を構成する。そのため、第 l 番高位階層符号を構成するランレベル情報の $level$ の値は $level = m + 1 - l$ なる関係を持つ。ただし、 $B_2(u, v) \neq 0$ の場合における再量子化演算に伴う剰余成分 (3.4.1 (2) 参照) は、第 1 高位階層の符号化情報として出力する。すなわち、第 1 高位階層は、まず本剰余成分信号が符号化され、その後ランレベル系列が続いて符号化される構成をとる。

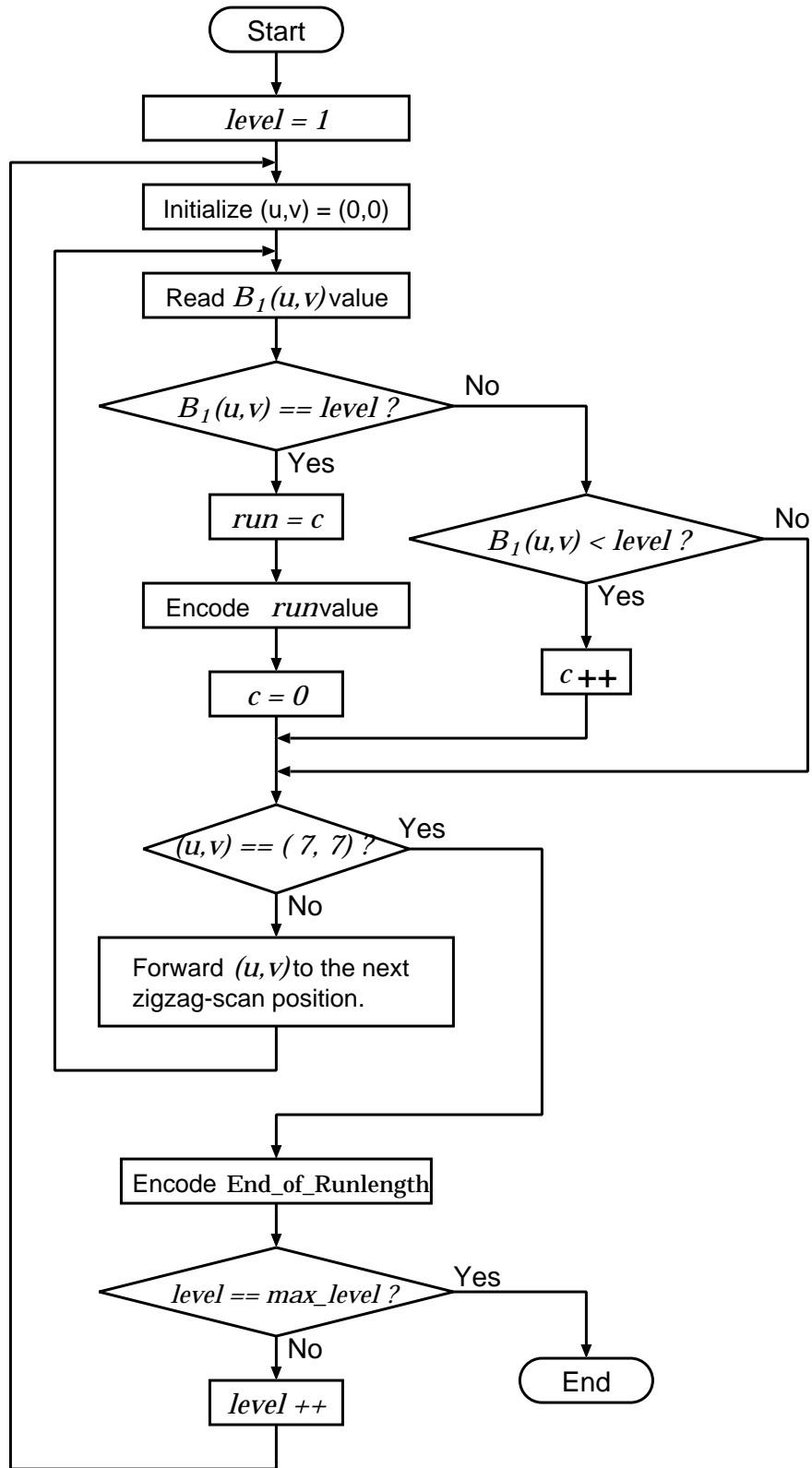


図 4-3: 分離処理フロー

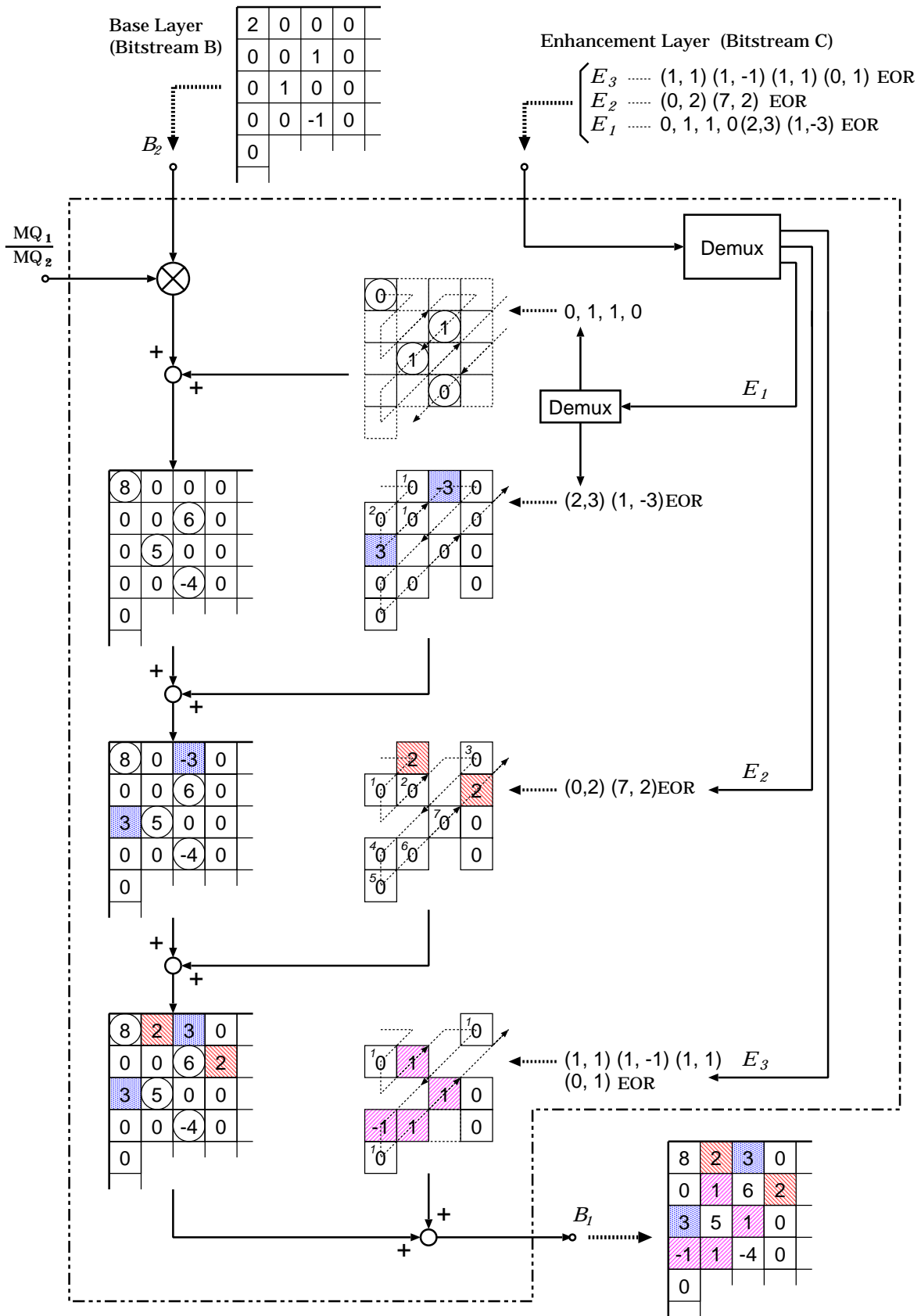


图 4-4: 合成方式

4.4.2 非階層符号の合成方法

4.4.1 より生成された基本階層と高位階層係数情報から再量子化前の係数ブロックを再構成する方法を示す。まず、 B_2 内の非ゼロ係数に関して再量子化前の値へ復元する。本処理は 3.4.2 と同様である。次に、ランレベル系列情報から残りの係数を復元する。復元対象の係数値を $level$ としたときに、復号ランレベル情報における run は $level$ より小さい値を持つ係数値の連続する個数を表している。そのため、挿入場所を示すインデックス値を、復元ブロック中の $level$ 以上の係数をスキップしながら run の数だけカウントアップして復号ランレベル情報のブロック内位置を特定し、 $level$ を挿入する。

1. run を復号し、復元対象係数レベル値 $level$ からランレベルイベント $(run, level)$ を生成する。カウンタ $p = run$ にセットする。
2. 復元係数ブロック B_1^* 内における当該スキャン位置 (u, v) の係数値 $B_1^*(u, v)$ により、以下の 2 とおりの処理へ分かれる。
 - 3-1. $B_1^*(u, v) > level$ の場合
スキャン位置 (u, v) を 1 つ進めて 4. へ進む。
 - 3-2. $B_1^*(u, v) \leq level$ の場合
 p を 1 つだけ減少させ、スキャン位置 (u, v) を 1 つ進める。4. へ進む。
4. $p = 0$ ならば、このときのスキャン位置 (u, v) に $level$ をセットし 5. へ進む。それ以外の場合は 2. へ戻る。
5. 次の符号が End_of_Runlength ならば 6. へ進む。それ以外の場合は 1. へ戻る。
6. $level = 1$ ならば終了する。そうでなければ $level$ を 1 だけ減少させ、スキャン位置を $(0, 0)$ へリセットし 1. へ戻る。

4.4.3 部分合成ブロックの生成方法

提案方式では B_1 中の有意係数のレベル値ごとに別々にスキャンすることで、高位階層符号は所定の量子化レベル値のみを選択的に合成可能な情報構造を有する。そのため、4.4.1 の手順で生成される m とおりのランレベル系列の中から特定のものを抽出して基本階層と合成することで、 B_1 と B_2 の中間ブロック B_M を生成可能となる。さらに、 B_M を基本階層として、残りの高位階層符号とともに合成器へ入力することで、再帰的な合成処理を実現する。本節では、図 4-6 を用いて再帰的な合成処理の具体例を説明する。

(1) 第 1 部分合成ブロック $B_M^{(1)}$ の合成

基本階層係数ブロック B_2 と第 1 高位階層係数信号 E_1 の合成方法を説明する。 E_1 は、 B_2 内の各有意係数に対応する再量子化演算時の剰余値符号と $level = 3$ なる係数値を有する

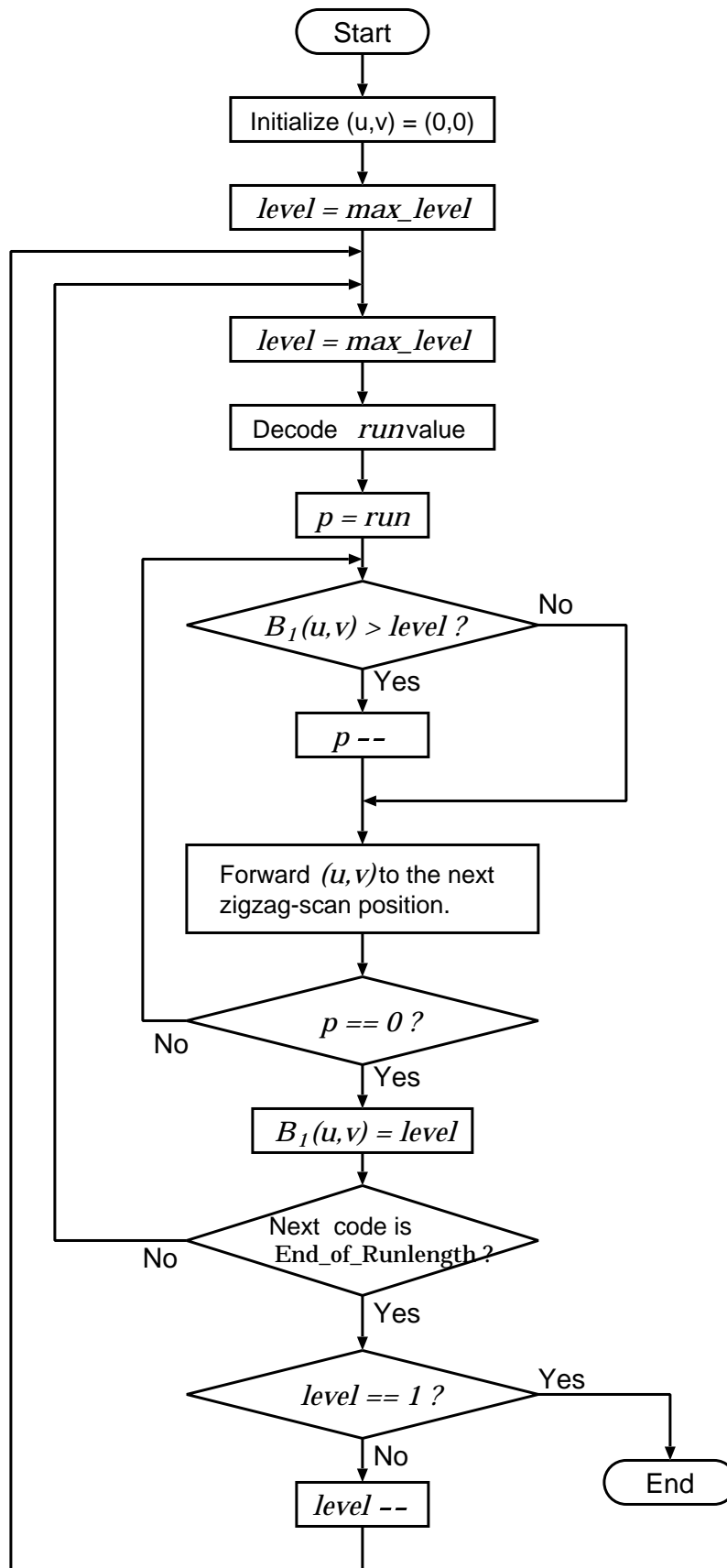


図 4-5: 合成処理フロー

ランレベル系列で構成される．まず，剰余符号 $0, 2, 1, 0$ を $B_2(u, v) \neq 0$ なる各係数に対応させて，式 (3.21) により $B_1(u, v)$ を算出し B_2 中の各有意係数の再量子化前の値を復元する．次に，ランレベル系列 $(2, 3)(1, -3)$ から $B_1(u, v) = 3$ の係数値が復号される．ここで， $B_1(u, v) > 3$ なる係数の場所は B_2 内の有意係数として復元済であるので，3 より大きい係数値をスキップしながらスキャンして，各イベントに対応する挿入位置を特定する．以上の手順により，3 以上の有意係数で構成される係数ブロックを復元して $B_M^{(1)}$ を構成する．

(2) 第2部分合成ブロック $B_M^{(2)}$ の合成

第1合成ブロック $B_M^{(1)}$ を基本階層として第2高位階層係数信号 E_2 と再帰的に合成して，第2部分合成ブロック $B_M^{(2)}$ を生成する方法を述べる． E_2 内のランレベル系列 $(0, 2)(7, -2)$ を復号して $B_1(u, v) = 2$ なる係数情報を復号する．ランレングス値には2より大きい係数値の個数は表現されていないが，これらの係数は(1)の段階で復元済であり $B_M^{(1)}$ 内の有意係数として特定可能である．そこで， $B_M^{(1)}$ の有意係数をスキップしながらスキャンして，これらの各ランレベルの挿入位置を特定して $B_M^{(2)}$ を構成する．

(3) 第3部分合成ブロック $B_M^{(3)}$ の合成

第2合成ブロック $B_M^{(2)}$ と第3高位階層係数信号 E_3 から再量子化前の係数ブロック B_1 を再構築する手順を述べる． E_3 を復号して $(1, 1)(1, -1)(1, 1)(0, 1)$ なるランレベル系列を得る．(2)と同様の方法により各ランレベルの挿入位置を特定して， B_1 を再構成する．

以上の(1)から(3)で述べた手順に基づき B_2 に対して E_1, E_2, E_3 を順次合成することで，最終的に再量子化前の係数ブロック B_1 を再構築可能となる．

4.5 評価実験

提案符号化方式について，(i) 階層符号変換時の総符号量，(ii) 階層数に対する総符号量変動，(ii) 中間合成映像の画質，の各観点についてシミュレーション実験により評価する．実験用ビットストリームの符号化条件を表4-1に示す．量子化ステップ MQ_1 の制御に関しては，(i) では TM5 方式，(ii)(iii) では固定量子化ステップで符号化されたビットストリームをそれぞれ使用する．

4.5.1 階層符号変換時の総符号量に関する評価実験

本実験では，Bitstream B と Bitstream C の合計符号量の観点で，提案方式と第3章で述べた2階層方式の符号化効率を比較する．表4-2のようにそれぞれの方式を定義する．

実験条件は5.5.1と同様とする．入力ビットストリーム Bitstream A，基本階層ビットストリーム Bitstream B，高位階層ビットストリーム Bitstream C のビットレートをそれぞれ R_A, R_B, R_C [Mbit/sec] と定義し， R_B を $2 \sim R_A - 1$ [Mbit/sec] まで変化させて階層符号へ

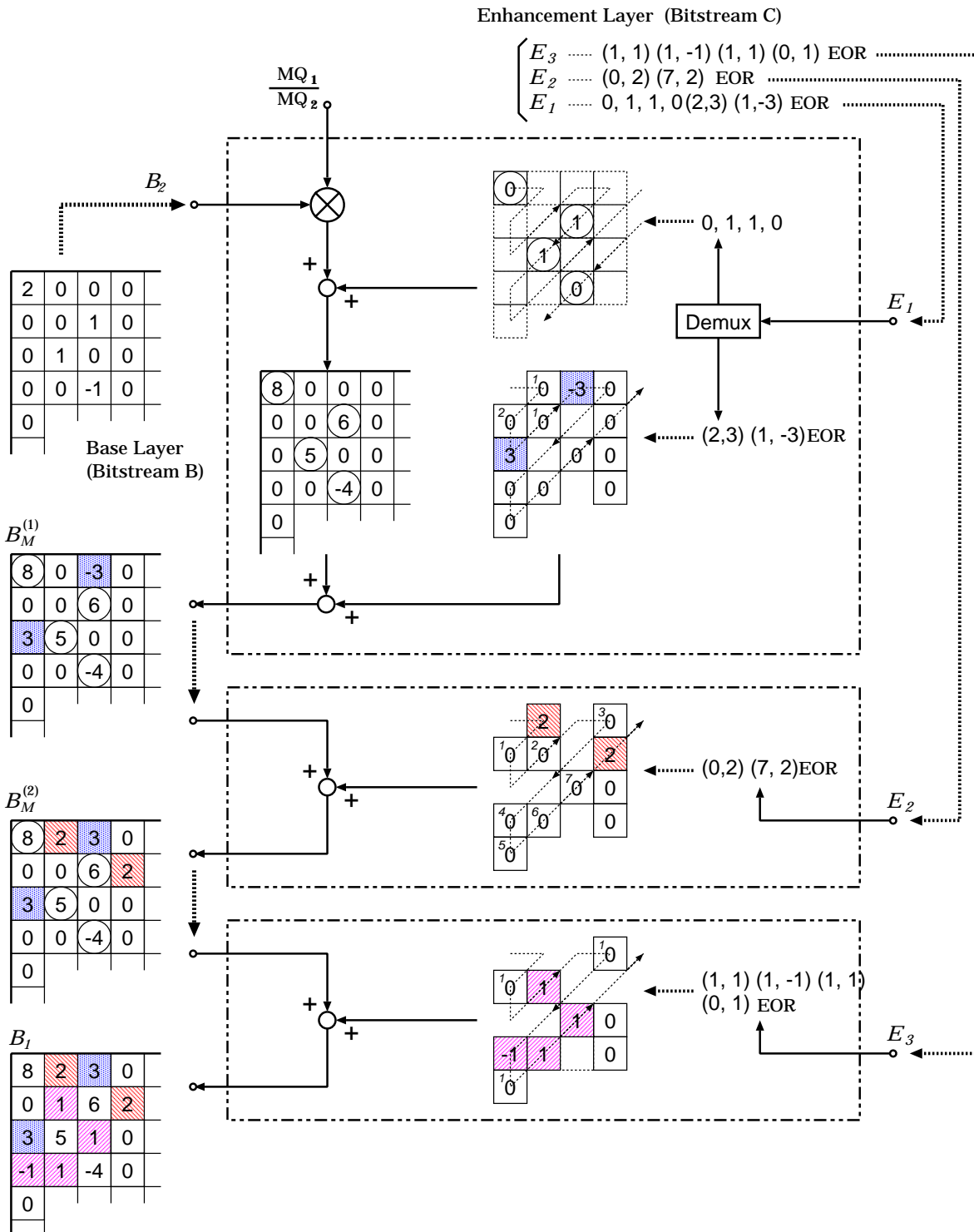


図 4-6: 差分ストリームの抽出による部分合成

表 4-1: 実験用ビットストリーム符号化条件

符号化方式	MPEG-2 Video MP@ML
符号化レート	15, 10 [Mbps]
符号量制御方式	TM5 方式 または 固定量子化ステップ
GOP 構造	N=15, M=3 (固定)
動き補償, DCT 方式	フレーム動き補償, フレーム DCT (フィールドモード OFF)
テストシーケンス	Bus, Table Tennis
画像フォーマット	ITU-R BT.601 4:2:0 Format 輝度信号 704pel × 480 line 色差信号 352pel × 240 line
符号化フレーム数	150 [Frame]

表 4-2: 比較対象方式

方式	スキャン方式
1	提案方式 (多階層型高位階層符号化方式)
2	2 階層型方式 (第 3 章)

変換したときの基本階層と高位階層の合計ビットレート $R_B + R_C$ を評価する。Bitstream C の可変長符号化テーブルは、方式 1, 方式 2 にそれぞれについてランレンジイベントの確率分布の測定実験の結果から最適設計されたテーブルを使用する。作成方法は 5.5.1 と同様であり、7 種類の標準映像を表 4-1 に示す条件で符号化して生成したビットストリームを用いて、基本階層のビットレートが $2 \sim R_A - 1$ [Mbps] の範囲で変換処理を行いランレンジの確率分布を取得し、ハフマン符号割り当てアルゴリズムを適用して符号化表を設計する。

R_B に対する $R_B + R_C$ の特性を図 4-7, 図 4-8 に示す。方式 2 では、 B_1 内の m 以下の係数はまとめて 1 本のランレベル系列として表現されるため、 $run=0,1$ などの短いランレンジ値が多く発生する。それに対して方式 1 では、入力係数レベル値ごとに別々にスキャンし、ランレンジ値にはゼロ係数だけでなくスキャン対象のレベル値よりも低い有意係数の個数も含まれる。そのため、方式 2 と比較して長いランレンジ値が発生する確率が高く、両方式間でのランレンジ値のエントロピーの違いが総符号量へ影響すると考えられる。

方式 1 ではランレベル系列ごとに別々に End_of_Runlength 符号が必要となるため、これに起因した符号量増加要因が存在する。しかし同時に、本符号を用いることで係数レベル値の自動算出が可能となり符号量の節約可能という特徴を有する。

以上の観点で図 4-7, 図 4-8 より各方式を比較評価すると、両方式間で方式間の明確な差は見られないことから、係数レベル値の自動算出の効果は符号量の増加要因を上回り、多階層構造へ変換しても 2 階層方式とほとんど変わらない符号化性能を有すると考えられる。

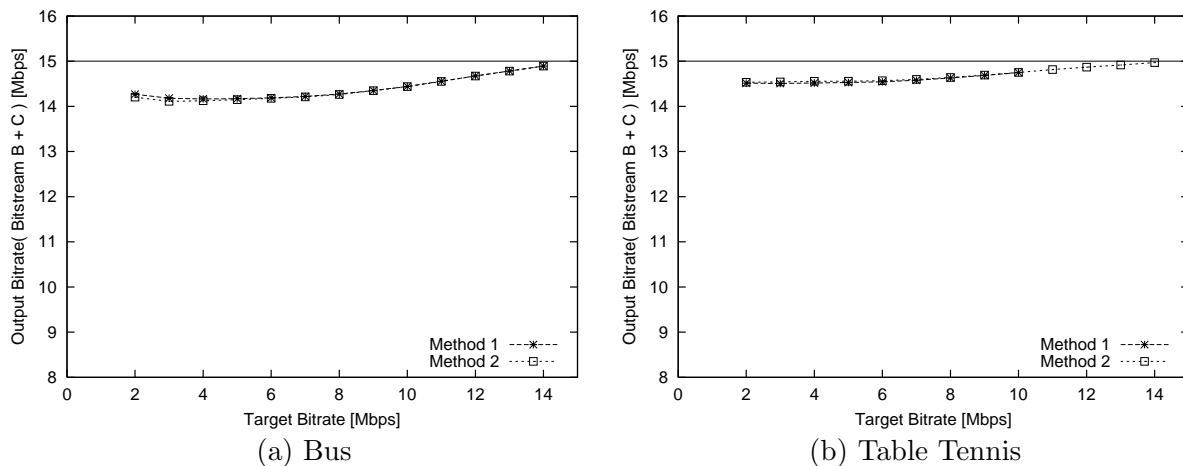


図 4-7: R_B に対する合計ビットレート $R_B + R_C$ の変化 (入力符号化レート 15[Mbit/sec])

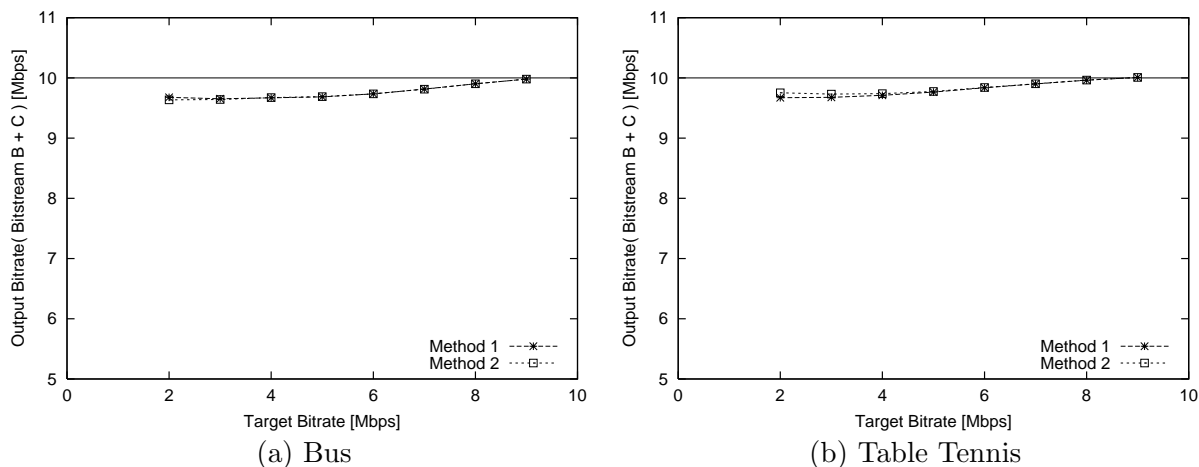


図 4-8: R_B に対する合計ビットレート $R_B + R_C$ の変化 (入力符号化レート 10[Mbit/sec])

4.5.2 階層数に対する総符号量変動に関する評価実験

本実験では、階層数に対する総符号量符号の特性を評価する。式(4.25)より階層数の最大値は符号化時点での量子化ステップ値 MQ_1 により制限される。そのため、MBごとに MQ_1 が異なると有効な階層数がMBごとに変動し、 MQ_1 が大きいMBでは指定した階層数が達成できない場合がある。そこで、表4-1に示す符号化条件において量子化ステップ値 MQ_1 を固定値にして符号化された評価用ビットストリームを実験に使用する。階層数を式(4.25)で与えられる最大値 m_{max} までの範囲にとり、階層数 m に対する基本階層と高位階層の合計符号量の特性を図4-9に示す。ここで、 MQ_1 と m_{max} の関係は表4-3のとおりである。

表 4-3: MQ_1 に対する m_{max} の関係

MQ_1	2	4	6
m_{max}	7	3	2

図 4-9 より、方式 1 と方式 2 の間で総符号量の差はわずかであり、階層数自体にも依存しない特性であることが分かる。提案方式の符号化効率の有効性は階層数が多い場合でも成立し、階層数の増加による効率低下を回避可能な方式といえる。一方、 MQ_1 による特性の違いから、符号化効率は階層数よりも MQ_1 の影響の方が支配的である。 $MQ_1 = 2$ の場合には Bus, Table Tennis とともに入力符号量に対して約 5% 少ないのに対して、 $MQ_1 = 4$ では入力符号量とほぼ等しくなる。すなわち、 $MQ_1 = 4$ で MPEG-2 と同等の符号化性能となる。また、 $MQ_1 = 6$ ではわずかに符号量の増加しており、入力ブロック B_1 の有意係数の個数の減少とともに符号化効率は低下する。この入力ビットストリームの符号量に依存する特性は第 3 章と同様の傾向である。以上より、多階層化による再圧縮効果は依然として有効であり、提案方式は階層数に関係なく第 3 章で提案した 2 階層方式と同等の符号化性能を実現可能である。

4.5.3 中間合成映像品質評価実験

本実験では、Bitstream B に Bitstream C を部分的に合成したときの、部分合成ビットストリームの符号量と画質を評価する。実験用ビットストリームの量子化ステップ MQ_1 および各 MB の階層数は固定値とする。

式 (4-22) より、 B_2 に k 番高位階層ブロック $B_*^{(m-k)}$ までを合成したブロック $B_M^{(m-k)}$ は、 B_1 を $MQ_2(m-k)$ なるステップ値で再量子化したブロック $B_2^{(m-k)}$ と同等の有意係数分布を有する。そこで、実験用ビットストリームを直接 $MQ_2(m-k)$ で固定的に再量子化した場合を基準にして評価する。Bitstream B に対して Bitstream C の部分階層を 1 階層ずつ合成したときの符号量 (合成器の入力および出力符号量) と画質との関係を図 4-10 に示す。

(1) 中間合成出力符号量の特性

図 4-10 より部分合成ビットストリームのレート歪み特性を見ると、同一の SN 比を得るための符号量が、直接再量子化した場合と比較して約 50~60% 増大しており符号化効率は大きく低下すること確認される。4.4.3 で述べたように、各高位階層符号は $m, m-1, \dots, 2, 1$ と降順の順番で B_2 と合成される。このとき、最終的には B_1 と全く同等のブロックを再構築するために、部分合成出力ブロック $B_M^{(1)}$ には再量子化演算を適用せずに、直接ジグザグスキャンして 2 次元ランレングスによるハフマン符号化する。 k 番階層までを合成したブロック $B_M^{(m-k)}$ は、 $MQ_2(m-k)$ なるステップサイズで再量子化したブロック $B_2^{(m-k)}$ と同等の有意係数分布となるが、 $B_M^{(m-k)}$ は $m-k$ 以上の有意係数値で構成されている。 $B_M^{(m-k)}$ をジグザグスキャンすると、 $B_1(u, v) = 1, 2$ などの係数が存在した場所がランレングス値としてカウントされるので、ランとレベルの双方が大きな値となる傾向がある。標準符号化表では、長いランとレベルに対しては長い符号語を割り当てているために、結果として符号量は増大すると考えられる。

(2) 部分階層符号量の特性

提案方式では、End_of_Runlength の出現回数によりレベル値を算出するので、ランレベル系列長が長いほど多くのランレベルイベントのレベル値をまとめて表現可能となるので符

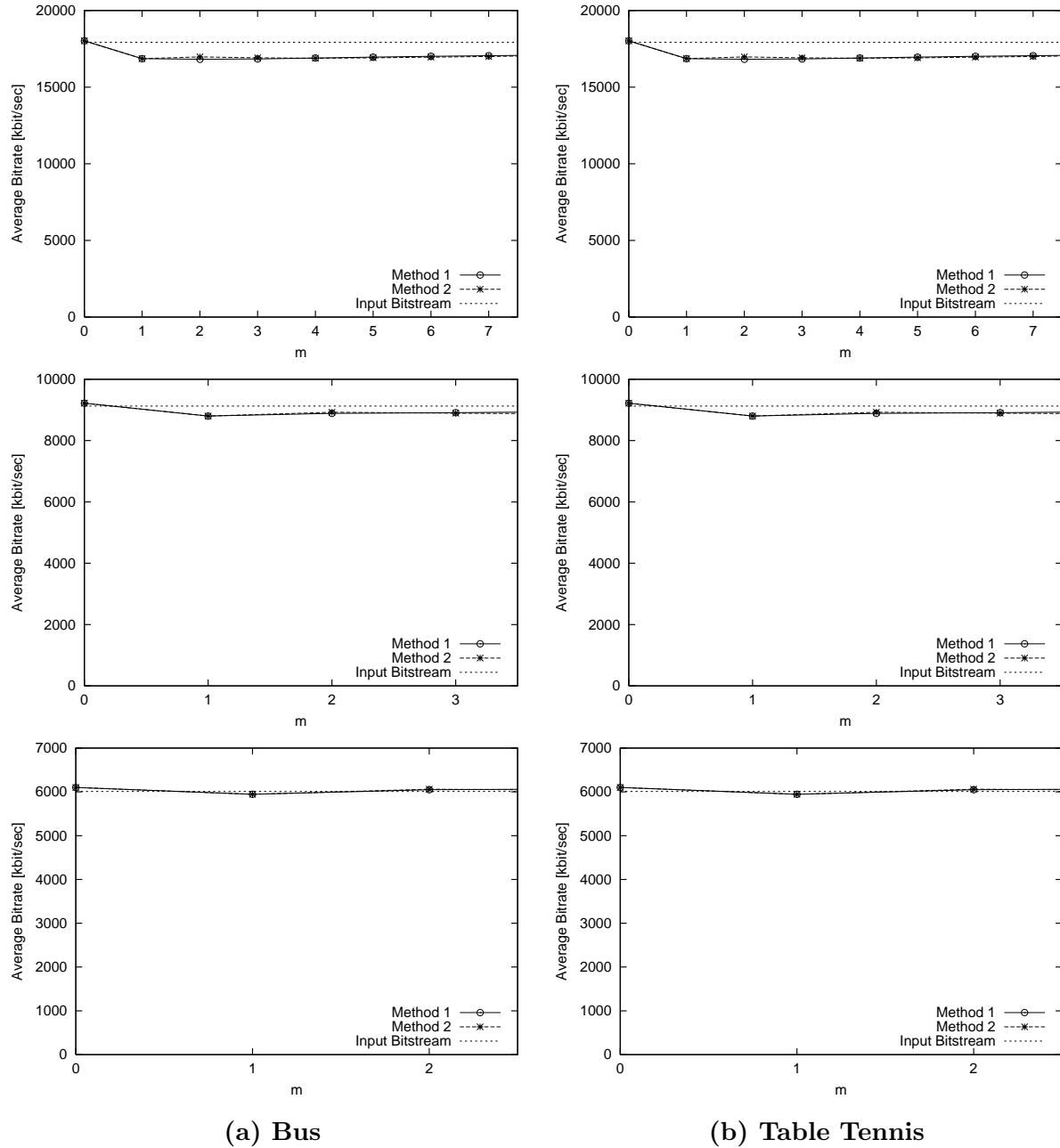
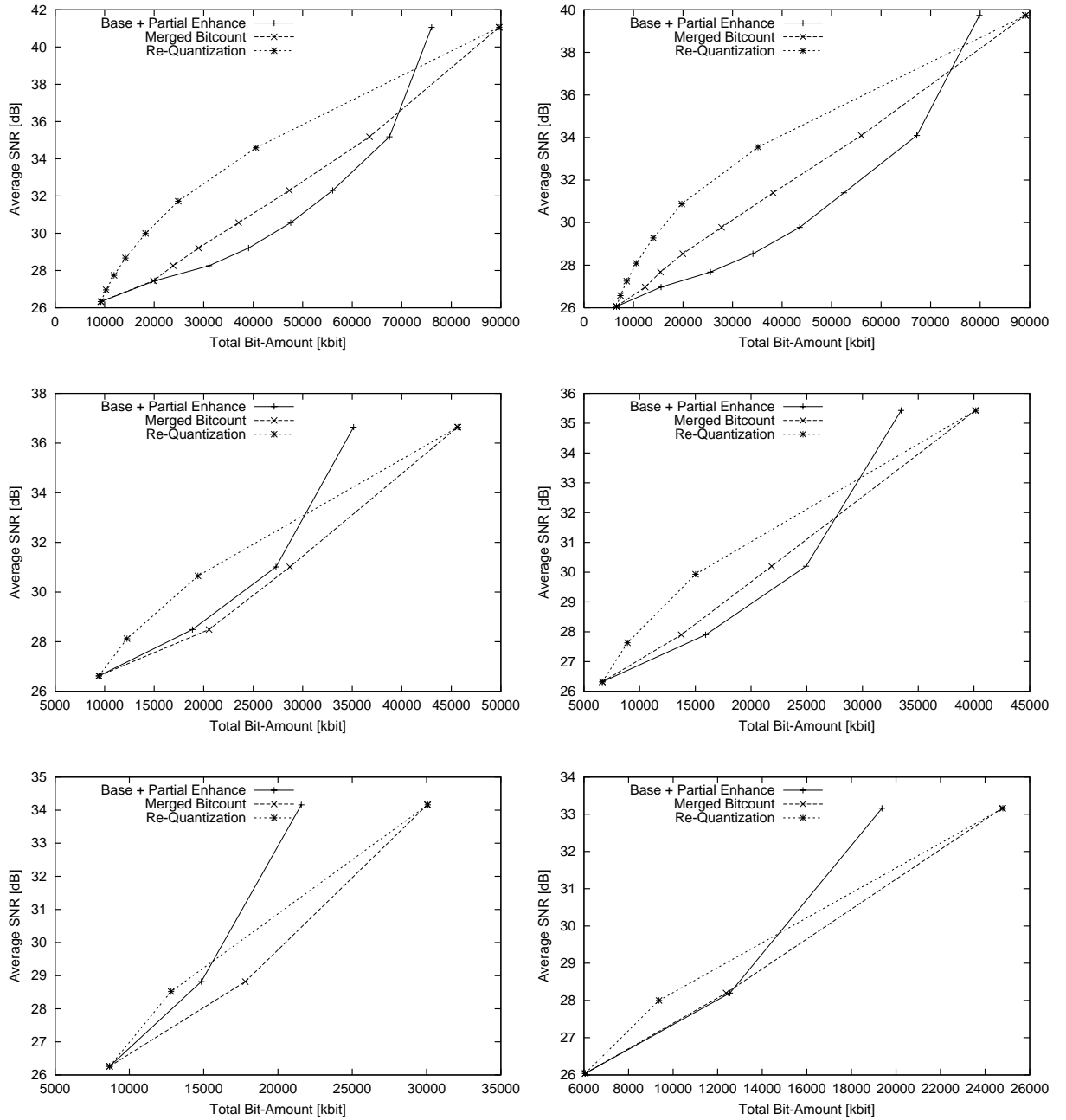


図 4-9: 高位階層数 m に対する階層符号化情報総符号量特性 (上) $MQ_1 = 2$, (中) $MQ_1 = 4$, (下) $MQ_1 = 6$



(a) Bus

(b) Table Tennis

図 4-10: 中間合成ビットストリーム符号化特性, (上) $MQ_1 = 2$, (中) $MQ_1 = 4$, (下) $MQ_1 = 6$

号化効率良く、逆に系列長が短い場合には End_of_Runlength のオーバーヘッドにより効率は低下する。このとき、低次の部分階層高位階層はレベル値が大きい係数で構成されるが1つの系列あたりのイベント数が少ないので、レベル値の自動計算の効果が薄いと考えられる。

ランレングス値の可変長符号化テーブルの傾向と併せて考察すると、ラン値の符号化表は低いランほど短い符号語が割り当てられる傾向を持つ。このため、 $level = m, m - 1$ などの低次の部分高位階層では、大きなラン値が比較的多く発生し、さらに End_of_Runlength によるオーバーヘッドも無視できなくなるために、1階層の部分高位階層を合成時に多くの符号量が必要となると考えられる。

4.6 まとめ

本章では、第3章で提案した2階層への階層符号変換方式を拡張し、高位階層符号化情報に階層構造を持たせることで、高位階層情報を部分的に抽出/合成が可能な、多階層/多品質対応型の階層符号への変換方式を提案した。まず、2種類のステップ値で再量子化された再量子化出力ブロック間での有意係数の分布形態の違いについて考察した。本考察に基づき、高位階層符号として符号化する再量子化前後間の差分量子化係数信号を、係数レベル値ごとに別々にランレングス符号化することで、1つのランレングス系列が1階層に対応する形で高位階層情報に階層構造を実現する方式を提案した。また、高位階層のランレングス情報の一部分のみを基本階層と合成し、さらに合成出力信号と残りの高位階層を再帰的に合成可能であり、高位階層信号を積み上げて変換前の符号化情報を再構築を行う手法を説明した。

評価実験により、提案方式によって高位階層の階層数を細分化してもそれに伴う符号化効率の低下は僅かであり、2階層方式とほぼ同等の符号化性能を有することを示した。また、高位階層を部分的に合成した部分合成映像の品質について評価から、部分合成符号の符号量は、入力非階層符号化情報を再量子化によって同等の品質を生成した場合と比較して大きく増大することを述べた。この部分合成段階での符号化性能の向上は今後の課題である。

本章で提案した階層符号変換方式は、MPEG-4 FGSと同種の機能を汎用の符号化器で実現可能な点を特徴とする。係数レベル値により階層化されたMBの情報構造により、MBごとに階層を選択することで任意の品質をもった合成ビットストリームを生成可能であり、合成時の品質選択性に高い自由度を提供する。提案方式は汎用性と柔軟性を有し同時に十分な符号化効率を実現可能であることから、階層符号化方式の新たな手法として有効な解となり得る方式と考えられる。今後の課題としては、中間合成時の画質改善の他に、高位階層の伝送ビットレートが与えられたときの各MBでの階層情報の選択制御、提案方式の符号化性能の理論的解析などが考えられる。

第5章

階層間独立性実現のための階層符号変換方式

5.1 はじめに

本章では、高位階層符号化情報に階層構造を有する多階層符号化情報構造において、高位階層間での依存関係を排除して高位階層符号化情報間では互いに独立な情報構造を有することで、基本階層符号に任意の高位階層符号を合成して意味のある映像シーンを再生可能な新たな多階層型符号化情報構造について検討する。

5.2 階層間非依存な多階層符号化方式

階層符号化を応用した映像伝送システムとして、マルチキャスト映像配信に階層構造を有する符号化情報を利用し、利用可能帯域に応じた階層の配分/選択により品質制御を行う階層化マルチキャスト映像配信方式が提案されている [79] [14]。また、[80] ではピクチャタイプと DCT 係数の周波数成分を用いて階層情報を構成したデータパーティショニングを拡張した階層符号化方式とこれを利用した映像伝送システムに関する検討が行われている。これらの検討においては、第 n 番階層は第 $n - 1$ 番階層に対する付加情報であることを前提としている。そこで、第 n 番階層相当の品質の取得のためには、基本階層から第 n 階層の全ての符号化情報が取得されていなければならない。中間の $m (< n)$ 階層が欠落していた場合には m 階層以上の符号化情報は無効となる。このように、各階層がそれ以下の階層に依存する情報構造を有する場合、下位側の階層は伝送時のデータ損失を回避するために上位階層と比較して強力なエラー対策機構を必要とする。したがって、階層数が多いほど品質選択の自由度は向上するが、それに伴い各階層の符号化情報の伝送において複雑な優先制御が必要となる。

この制約は第 4 章で提案する階層化方式においても同様に成立し、受信側に第 n 番階層までを合成したビットストリームが蓄積されている場合には、送信側では $n + 1$ 階層以上の符号化信号を抽出して伝送済の部分階層符号の管理する仕組みが要求される。ここで、高位階層符号化信号を互いに独立な情報構造を有していれば、階層数の増加に対して優先制御は基本階層と高位階層の 2 種類のみを考慮すればよく、優先制御や高位階層符号の伝送管理の簡略化が期待される。

そこで、本研究では、 N 階層 ($N > 2$) の階層構造を有する符号化方式において、基本階

層の差分信号である各高位階層信号を互いに独立な情報構造を有するビットストリームとして符号化することで、階層間の依存関係を回避可能し、任意の高位階層符号を基本階層と合成して有効な映像情報の取得が可能な階層符号化方式について検討する。

5.3 独立情報構造を実現する量子化係数差分信号の符号化方式

高位階層符号化情報間の独立性を実現するための符号化方式について述べる。DCT 係数領域内の再量子化演算により基本階層を生成し、再量子化前後の量子化係数領域での差分信号 (量子化係数差分信号) を高位階層として符号化する点は第3章、第4章と同様である。本節では、5.2 で述べた要件を満たすための量子化係数差分信号の符号化方法を説明する。

5.3.1 階層符号への変換方法

階層符号への変換方式の概念を図5-1に示す。本方式は方式は Step1, Step2 の2つのステップで構成される。Step1 では、入力ビットストリームを復号して取得した入力係数を基本階層と高位階層へ分離する。Step2 では、高位階層へ符号化する量子化係数列を N 通りの部分系列へ分解し、それぞれを高位階層符号として符号化する。詳細を以下に述べる。

Step 1 : 基本階層と高位階層への分離

入力ビットストリームの DCT 係数符号を復号した2次元ランレベル系列を量子化係数列へ復元した入力量子化係数ベクトルを X として式 (5.1) に定義する。

$$X = \{x(0), x(1), \dots, x(63)\} \quad (5.1)$$

X の各要素を再量子化処理して生成した再量子化出力係数ベクトル B を式 (5.2) に定義する。ただし、再量子化演算式は第3章と同様である。

$$B = \{b(0), b(1), \dots, b(63)\} \quad (5.2)$$

ただし、 X および B の各要素はジグザグスキャンの順番で並ぶものとする。

X の中から $b(i) = 0$ に対応した係数値のみを抽出した量子化係数列を C とする。 C の生成方法は以下の手順に従う。

1. $b(i)$ を読み出し、 $b(i) = 0$ の場合は2.へ、 $b(i) \neq 0$ の場合は3.へそれぞれ進む。
2. $c(j) = x(i)$ として C 内の j 番要素に $x(i)$ を代入し、 j を1だけ増加。
3. B のカウンタ i を1だけ進める。 $i > 63$ ならば終了。そうでなければ1.へ戻る。

B の符号化方法は第3章と同様であり、標準方式に準拠したランとレベルによる2次元ランレングス方式に基づくエントロピ符号化処理を行い、基本階層符号化情報を入力する。提案方式は、 $b(i) = 0$ に対応する入力係数値 $x(i)$ の符号化方法を置き換えたものであり、

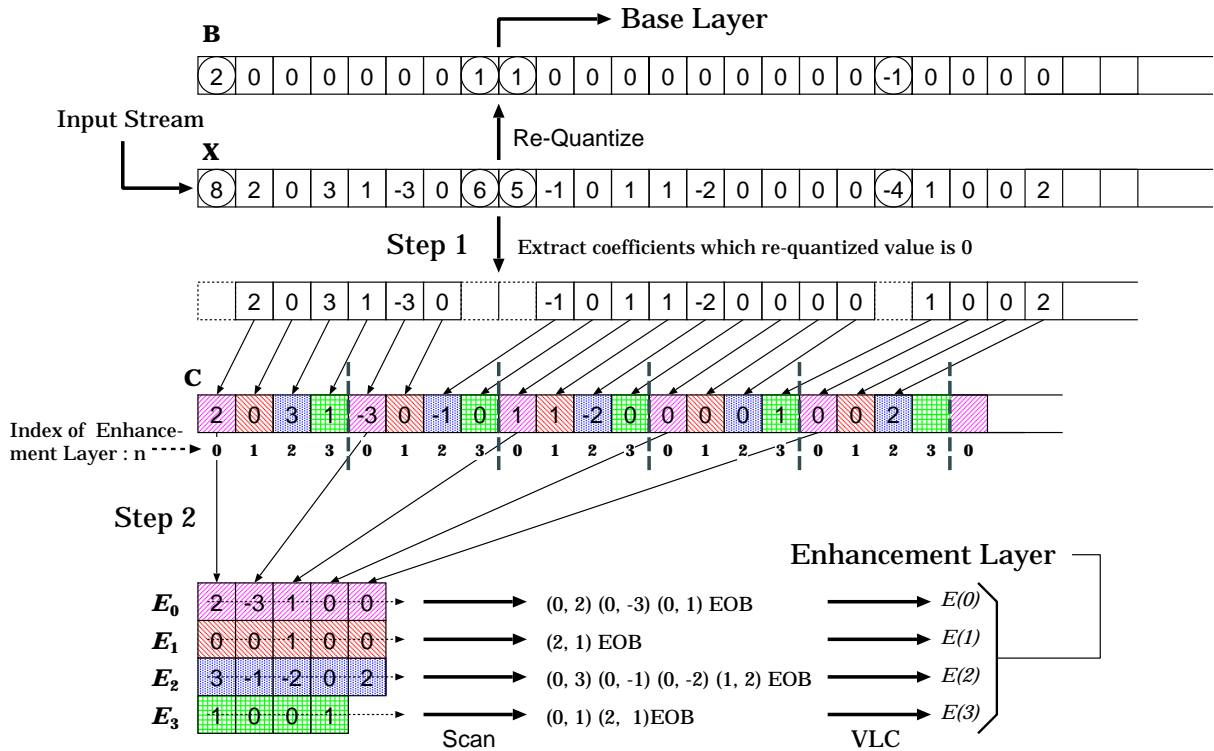


図 5-1: 独立性を有する階層符号への変換方法

$b(i) \neq 0$ に対する $x(i)$ との差分信号の符号化方法についても第 3 章と同様の方式とする。すなわち、 $b(i)$ を入力および出力量子化パラメータの比率 M_{Q1}/M_{Q2} を乗じた値と $x(i)$ との差分値を符号化する。そこで、以下では C の符号化方法のみを説明の対象とする。

Step 2 : 高位階層の符号化处理

Step 2 では C を N 通りの部分量子化係数列へ分解し、それぞれの部分係数列 (以下では、高位階層量子化係数列と呼ぶ) に対してエントロピ符号化处理を行い N 通りの高位階層符号を出力する。

C を分解して生成された第 n 番高位階層量子化係数列を E_n ($0 \leq n < N$) とおく。Step 1 で生成された C において、 C の各要素 $c(j)$ を N 個ごとに区切った区間において、各区間内の n 番目に位置する係数值を E_n へ出力する。そこで、 C から E_0, E_1, \dots, E_{N-1} への分解は式 (5.3) により行われる。

$$\left\{ \begin{array}{l} E_0 = \{ c(0), \quad c(N), \quad c(2N), \quad \dots \quad c(kN), \quad \dots \} \\ E_1 = \{ c(1), \quad c(N+1), \quad c(2N+1), \quad \dots \quad c(kN+1), \quad \dots \} \\ \vdots \\ E_n = \{ c(n), \quad c(N+n), \quad c(2N+n), \quad \dots \quad c(kN+n), \quad \dots \} \\ \vdots \\ E_{N-1} = \{ c(N-1), \quad c(N+N-1), \quad c(2N+N-1), \quad \dots \quad c(kN+N-1), \quad \dots \} \end{array} \right. \quad (5.3)$$

E_n の k 番目の要素を $e_n(k)$ とすると, $e_n(k)$ は式 (5.4) となる.

$$e_n(k) = c(kN + n) \quad (5.4)$$

したがって, 高位階層数 N を用いて n, k と j は一意に関係づけられ式 (5.5) の通りとなる.

$$\begin{cases} n = j \bmod N \\ k = \left\lfloor \frac{j}{N} \right\rfloor \end{cases} \quad (5.5)$$

ただし, $[a]$ は a を越えない最大の整数を, $a \bmod p$ は a を p で除算した余りを意味する.

式 (5.3) で生成された E_0, \dots, E_{N-1} について, それぞれをスキャンしてランとレベルによるランレングス表現へ変換し, その後エントロピ符号化処理により各高位階層符号化情報を出力する.

5.3.2 非階層符号化信号への合成方法

基本階層符号と N 通りの高位階層符号から非階層符号化情報を合成する方法を述べる. 図 5-2 に本処理の概念を示す.

B と $E_n (n = 0, 1, \dots, N-1)$ から入力量子化係数列 X への復元方法は Step1, Step2 の 2 つのステップで構成する. Step 1 では, E_0, \dots, E_{N-1} を合成して C を復元する. Step 2 では, B と C を合成して X を復元する. 詳細を以下に述べる.

Step 1 : 高位階層係数列の合成処理

高位階層量子化係数列 $E_n (n = 0, 1, \dots, N-1)$ を式 (5.6) に表す.

$$\begin{cases} E_0 = \{ e_0(0), & e_0(1), & \dots & e_0(k), & \dots \} \\ E_1 = \{ e_1(0), & e_1(1), & \dots & e_1(k), & \dots \} \\ \vdots & & & & \\ E_n = \{ e_n(0), & e_n(1), & \dots & e_n(k), & \dots \} \\ \vdots & & & & \\ E_{N-1} = \{ e_{N-1}(0), & e_{N-1}(1), & \dots & e_{N-1}(k), & \dots \} \end{cases} \quad (5.6)$$

E_n それぞれに関して, 式 (5.7) のように G_n なる量子化係数列を生成する.

$$\begin{cases} G_0 = \{ \underbrace{e_0(0), 0, 0, \dots}_N, & \underbrace{e_0(1), 0, 0, \dots}_N, & \underbrace{e_0(2), 0, 0, \dots}_N, & \dots \} \\ G_1 = \{ \underbrace{0, e_1(0), 0, \dots}_N, & \underbrace{0, e_1(1), 0, \dots}_N, & \underbrace{0, e_1(2), 0, \dots}_N, & \dots \} \\ \vdots & & & \\ G_{N-1} = \{ \underbrace{0, 0, \dots, e_{N-1}(0)}_N, & \underbrace{0, 0, \dots, e_{N-1}(1)}_N, & \underbrace{0, 0, \dots, e_{N-1}(2)}_N, & \dots \} \end{cases} \quad (5.7)$$

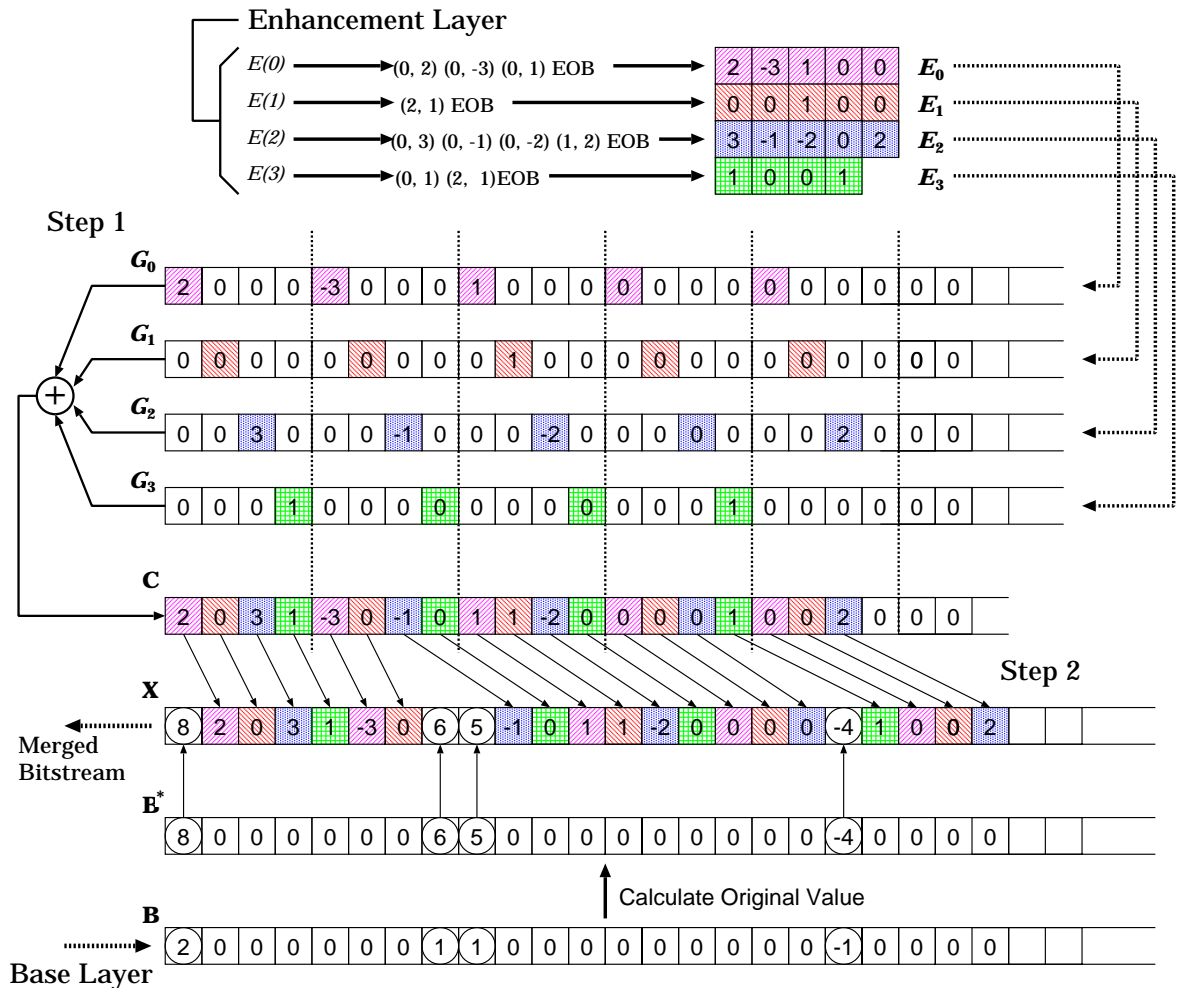


図 5-2: 合成方法

G_n の j 番目の要素を $g_n(j)$ と表すと, $g_n(j)$ は式 (5.8) のように表される .

$$g_n(j) = \begin{cases} e_n(k) & \text{for } j = kN + n \\ 0 & \text{otherwise} \end{cases} \quad (5.8)$$

G_0, \dots, G_{N-1} を用いて式 (5.9) より, E_0, \dots, E_{N-1} の合成量子化係数 C を復元する .

$$C = \sum_{n=0}^{N-1} G_n \quad (5.9)$$

Step 2 : 基本階層と高位階層の合成処理

Step 2 では, Step 1 で合成した C と基本階層を復号して取得した再量子化出力係数ベクトル B を合成して入力量子化係数ベクトル X を復元する .

まず, 再量子化前後の量子化パラメータ MQ_1, MQ_2 および $x(i)$ との差分信号を用いて, B の各有意係数を再量子化前の値へ復元する . 本処理は第 3 章と同様である . 本処理によって B 内の有意係数を再量子化前の値へ復元して生成した量子化係数ベクトルを B^* , B^* の i 番目の要素値を $b^*(i)$ とそれぞれ定義する .

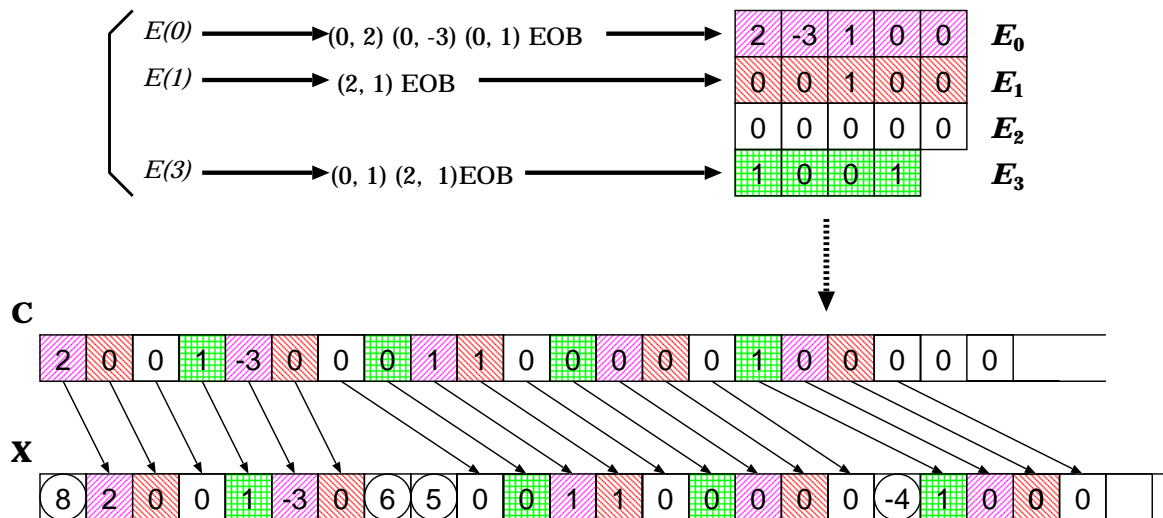


図 5-3: 高位階層損失時の合成処理

入力量子化係数ベクトル X の生成方法は以下の手順により行う。

1. $b^*(i)$ を読み出し, $b(i) = 0$ の場合は 2. へ, $b(i) \neq 0$ の場合は 3. へそれぞれ進む。
2. $x(i) = c(j)$ として X の i 番要素に $c(j)$ を代入し, j を 1 だけ増加。4. へ進む。
3. $x(i) = b^*(j)$ として X の i 番要素に $b^*(j)$ を代入。4. へ進む。
4. カウンタ i を 1 だけ増加。 $i > 63$ ならば終了。そうでなければ 1. へ戻る。

以上の手順により復元された入力量子化係数ベクトルをスキャンしてランとレベルによる 2次元ハフマン符号化処理を行い, 変換前の非階層ビットストリームを符号化出力する。

ここで, 式 (5.8) より E_n を構成する各量子化係数の C 内のインデックス値は N と n によって決定される。そこで, N と n を高位階層符号のヘッダ情報に符号化することで, 各高位階層は互いの復号量子化係数を参照せずにインデックス値を特定可能となり, 階層間の独立性が実現される。そのため, 任意の高位階層符号を損失した場合でも, 図 5-3 のように損失高位階層の係数ベクトルを $E_n=0$ において合成処理を適用することで, 各係数の位置構造を失うことなく残りの高位階層からの合成映像を取得が可能となる。

5.4 評価実験

提案方式に関するシミュレーション実験を行い, 総符号量と画質の特性について評価する。実験用ビットストリームの符号化条件を表 5-1 に示す。また, 基本階層の符号化レートは $4[\text{Mpsbit}/\text{sec}]$ とする。

5.4.1 階層数に対する総符号量特性

階層符号変換時の総符号量の特性を評価する。可変長符号化テーブルには MPEG-2 標準で規定されるものを適用する。

表 5-1: 実験用ビットストリーム符号化条件

符号化方式	MP@ML(N=15, M=3)
符号化レート	10, 15 [Mbps]
符号量制御方式	TM5 方式
動き補償, DCT 方式	フレーム動き補償, フレーム DCT (フィールドモード OFF)
テストシーケンス	Bus, Table Tennis
画像フォーマット	ITU-R BT.601 4:2:0 Format 輝度信号 704 pel × 480 line 色差信号 352 pel × 240 line
符号化フレーム	150 [Frame]

高位階層数 N に対する階層符号変換後の総符号量の特性を図 5-4 に示す。

図 5-4 より, 第 3 章, 第 4 章の方式と比較して総符号量は増大するが, $N = 10$ の場合でも総符号量の増加率は 1.25 倍程度であり, 階層間の依存構造の排除と階層数の増加に伴う大幅な符号化効率の低下は回避可能である。提案方式では, 高位階層符号のランレングス値の値域は X に比べて大幅に減少する。入力係数ベクトル X においてはランレングスの最大値は 63 であり, 0 から 63 までの 64 通りのシンボルを考慮する必要がある。一方, 提案方式では C を N 通りへ分解して E_n を生成しており, E_n 内のランレングス値は最大でも $\lfloor 64/N \rfloor$ である。このため, N が大きいほど短いランレングス値が集中することで, 1 個の有意係数あたりの平均符号長は減少し, 総符号量を抑制させる効果が働く。

第 4 章では, 多階層化に伴う効率低下を回避し入力符号量よりも少ない総符号量を実現可能な階層化方式を提案した。それに対して本提案方式では, 階層符号変換後の総符号量は N に比例して増加する傾向にある。これは, 高位階層を独立に符号化したことによる効率の低下と考えられる。提案方式では, N 本の高位階層符号を出力するので N とともに符号化対象の係数ブロックの総数は増加し, 各ブロックにおける係数の終了符号に要する符号が増加する。また, MB の符号化モードに必要な符号量も加算される。したがって, 総符号量の増大は, これらのオーバーヘッド符号に起因すると考えられる。本実験では標準方式で規定される符号化テーブルを用いた評価に留まっており, 符号化効率の最適化は施されていない。今後は, ランレングスの確率分布の定量的評価とそれに基づく符号化テーブルの最適設計により, N に対する総符号量の増加率を抑制して符号化効率を改善することが課題となる。

5.4.2 合成階層数と画質の関係

本実験では, 高位階層数を $N = 4$ に固定し基本階層に対して高位階層を 1 階層ずつ合成した場合の, 合成出力総符号量と画質の関係について考察する。提案方式では, 任意の順序で各高位階層符号を合成可能であるので, 表 5-2 に示す 4 通りの場合について比較する。

実験結果を図 5-5, 図 5-6 に示す。ここで, 基本階層のビットレートは 4[Mbit/sec] であり, 4[Mbit/sec] 時のプロットは基本階層のレート歪み特性を表す。図 5-5, 図 5-6 から, 任

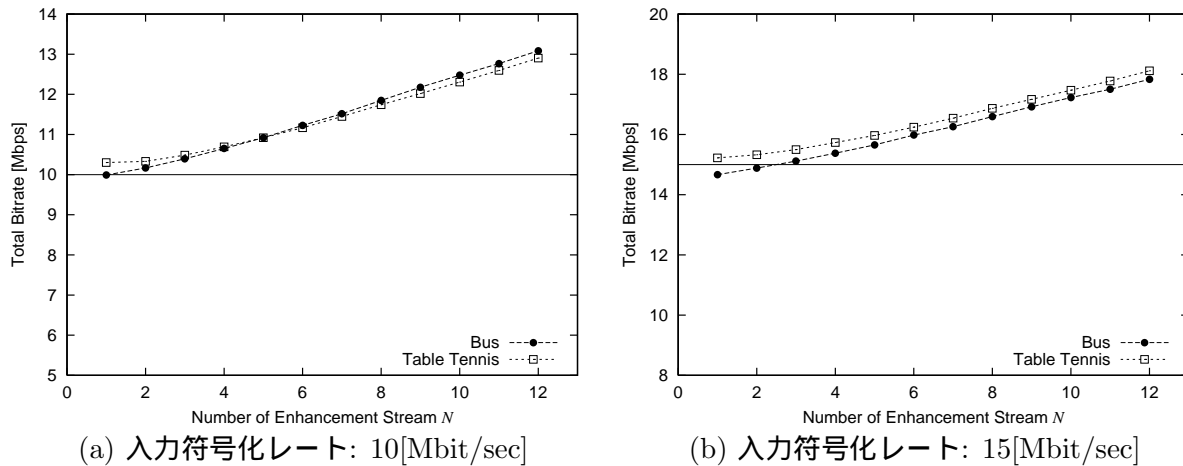
図 5-4: 高位階層数 N に対する階層符号総符号量の特性

表 5-2: 高位階層符号の合成順序

	高位階層合成順番			
	1 番目	2 番目	3 番目	4 番目
Case 1	E_0	E_1	E_2	E_3
Case 2	E_1	E_2	E_3	E_0
Case 2	E_2	E_0	E_1	E_3
Case 4	E_3	E_2	E_1	E_0

意の基本階層と高位階層の組み合わせで合成符号化情報の取得が可能であり、品質および符号量は高位階層の選択に依存しないことが確認される。これは、いかなる高位階層を選択しても同等の品質を取得可能であることを意味しており、提案方式によって階層選択に自由度を持たせた高い柔軟性を有する階層符号化機能が実現された。

画質の変化特性は 4.5.3 と同様の傾向にあり、中間合成段階での画質への課題があること。これは、階層間の依存構造を回避のために、各係数の周波数や信号レベルに基づく階層化ではない点に起因した考えられる。提案方式では、中間段階において通常の符号化処理器内の量子化演算で発生しないパターンが発生し、結果として係数情報の符号が増加したと考えられる。そのため、中間合成段階における符号化効率の改善は今後の課題である。

5.5 提案符号化方式の応用例

5.5.1 複数デバイスの利用による耐性の向上

複数の蓄積デバイス (HDD) を利用した記録方式として RAID が知られている。この中で、同一の情報内容を複数の HDD に複製して記録することで、HDD の破損に対する耐性を向上させるミラーリング記録方式が提案されている。本方式は、RAID1 として知られており、1 台のディスクが破損しても残りのディスクから記録情報を取り出すことが可能であり、冗

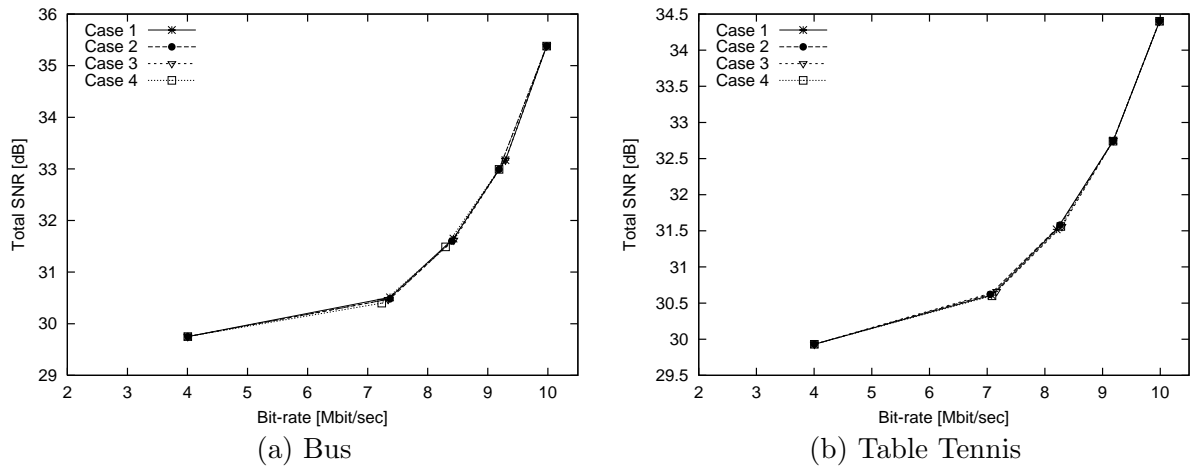


図 5-5: 中間合成出力符号量と画質の関係 (入力符号化レート 10[Mbit/sec])

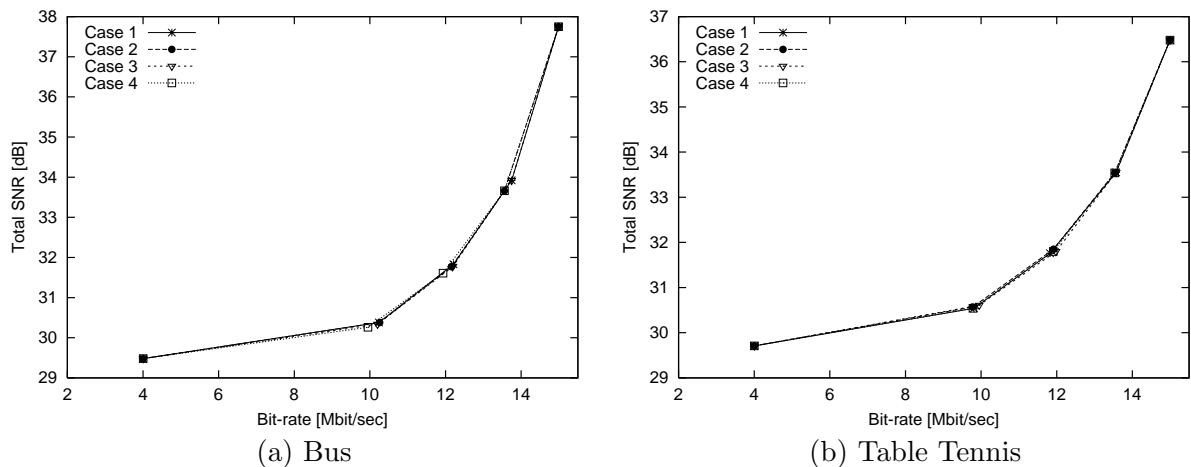


図 5-6: 中間合成出力符号量と画質の関係 (入力符号化レート 15[Mbit/sec])

長化により破損に対する耐性を向上させるものである。

伝送媒体でも同様の考え方を適用可能である。1つの符号化情報を N 本の伝送路により伝送することで、ある伝送路が寸断されても残りが有効である限り映像伝送は継続可能となる。1本の伝送路が切断される確率を p ($0 \leq p \leq 1$) とすると、 N 本へ冗長化することで映像伝送が完全に寸断される確率は p^N に減少し、伝送路の切断に対する耐性が向上する。

5.5.2 高位階層信号成分の分割符号化による伝送路の有効利用

N 本の伝送路の利用において、符号化ビットストリームを N 本の伝送路に複製して伝送すると、 N 倍の伝送帯域が必要となるので、冗長化による耐性の向上のトレードオフとして、伝送路の利用効率が低下する。

映像信号の伝送に特化した場合、複数伝送路を利用するときに符号化映像信号の特徴を考慮することで伝送帯域の効率利用が可能である。映像信号の特徴として、各種ヘッダ情報や動きベクトル、低周波 DCT 係数信号等の再生時に重要となる信号成分以外の大部分を占め

る高周波 DCT 係数信号は、全ての符号化信号を復号できなくても意味のある映像シーンを再生可能である点に着目する。そこで、複数の伝送路上へ複製する信号はヘッダ情報や低周波信号成分などの映像再生時に最低限必要となる情報成分のみに留めて、それ以外の高周波 DCT 係数信号を複数の符号化情報へ分離して符号化して各符号化情報を別々の伝送路上へ分割伝送する。これにより、一部分の情報成分のみを冗長化するので 1 本の伝送路の伝送帯域は低減され、複数の伝送路利用による総利用帯域の増加を回避する。伝送路が切断された場合には、切断された伝送路上の高周波信号成分は欠落するが、残りの情報成分が欠落信号に依存しなければ、欠落信号成分相当の品質劣化はあるが映像再生は継続可能となる。

5.6 まとめ

本章では、 N 階層の階層構造を有する符号化方式において、複数の高位階層を互いに独立した情報構造として符号化することで、階層間の依存関係を排した階層符号化方式を提案した。第 3 章、第 4 章と同様の非階層符号化情報に対する符号変換処理をベースとして、基本階層は従来と同様の DCT 係数領域内再量子化方式により符号化し、高位階層は、量子化係数列の位置インデックス値を高位階層の数で除算した剰余値により出力先の高位階層番号を決定して、それぞれを独立にスキャン、エントロピ符号化処理をすることで、独立な情報構造を有する階層符号化を実現する方式を提案した。

評価実験から、提案方式により階層化に伴う大幅な総符号量は回避可能であることを確認し、符号化効率の有効性を示した。また、提案方式を応用による効率性と損失耐性の両立した映像蓄積、伝送方式への実現可能性を言及し、実用性がある方式であることを述べた。

第6章

輻輳制御の影響を利用したトランスコード変換 レート動的制御方式

6.1 はじめに

本章では、伝送帯域に適したビットレートへの変換を行うトランスコードにおいて、帯域非補償型の伝送路環境に置いて、変動する帯域に応じて適切な変換レートの制御を行うためのトランスコード変換レート動的制御方式について研究する。TCP/IP 上での映像伝送を想定し、TCP/IP の輻輳制御による影響を制御に利用することで、プロトコルに特別な実装をせずに変換レートの動的制御を実現する方式について検討する。

6.2 TCP/IP 輻輳制御による影響を利用した変換レート動的制御

TCP/IP 等の輻輳制御を有する伝送チャネルでは、送信情報を確実に伝送することを保証する目的で、伝送能力を越えた情報量が発生した場合には伝送路上での情報の欠落を回避のための伝送速度制御（輻輳制御）が働く。これにより、伝送路上の混雑状況やルータ/受信端末上の受信バッファの状態に応じた送信データの転送速度の制御（フロー制御）とパケット再送制御が働き、転送される情報量は伝送能力による制限を受ける。

映像伝送ネットワークシステムのルータやゲートウェイなどの中継ノードにトランスコードを適用した場合、トランスコードの出力がネットワークのインタフェース部分に接続される実装形態となる。ここで、上記の輻輳制御に起因した転送速度の制限が働くとインタフェース部分で送信待ち時間が発生し、この待ち時間が前段に位置するトランスコードへ波及すると、結果として、トランスコード自体の処理能力がインタフェース部分の動作による影響を受けると考えられる。

入力ストリームのビットレートが $R_{in}[\text{bit/sec}]$ として与えられた場合、 $R_{in}[\text{bit}]$ のビットストリーム素片は入力ストリームの 1 秒分の映像内容に相当する。トランスコード自体は十分に高速に動作している仮定すると、 $R_{in}[\text{bit}]$ の符号化情報の処理に 1 秒以上の所要時間を要した場合にはネットワークインタフェース領域でのデータ送出部分にボトルネックがあると考えられる。TCP/IP では転送速度が十分でない場合には送信待ち時間が発生することから、この処理速度の低下要因はこの輻輳制御に起因した送信待ち制御と推定される。

そこで、TCP/IP プロトコルの振舞に起因した処理能力の低下現象を入力ストリーム内の時刻情報を基準にして評価することで帯域変動の推定を行い、それに基づいてトランスコードの変換レートを制御する動的レート制御方式について検討する。ストリーム内の時刻情報は、MPEG-1 システムストリームおよび MPEG-2 PS に準拠する多重化ストリームの場合には、各パックヘッダに符号化されている SCR 符号が基準にして算出可能である。入力ストリームとして MPEG-1 システムストリーム、MPEG-2 PS を対象として、SCR 符号の復号時点でのトランスコード処理の実際の経過時間を観測し、これに SCR 復号値を基準として評価することで、伝送帯域の推定と変換レートを制御する。

6.3 基準時刻情報に基づく帯域推定に関する基礎実験

本節では、TCP/IP で接続された伝送路上でトランスコードを用いたストリーム伝送の基礎実験を行い、輻輳制御によるトランスコードに対する影響について検証する。

6.3.1 想定する伝送モデル

TCP/IP で接続された映像送出系において、伝送速度を R_N [bit/sec] とおき、トランスコードは十分高速に動作し伝送路とのインタフェース部分の処理能力は伝送速度に等しいと仮定する。この場合、送出系全体では R_N [bit/sec] の速度でビットストリームを出力する。

ビットストリームのビットレートを R_T とすると、 $\Delta\tau$ [sec] 相当分のビットストリーム素片の送出処理を完了した時点では $R_T \cdot \Delta\tau$ [bit] の符号が出力される。実際のストリーム出力速度が伝送速度 R_N に等しいと仮定すると、 $R_T \cdot \Delta\tau$ [bit] の符号量の処理に要した時間が Δt [sec] として観測されたときの伝送速度 R_N は式 (6.1) として求められる。

$$R_N = \frac{R_T \cdot \Delta\tau}{\Delta t} \quad (6.1)$$

ここで、ビットストリームを先頭ビットから 1 [byte](=8 [bit]) 単位で区切った符号化データをバイトデータと呼び、バイトデータが先頭から i [byte] 目に位置するとき、 i を本ストリームのバイト位置と定義する。 i なるバイト位置にあるバイトデータがストリーム中の $\tau(i)$ [sec] なる時刻を表すときには、 $\tau(i)$ は式 (6.2) のように表される。

$$\tau(i) = \frac{SCR(i')}{f} + \frac{i - i'}{R_{in}} \quad (6.2)$$

i' はストリーム中の基準時刻を与える SCR(System Clock Reference) 符号が符号化されているバイト位置であり*、 $SCR(i')$ をこのときの SCR 符号の復号値とする。また、 f はシステムターゲット復号器の動作周波数(システムクロック周波数)を表し MPEG-1 システム規格では $f=90000$ [Hz]、MPEG-2 システム規格では $f=27000000$ [Hz] としてそれぞれ規定される。

*厳密には、パックヘッダ中の system_clock_reference_base が符号化されているビットフィールドの最終ビットを含むバイト位置として定義される。

$\Delta\tau$ [sec] なる時間相当分のビットストリーム素片の先頭, 末尾のバイト位置をそれぞれ i_1, i_2 とおく. i_1, i_2 が SCR 情報が符号化されているバイト位置であれば, $\Delta\tau$ は式 (6.3) のように表される.

$$\Delta\tau = \frac{SCR(i_1) - SCR(i_2)}{f} \quad (6.3)$$

式 (6.1), 式 (6.3) より式 (6.4) を得る.

$$\frac{R_N}{R_T} = \frac{SCR(i_1) - SCR(i_2)}{\Delta t \cdot f} \quad (6.4)$$

ストリーム中の i_1, i_2 なるバイト位置に SCR が符号化されておりそれぞれの SCR 復号値を $SCR(i_1), SCR(i_2)$ としたときに, 式 (6.4) より, 伝送速度 R_N とトランスコード出力ビットレート R_T の比率 R_N/R_T は, i_1, i_2 なるバイト位置にある任意の 2 つの SCR 復号値と, その間のバイトデータの処理に要した時間から推定可能となる. したがって, 式 (6.4) より伝送路の伝送速度は, i_1 から i_2 なる区間の入力ビットストリーム素片のトランスコード処理に要した時間と, 本区間での SCR 復号値の増分量から推定可能と考えられる.

そこで, トランスコードを用いてレート変換を行いながら TCP/IP で接続された伝送路上でストリーム伝送実験を行い, 各入力パックヘッダの SCR 復号値とトランスコード処理開始からの経過時間との関係を測定することにより, TCP/IP 伝送路の伝送速度に基因したトランスコードの処理速度への影響を実験的に検証する.

6.3.2 実験条件

(1) 実験系の構成

評価実験のための映像伝送系を図 6-1 に示す. 符号化映像の配信源 (サーバ) と受信端末 (クライアント) 間を TCP/IP 上の伝送路で接続する. サーバ上にトランスコードを実装し, 伝送路帯域に応じてビットレートスケールリングを行いながら TCP/IP で伝送する. このときに, 受信側で受信速度の制限をかけることで, 擬似的に伝送帯域の制御を実現する. サーバとクライアントの各機能は PC 上のソフトウェアとして実装する. 実験用 PC の諸元は, サーバ用とクライアント用で同一であり表 6-1 に示す.

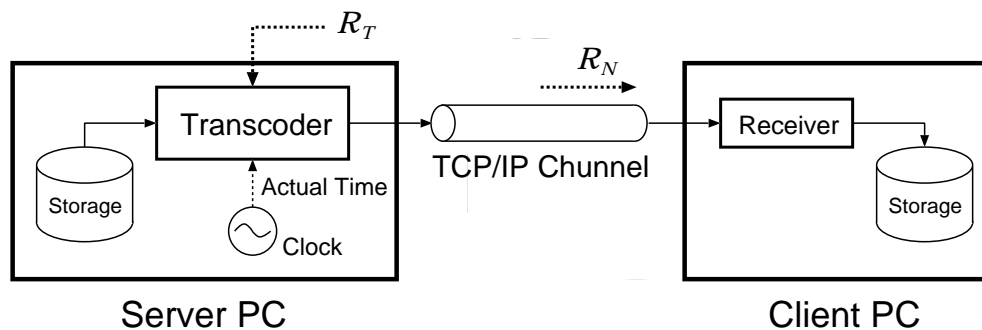


図 6-1: 実験系の構成

表 6-1: 実験用 PC の諸元

CPU	Pentium III 933[MHz]
メモリ	SDRAM 256[MByte]
NIC	3Com 製 100BASE-TX Ethernet Card(PCI 接続)
OS	Vine Linux 2.5 CR (kernel 2.2.20)

表 6-2: 実験用ビットストリーム符号化条件

多重化条件	多重化方式	MPEG-1 System
	多重化レート	2000 [kbit/sec]
	PES パケット長	2048 [byte] (固定)
	パック構造	1 パック = 1PES パケット (固定)
	ストリーム時間	約 62 [sec]
ビデオ符号化条件	符号化方式	MPEG-1 Video
	符号化レート	1500 [kbit/sec]
	GOP 構造	N=15, M=3 (固定)
	画像サイズ	4:2:0 SIF Format 輝度信号 352 pel × 240 line 色差信号 176 pel × 120 line
	フレームレート	29.97 [frame/sec]
オーディオ符号化条件	符号化方式	MPEG-1 Audio Layer II
	符号化レート	128 [kbit/sec]

1PES パケット長 = 1 フレームとなるように変換

(2) 実験用ビットストリームの作成

実験用ビットストリームには、市販のハードウェアエンコーダ (Canopus 社製 MVR-D2000) により符号化されたビットストリームの多重化構造を変換した実験用多重化ストリームを使用する。ハードウェアエンコーダにおける符号化条件を表 6-2 に示す。

評価用トランスコーダにおいて、パックヘッダからの SCR 情報の取得とそれに基づくレートの推定を同期させるために、表 6-2 で符号化された多重化ストリームの PES パケット長を、1 パック=1PES パケット=1 ピクチャとなるように変換する。このとき、パックヘッダのバイト位置がピクチャヘッダ直前となるように変更されるので、これに伴って各パックヘッダの SCR を再計算する。

実験用ビットストリームの多重化レートを固定として R_{in} [bit/sec] とおき、ストリーム先頭のパックヘッダの SCR を SCR_0 、本 SCR 情報が符号化されているバイト位置を i_0 と定義する。バイト位置変更後のパックヘッダにおける SCR 情報がストリーム先頭から i 番目

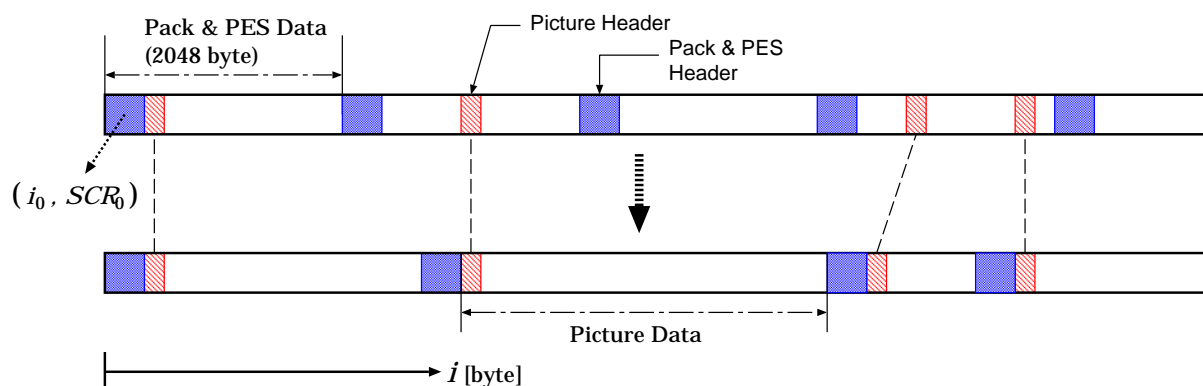


図 6-2: 実験用ビットストリーム作成のための多重化ヘッダ構造の変換

のバイト位置にあるときに，このときの SCR 値 $SCR(i)$ は式 (6-5) として算出される．

$$SCR(i) = \frac{8 \times f}{R_{in}} \times (i - i_0) \quad (6-5)$$

表 6-2 に示すパラメータで符号化された多重化ビットストリームの先頭パックヘッダを探索して SCR_0 を復号し，このときのバイト位置から i_0 を取得する．次に，多重化ストリーム中のパックヘッダ，PES ヘッダ，オーディオ PES パケットを取り除きビデオエレメンタリストリームへ多重分離する．その後，ビデオエレメンタリストリーム中の各ピクチャヘッダの直前にパックヘッダおよび PES ヘッダを挿入する．このときに各ピクチャにおけるストリーム先頭からのバイト位置 i を求めて，式 (6-5) により当該ピクチャ直前のパックヘッダにおける SCR 値を算出する．

6.3.3 実験結果

受信速度固定時における処理経過時間と SCR 復号値の関係

トランスコードの目標変換ビットレート R_T を 1500, 1250, 1000, 750, 500, 250[kbit/sec] に設定し，受信速度の制限を加えることで伝送速度 R_N の制御を行いストリームを伝送したときの，入力ストリームの各パックヘッダ検出時点でのトランスコード処理経過時間 T [sec] と，当該パックヘッダの SCR 復号値との関係を図 6-3 に示す．ただし，図 6-3 に示す SCR 復号値は，システムクロック周波数 f [Hz] で除算して時間 ([sec]) へ換算した値である．

図 6-3 より，SCR 復号値と各 SCR 復号時のトランスコード処理経過時間の関係は比例直線で近似可能であり，変換レート R_T と伝送速度 R_N が一定の場合には両者は互いに一定の比率を保ちながら増加することが分かる．そして， R_T と R_N による違いは直線の傾きとして観測されることが，それぞれの図中のプロットおよび各図同士の比較により確認できる．そこで，この近似直線の傾きを処理経過時間増加率と定義する．

次に，伝送速度 R_N による違いについて見ていくと， R_N が低いほど各 SCR 復号時時点でのトランスコード処理の経過時間は長く，単位時間あたりのトランスコードの処理性能に

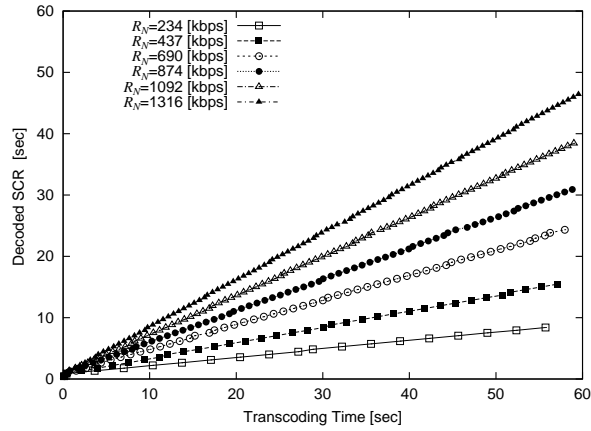
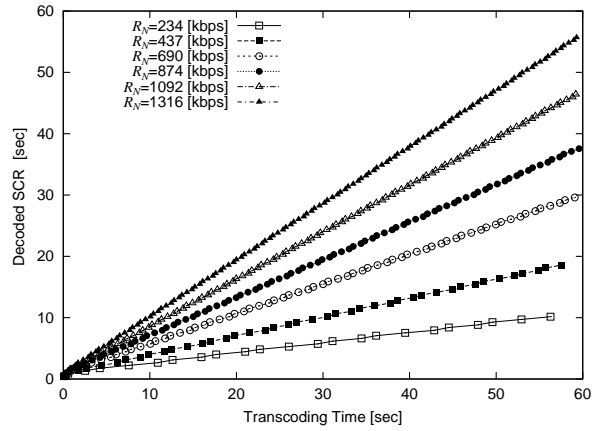
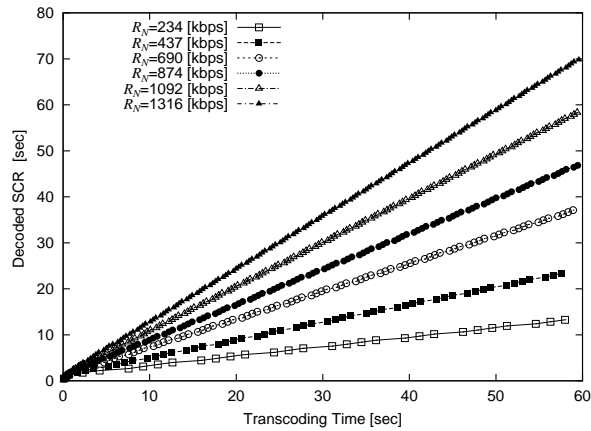
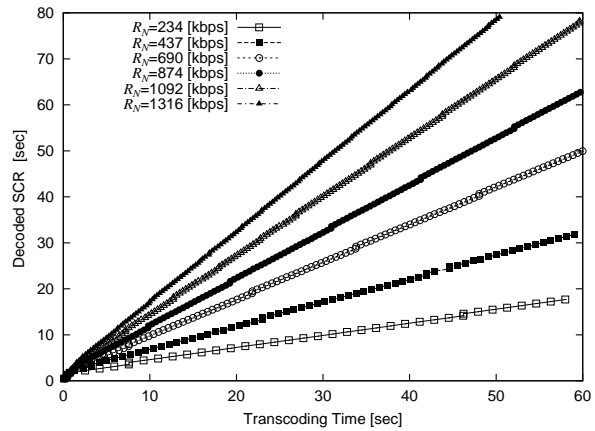
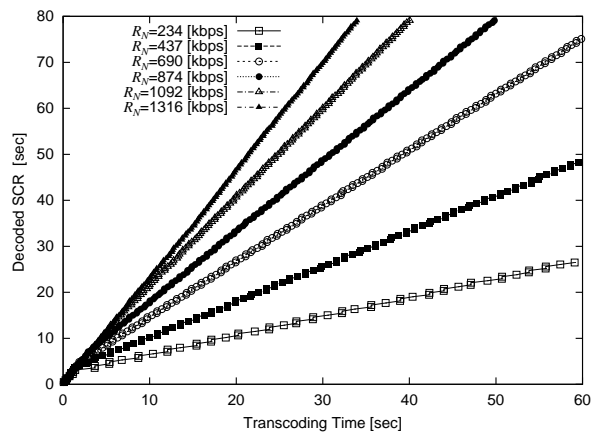
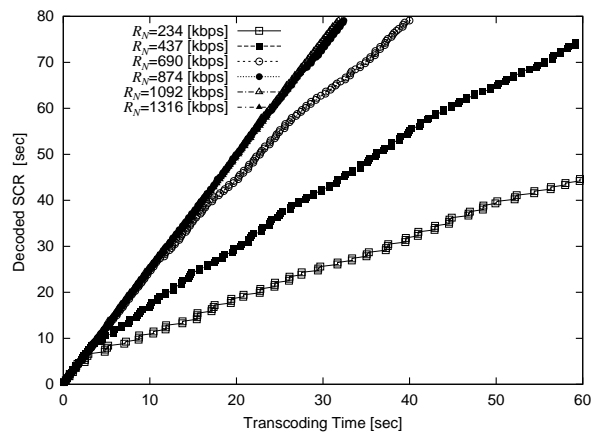
(a) $R_T = 1500$ [kbit/sec](b) $R_T = 1250$ [kbit/sec](c) $R_T = 1000$ [kbit/sec](d) $R_T = 750$ [kbit/sec](e) $R_T = 500$ [kbit/sec](f) $R_T = 250$ [kbit/sec]

図 6-3: トランスコード処理時間と SCR 復号値の関係

影響を与えていることが確認される。 $R_T = 1500$ [kbit/sec] の場合についてトランスコード処理開始から 30[sec] 経過時点で処理中のバイト位置を比較すると、 $R_N = 1316$ [kbit/sec] の場合には 23.83[sec] なる時刻の入力ストリームが処理されるのに対し、 $R_N = 437$ [kbit/sec] では 8.400[sec] となる。 $R_N = 1316$ [kbit/sec] の場合と比較して 15.43[sec] に相当する符号量が未処理であり、伝送速度に起因した処理の遅れが発生している。したがって、伝送能力が低いほど符号化情報の送出处理に多くの時間を要し、結果としてトランスコード処理の経過時間の増加として観測されると考えられる。

また、変換レート R_T の違いに着目すると、 R_T が R_N に対して大きいほど処理経過時間は長くなり、処理の遅れは増加する。これより、経過時間の増加は、伝送路の伝送処理能力に対する伝送符号量として関連性を持つと考えられる。

以上の考察より、各パックヘッダ復号時のトランスコード処理経過時間はトランスコーダの変換レートと伝送速度による影響を受けており、SCR 復号値に対する処理経過時間が伝送路の伝送能力の評価指標となりうる可能性があることが推察される。

トランスコード処理経過時間増加率を Y とおき、 Y と R_T および R_N との関係について考察する。図 6-3 の各グラフの近似直線の傾きにより求めた Y と R_N/R_T の比率 R_N/R_T の関係を図 6-4 に示す。図 6-4 より、 R_N/R_T に対する Y はほぼ同一直線上に並び、 R_T 、 R_N 自身の大きさに関係なく一次関数で近似可能となる。 R_T はトランスコーダへの動作パラメータであり取得可能な値であるから、トランスコード処理において SCR 復号値とトランスコード処理の経過時間より処理経過時間増加率 Y を求めて、さらに図 6-4 に示す近似直線を利用して R_N を推定可能となる。図 6-4 の近似直線を表す一次関数を $F[\cdot]$ と定義し、 Y と R_N/R_T の関係を式 (6.6) に示す。

$$Y = F\left[\frac{R_N}{R_T}\right] = C \times \frac{R_N}{R_T} \quad (6.6)$$

図 6-4 から $C = 0.888$ として求められる。

伝送速度変動による処理経過時間増加率への影響

本実験では、伝送速度が変化した場合のトランスコード処理経過時間の増加率への影響について検証する。トランスコード処理開始から 30[sec] が経過した時点で伝送速度が $R_N^{(1)}$ [kbit/sec] から $R_N^{(2)}$ へ瞬時に変化する場合を考える。トランスコーダの変換レートを $R_T = 1000, 500$ [kbit/sec] に設定し、 $R_N^{(1)} = 437, 874$ [kbit/sec] のそれぞれの場合における $R_N^{(2)}$ に対する処理時間増加特性を図 6-5 に示す。

図 6-5 より、処理経過時間と SCR 復号値の増加割合は伝送速度との関連性があるために、伝送速度が $R_N^{(1)}$ から $R_N^{(2)}$ へ変化した場合、処理時間増加率はそれに呼応して変動し、その応答は瞬時的となることが確認できる。また、図 6-5(a)(b) 間および図 6-5(c)(d) 間の比較により $R_N^{(2)}$ が等しいプロット同士での直線の傾きは等しく、以前の伝送速度 $R_N^{(1)}$ の影響を受けないことが分かる。したがって、伝送速度の推定値は処理時間増加率の現時点での観測値のみを用いて瞬時的に求める可能といえる。

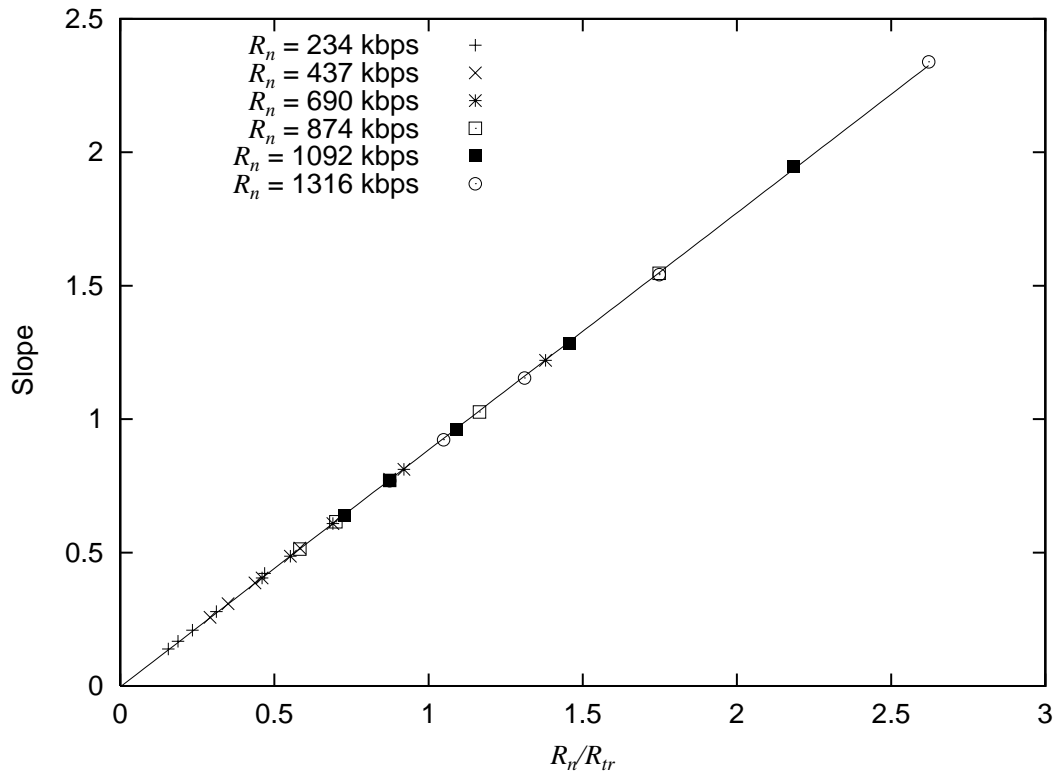


図 6-4: 処理経過時間増加率 Y と R_N/R_T の関係

以上の考察より，伝送速度 R_N は，トランスコードの変換レート R_T と，入力ストリームの各パックヘッダから取得される SCR 復号値とトランスコード処理の経過時間から求められた処理経過時間増加率 Y を式 (6.6) に代入することにより推定可能となる．

6.4 提案方式

本節では，基礎実験の結果を受けてトランスコード処理経過時間と SCR 復号値を利用してトランスコードの目標ビットレートの算出を行うレート制御方式を提案する．

提案方式では，入力ストリームにおける W 枚のピクチャ間隔でトランスコードの変換レートを更新する． W 枚のピクチャで構成されるレート更新単位をセグメントと定義する．第 n 番セグメントのトランスコード処理直前において，第 $n-1$ 番セグメント終了時点での目標変換レート $R_T(n-1)$ を更新して当該セグメントへ適用する変換レート $R_T(n)$ を導出する制御を行う．提案方式は以下の 3 ステップで構成される．

Step 1 : 処理経過時間増加率導出制御

Step 2 : 処理経過時間収束目標値制御

Step 3 : 目標変換レート更新制御

$n-1$ 番セグメントの処理終了時点での変換レート，伝送速度，処理経過時間増加率をそれぞれ $R_T(n-1)$ ， $R_N(n-1)$ ， $Y(n-1)$ とする．Step1 では，トランスコード処理経過時間

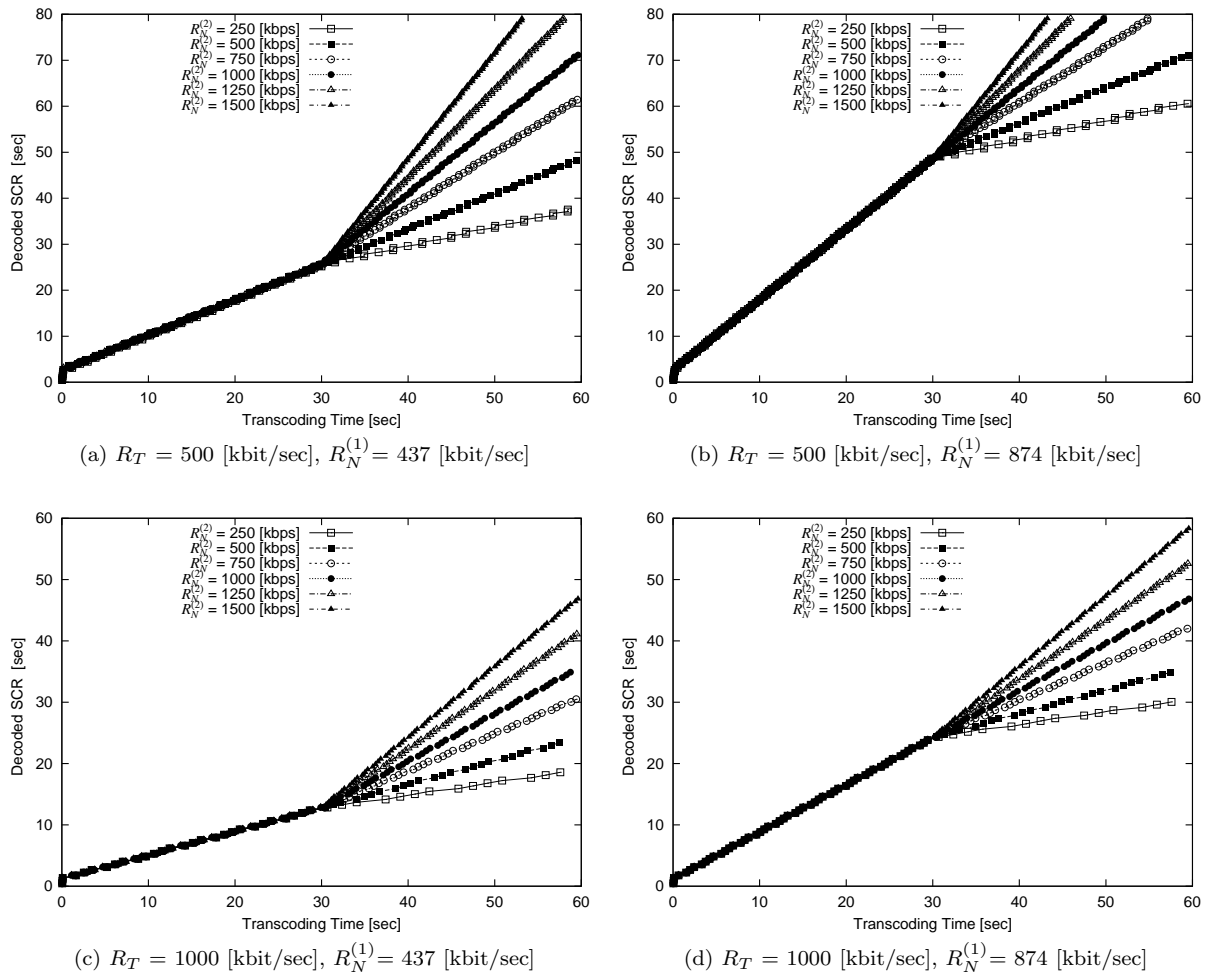


図 6-5: トランスコード処理経過時間と SCR 復号値の関係 (伝送速度変動あり)

と復号 SCR 値を用いて $Y(n-1)$ を導出する。Step2, Step3 では, 変換レート $R_T(n-1)$, 伝送速度 $R_N(n-1)$, 処理経過時間増加率 $Y(n-1)$ が式 (6-6) で関係づけられることを利用して, $Y(n)$ を特定値へ収束させるように $R_T(n-1)$ を補正して $R_T(n)$ へ更新する。

6.4.1 処理経過時間増加率導出制御 (Step 1)

Step1 では $n-1$ 番セグメントのトランスコード処理終了時点での処理経過時間増加率 $Y(n-1)$ を取得する。 $n-1$ 番セグメント末尾が入力ストリームの $\tau(n-1)$ [sec] なる時刻の符号化データであるとする。この符号化データのバイト位置を b , 直前のパックヘッダにおける SCR の復号値を $SCR'(n-1)$, この SCR 情報が格納されているバイト位置を b' とそれぞれ定義すると, $\tau(n-1)$ は式 (6-7) のように求められる。

$$\tau(n-1) = \frac{SCR'(n-1)}{f} + \frac{8 \times (b - b')}{R_{in}} \quad (6-7)$$

R_{in} は入力ストリームのビットレートである。 $\tau(n-1)$ は b なるバイト位置にある符号化データのシステムターゲット復号器への入力時刻を表し, 以下では, $\tau(n-1)$ をストリー

Δ 時刻と定義する．また，この符号化データ処理時点でのトランスコード処理経過時間を $t(n-1)$ とする．

処理経過時間増加率 $Y(n-1)$ を， $n-1$ 番を含む直前の S 個のセグメントで観測された処理経過時間 $\{t(n-1), t(n-2), \dots, t(n-S)\}$ およびストリーム時刻 $\{\tau(n-1), \tau(n-2), \dots, \tau(n-S)\}$ を用いて求める．7.3 より， $t(n)$ と $\tau(n)$ の関係は一次関数で近似可能であり，近似関数の変化の割合が $Y(n-1)$ に相当する．そこで， S 個の $[t(n-i), \tau(n-i)]$ プロット ($i = 1, 2, \dots, S$) を用いて最小二乗近似で求めた近似直線の傾きとして $Y(n-1)$ を導出し，式 (6.8) に示す．

$$Y(n-1) = \frac{S \sum_{i=1}^S t(n-i)\tau(n-i) - \sum_{i=1}^S t(n-i) \cdot \sum_{i=1}^S \tau(n-i)}{S \sum_{i=1}^S \{t(n-i)\}^2 - \left(\sum_{i=1}^S t(n-i)\right)^2} \quad (6.8)$$

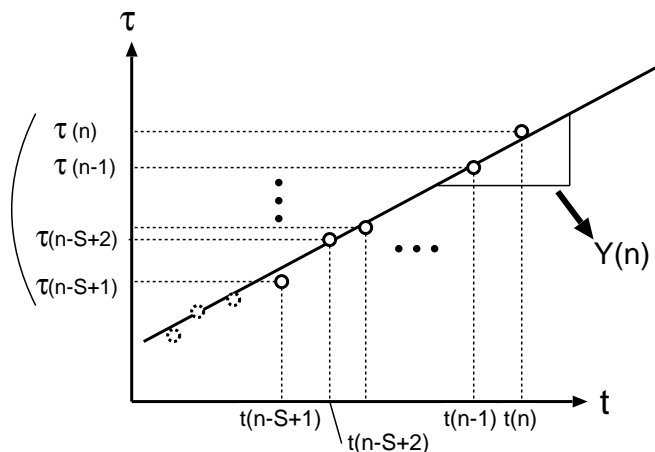


図 6-6: 最小二乗近似による $Y(n)$ の導出

6.4.2 処理経過時間収束目標値制御 (Step 2)

Step 2 ではトランスコード処理経過時間 $t(n-1)$ とストリーム時刻 $\tau(n-1)$ を比較評価して，両者の偏差から Step3 で適用するための処理経過時間増加率の収束目標値を設定する．

$t(n-1) < \tau(n-1)$ の場合， $n-1$ 番セグメント末尾の符号化データが表すストリーム時刻が，本データバイトを処理した時刻よりも先行している．トランスコードは， $t(n-1)$ [sec] が経過した時点で $t(n-1)$ よりも後方の時刻の符号化データを処理しており，トランスコード処理が実際の時間よりも先行していることを意味する．このような場合には，トランスコードの符号発生量を増大させることによって処理経過時間を遅れさせて， $t(n)$ と $\tau(n)$ を漸近させる制御を行う．逆に， $t(n-1) > \tau(n-1)$ の場合には， $t(n-1)$ [sec] 経過時点で処理中のデータバイトはそれより前の時刻を表すから，処理の遅れを意味する．この場合には， $t(n)$ が $\tau(n)$ へ追いつくように目標ビットレートを下げる制御を働かせる．

そこで， $R_N = R_T$ における処理経過時間増加率 $Y|_{\frac{R_N}{R_T}=1.0} = F[1.0]$ を基準にして，

$t(n), \tau(n)$ から Y の収束目標値 B を式 (6-9) のように設定する .

$$B = F[1.0] \times \left[1 - r \times \left\{ t(n) - \tau(n) \right\} \right] \quad (6-9)$$

式 (6-9) において, r はリアクション係数を表し, Step 2 のフィードバック制御の強さを制御するパラメータである .

6.4.3 目標変換レート更新制御 (Step 3)

Step 3 では, Step 2 で設定した収束目標値 B へ収束するように $Y(n)$ を制御することで, n 番セグメントへ適用する目標変換レート $R_T(n)$ を算出する . Step 1 で得られた $Y(n-1)$ と式 (6-6) を用いて伝送速度 $R_N(n-1)$ と目標変換レート $R_T(n-1)$ との関係を評価し, $Y(n)$ を B へ漸近させるように $R_T(n)$ を制御する . 手順の詳細は以下のとおり .

(1) $Y(n-1)$ に基づく伝送速度の推定

$n-1$ 番セグメントで設定された変換レート $R_T(n-1)$ と式 (6-6) で求められる $Y(n-1)$ を用いて, 伝送速度 $R_N(n-1)$ の推定値を式 (6-10) により算出する .

$$R_N(n-1) = F^{-1}[Y(n-1)] \cdot R_T(n-1) \quad (6-10)$$

(2) 処理経過時間目標値の設定

n 番セグメントに置ける Y の目標値 $Y_t(n)$ を式 (6-11) により求める .

$$Y_t(n) = A \cdot (Y(n-1) - B) + B \quad (6-11)$$

ここで, $Y_t(n) = Y(n)$ であるならば式 (6-11) は 1 階の線形差分方程式となる . したがって, 式 (6-11) は $Y(n)$ を指数関数的に B へ収束させるように目標値 $Y_t(n)$ を設定することを意味し, このときの時定数を A とする . 時定数 A は制御のフィードバックの強さを表すパラメータとなる .

(3) 目標変換レート $R_T(n)$ の導出

式 (6-6), 式 (6-11) より $R_T(n)$ は式 (6-12) となる .

$$R_T(n) = \frac{R_N(n)}{F^{-1}[Y_t(n)]} \quad (6-12)$$

$R_N(n) = R_N(n-1)$ を仮定して式 (6-10), 式 (6-11), 式 (6-12) より n 番セグメントへ適用する目標変換レート $R_T(n)$ への更新式を式 (6-13) に導く .

$$R_T(n) = \frac{F^{-1}[Y(n-1)]}{F^{-1}[A \cdot (Y(n-1) - B) + B]} \cdot R_T(n-1) \quad (6-13)$$

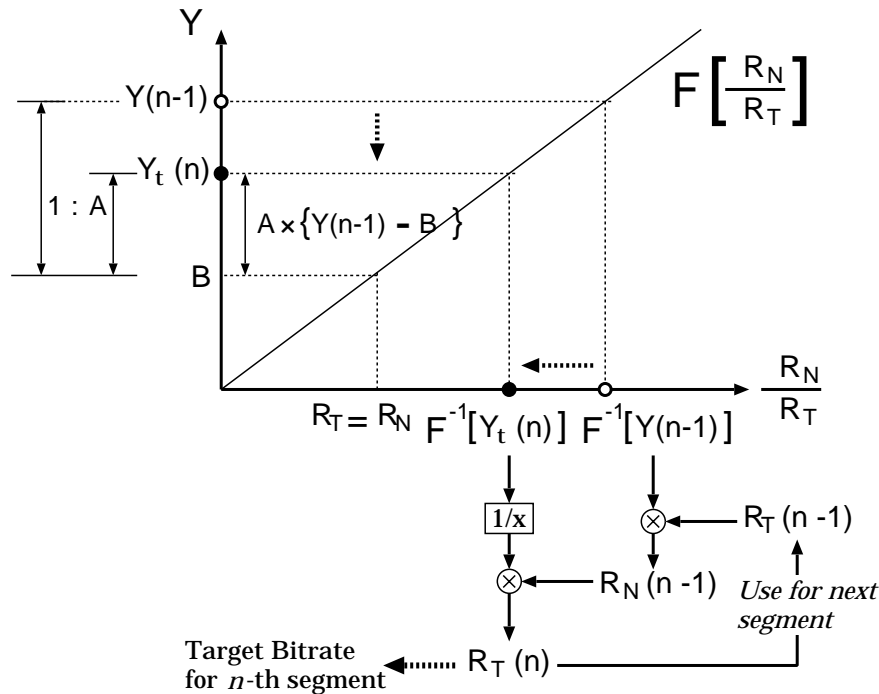


図 6-7: 処理経過時間増加率を用いた目標レートの算出

6.5 シミュレーション実験

提案方式について，(i) 目標変換ビットレートの変動，(ii) 処理経過時間増加率，(iii) 復号バッファ内滞留時間，の各観点に関してシミュレーション実験から評価を行う．ただし，本評価実験では，図 6-3 に示した $t(n)$ と $\tau(n)$ の変化率 (処理経過時間増加率) に着目したレート制御方式の実現可能性を言及するまでに留めて，Step 2 を適用せずに $r = 0$ とした場合に限定する．ただし，リアクション係数 r に関する検討と $t(n)$ と $\tau(n)$ の比較に基づく制御方式の確立は今後必要である．

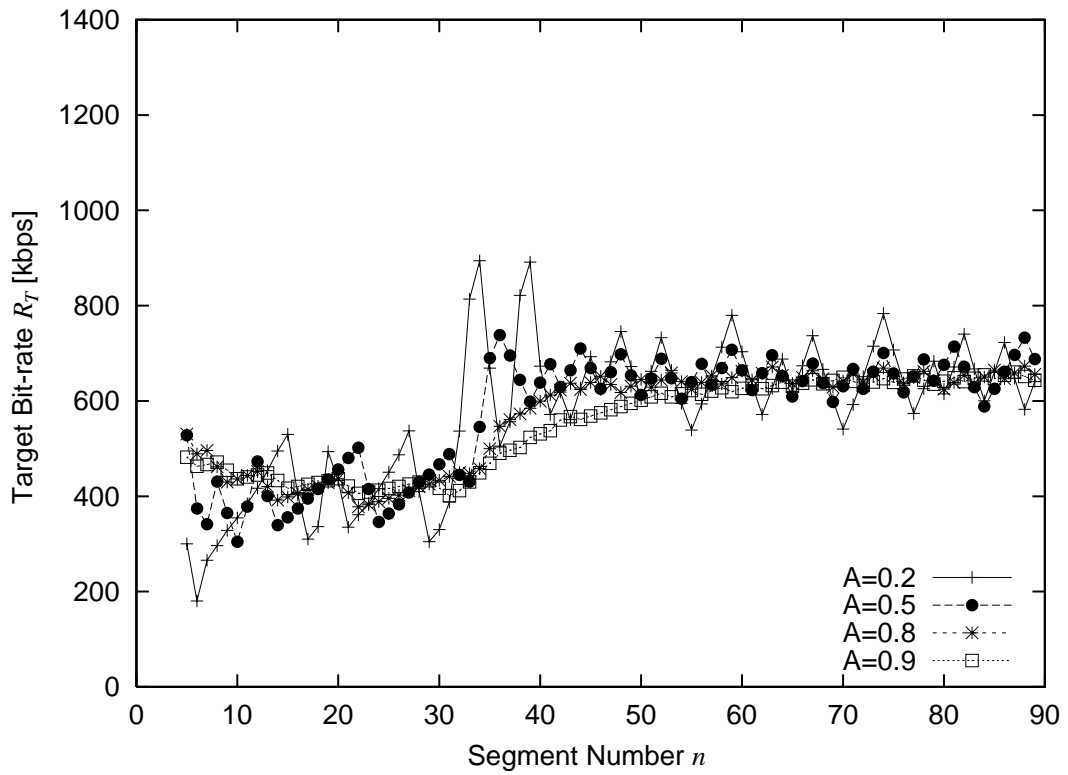
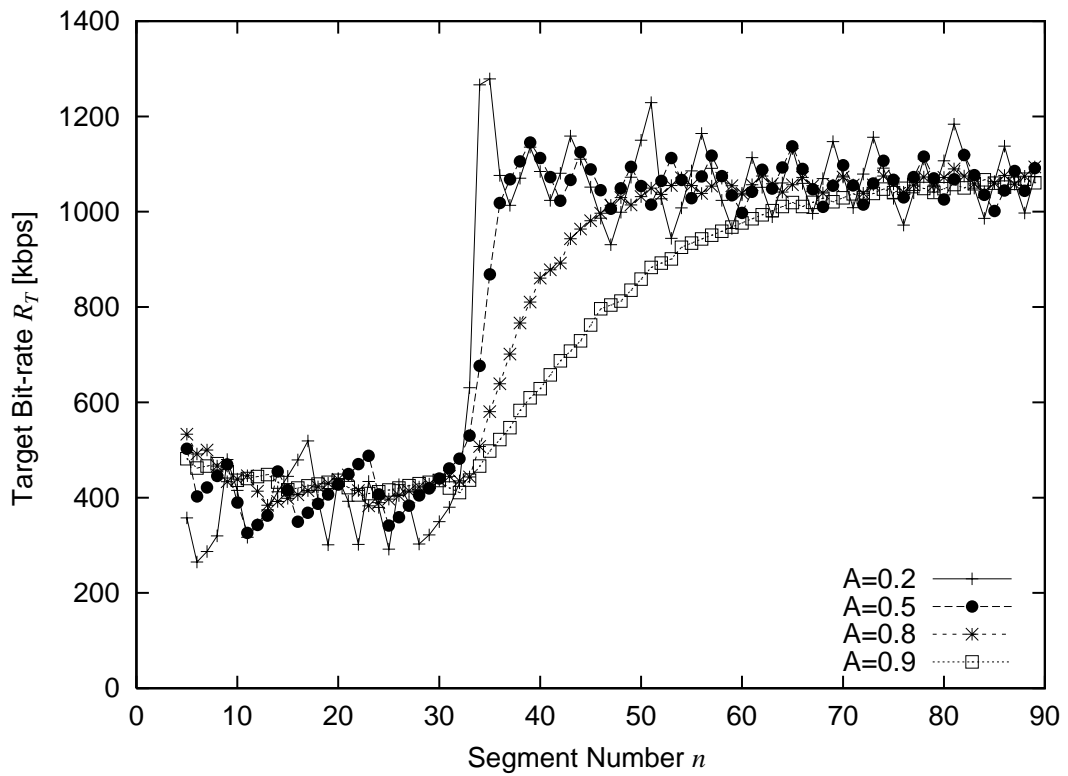
実験系の構成および実験用ビットストリームの条件は 6.3.2 と同様である．トランスコード処理開始から 30[sec] 後に伝送速度 R_N が変化する環境において，速度変動に対する制御の応答特性を評価する． R_N の設定値を式 (6.14) に示す．

$$R_N = \begin{cases} 473 \longrightarrow 689, 1092 \\ 1092 \longrightarrow 872, 473 \end{cases} \quad [\text{kbit/sec}] \quad (6.14)$$

目標レート更新間隔を $W = 30$ に，時定数を $A = \{0.9, 0.8, 0.5, 0.2\}$ にそれぞれ設定した場合における各種特性を比較する．

6.5.1 目標変換レート変動特性

式 (6.14) に示す R_N の条件における R_T の特性を図 6-8，図 6-9 に，処理経過時間増加率 $Y(n)$ の特性を図 6-10，図 6-11 にそれぞれ示す．提案方式により R_N が未知の場合でも， R_T を R_N を重心とした領域で推移させる制御を実現可能であることが分かる．これよ

(a) $R_N = 473 \rightarrow 689$ [kbit/sec](b) $R_N = 473 \rightarrow 1092$ [kbit/sec]図 6-8: 目標変換ビットレート R_T の変化 (方式 I, その 1)

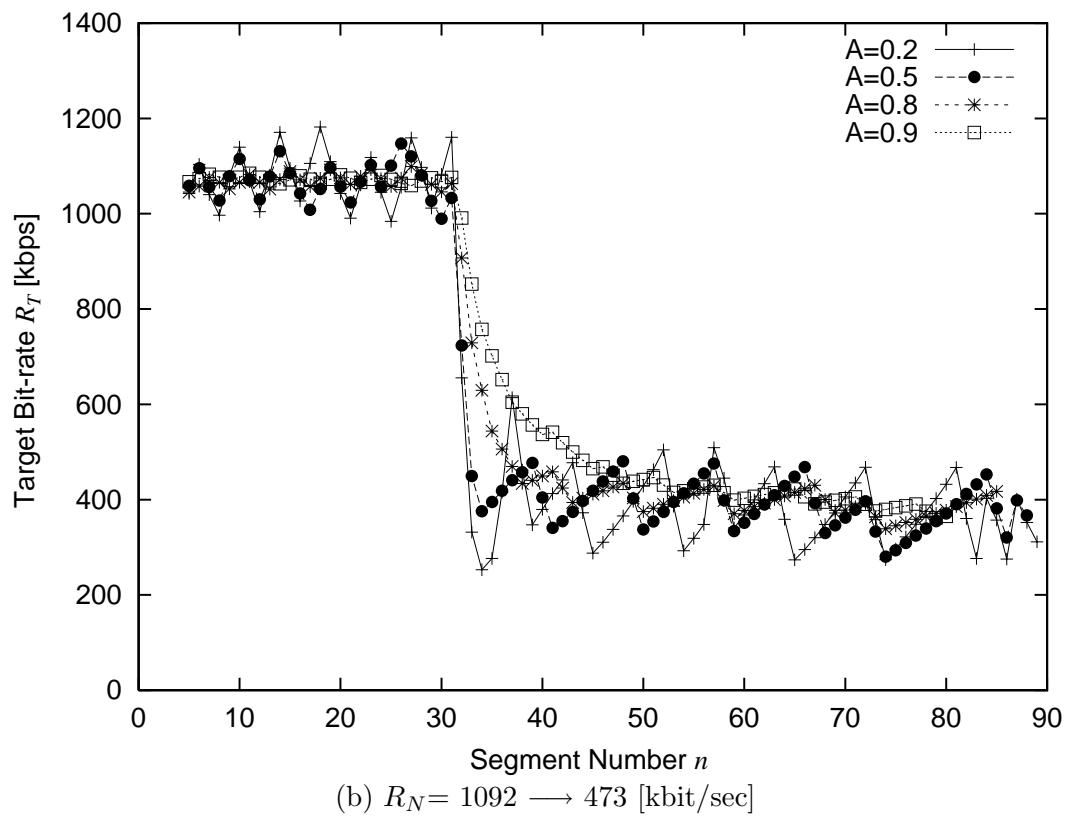
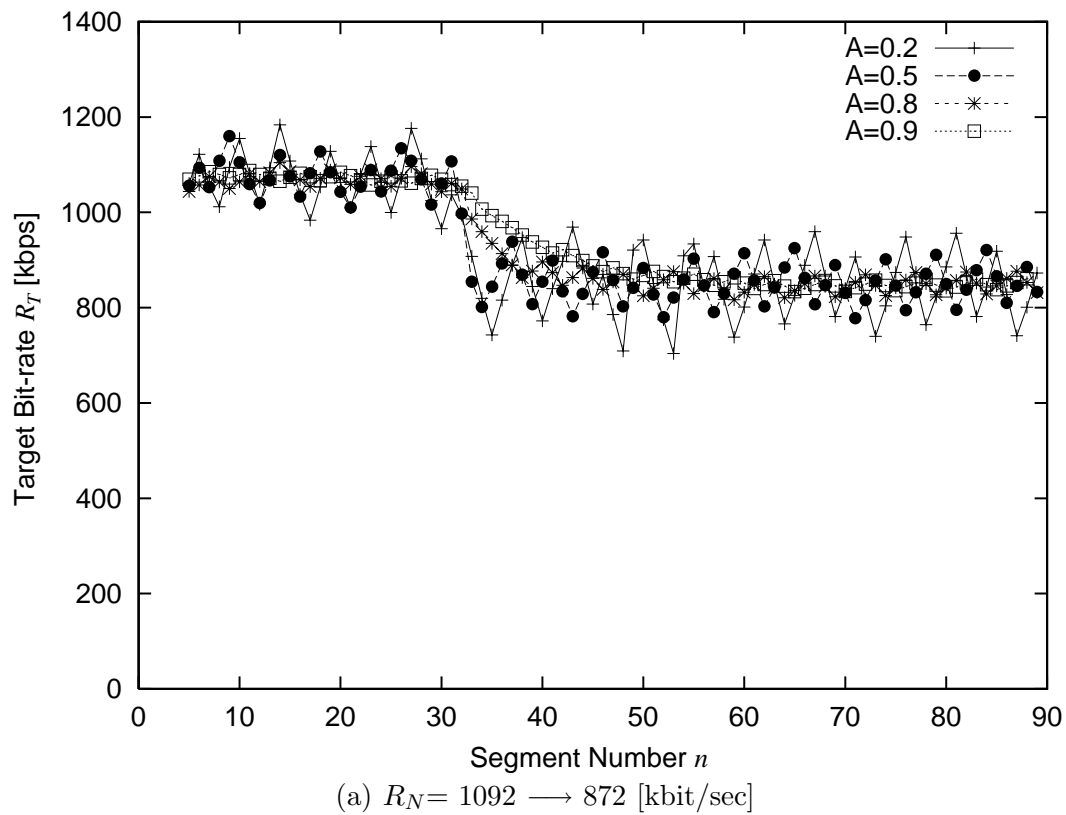
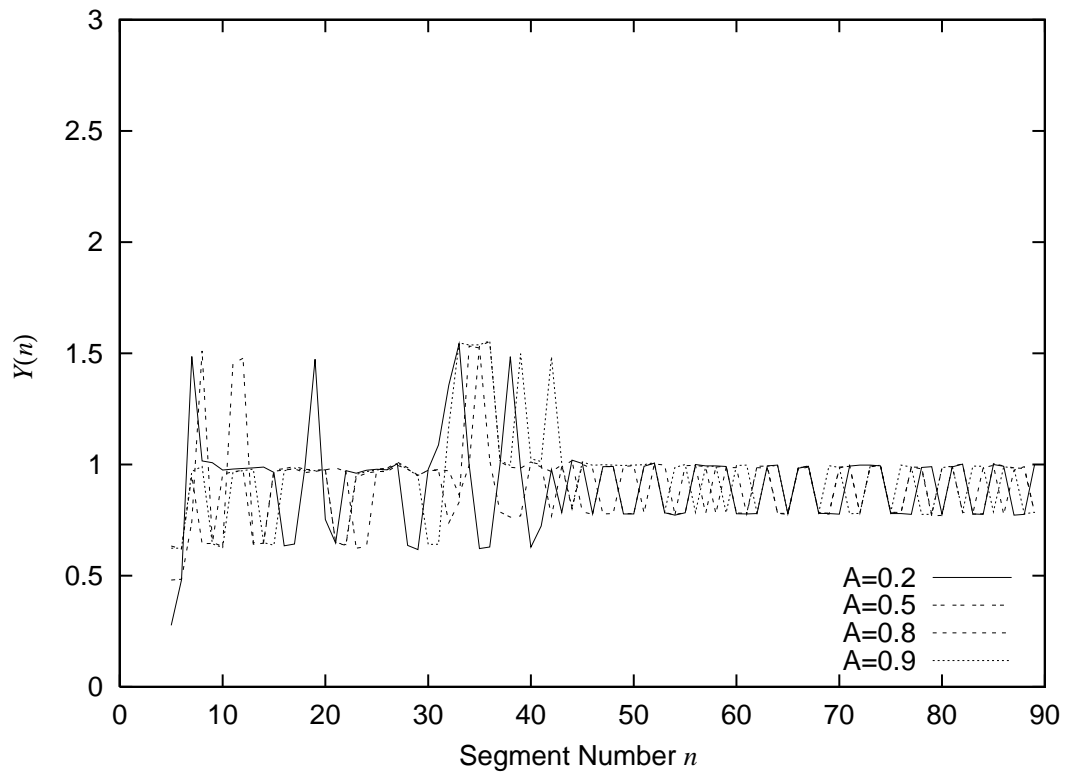
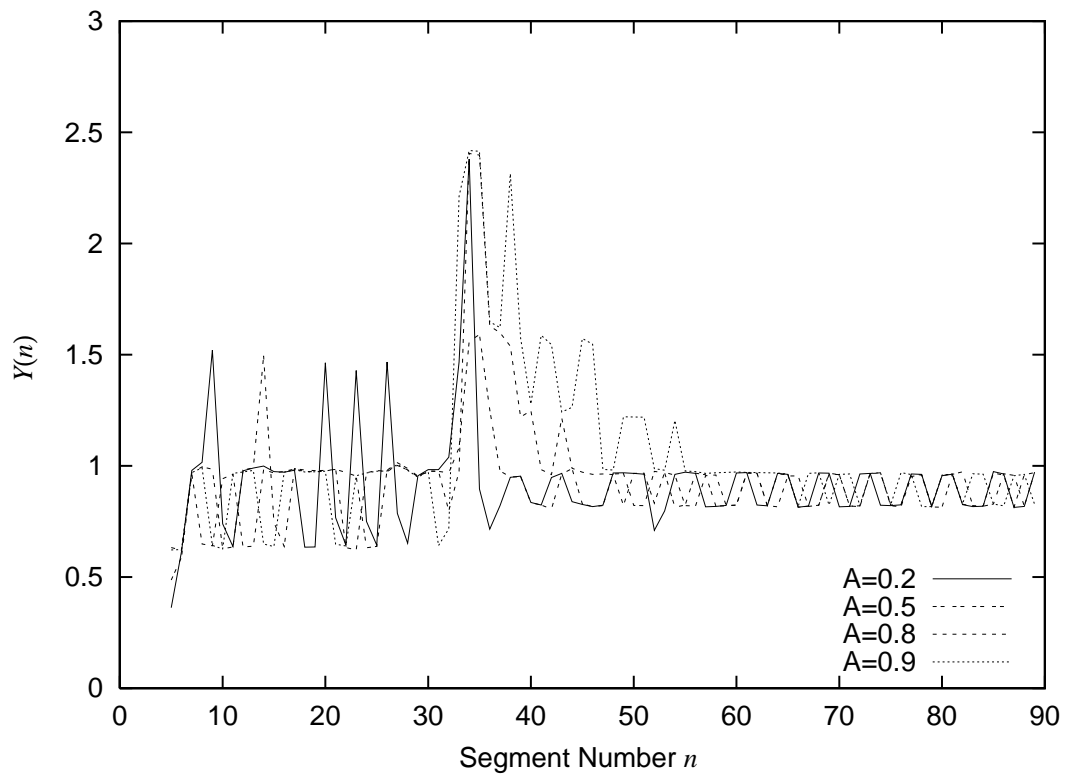


図 6-9: 目標変換ビットレート R_T の変化 (方式 I, その 2)

(a) $R_N = 473 \rightarrow 689$ [kbit/sec](b) $R_N = 473 \rightarrow 1092$ [kbit/sec]図 6-10: 処理経過時間増加率 $Y(n)$ の変動 (方式 I, その 1)

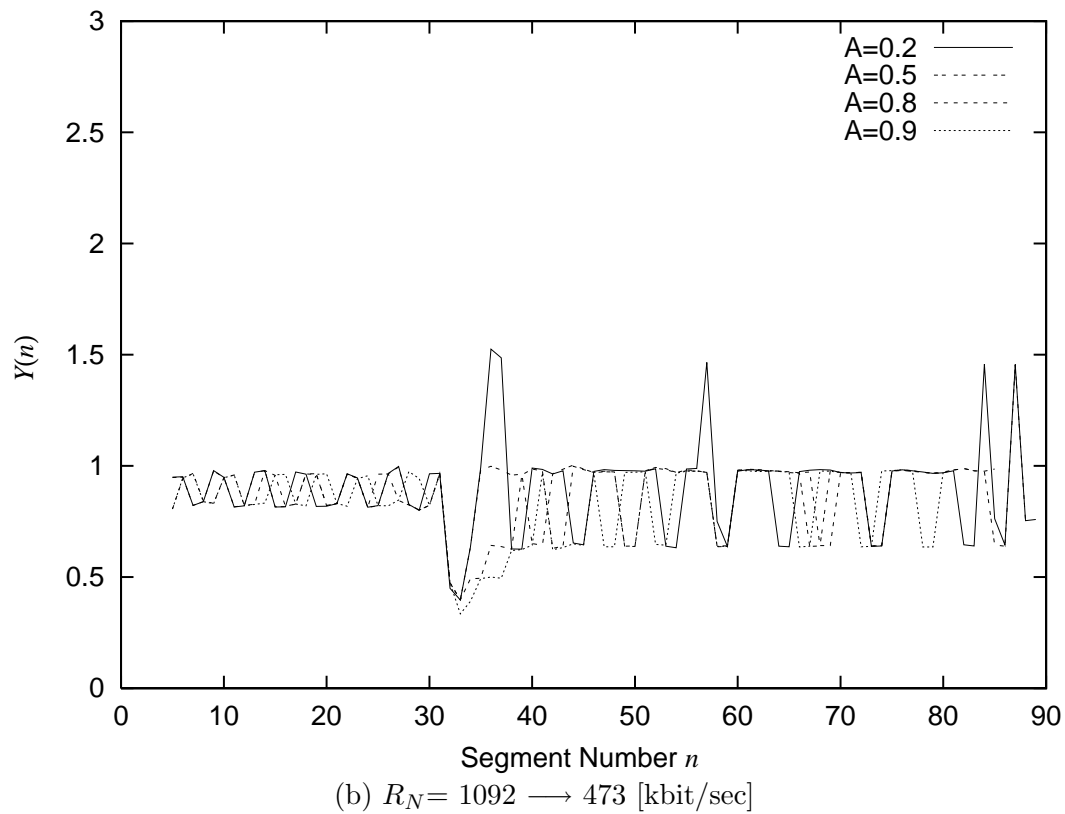
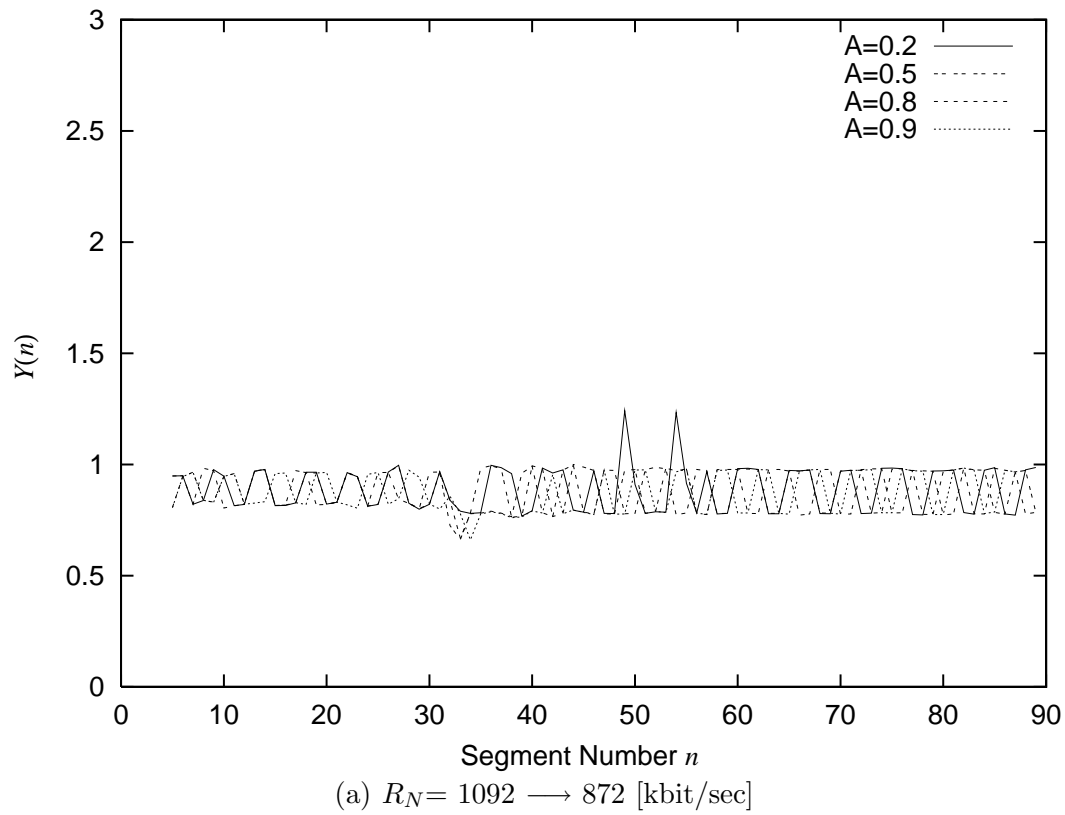


図 6-11: 処理経過時間増加率 $Y(n)$ の変動 (方式 I, その 2)

り、提案制御方式の適用によりトランスコーダに対して目標ビットレートを与えなくても自動的に現在の伝送速度に応じたビットレートへの変換が実現される。 R_N が変化した場合でも R_N の変化に追従可能であり、変動後の R_N を重心とする領域へ R_T の変動領域を収束可能であることが図 6-8、図 6-9 により確認される。そして、このとき収束速度は時定数 A によって制御される。

提案方式では、Step 1 での経過時間 $t(n-1)$ とストリーム時刻 $\tau(n-1)$ から求められた処理経過時間増加率 $Y(n)$ と収束目標値 B との偏差 $Y(n-1) - B$ に一定比率 $A (< 1.0)$ を乗じて $Y(n-1) - B$ を減じるように目標値 $Y_t(n)$ を設定することで、 $Y(n)$ を B へ指数関数的に収束させる。このような $Y(n)$ の遷移は図 6-10(b) で明確に確認できる。このとき、 A が小さいほど $Y(n-1) - B$ を大幅に減じる制御が働く。この場合、前回設定時に対する $Y_t(n)$ の変化量は大きく、それに伴い $R_T(n)$ を大きく変化させる制御となる。結果として R_N の変化に対する感度が高くなり $R_T(n)$ を瞬時に変化させることが可能となる。

ただし、感度が高いほど $Y(n)$ の微小な変動による影響を強く受けようになり、 R_N が一定の定常状態において R_N の振動幅が大きくなり安定性は低下する。この傾向は R_N の大幅な変化後に顕著に観られ、図 6-8(b) および図 6-9(b) において $A = 0.2$ のときに定常状態で最大で 200[kbps] 前後の振動が発生する。

以上の考察より、 A が大きいほど応答速度は遅いが高い安定性を有し、 A が小さいほど応答速度が向上する反面、安定性は劣化することが分かる。定常状態での安定性と、 R_N の変化に対する応答性能はトレードオフの関係にあり、その特性は時定数 A によって決まる。安定性を維持可能な条件のもとで最大の応答性能を実現可能な時定数 A の最適値の導出と、応答性能及び安定化条件の理論的な解析は今後の課題である。

6.5.2 処理経過時間増加率の特性

図 6-10、図 6-11 より、定常状態では $Y(n)$ は 0.6 ~ 1.0 を中心に安定的して変動しており、 R_N が高いほど変動幅が狭いことが確認される。 R_N が低い領域ではトランスコーダからの発生符号量が少ないために 1 セグメントに要する処理時間が短くなり、式 (6-8) に示す経過時間 $t(n-1)$ の間隔が密になる。これにより、 $Y(n)$ の測定対象となる $t(n-1)$ の幅が狭くなるので近似直線の傾きへの感度が高くなるために、 R_N が低いほど $Y(n)$ の変動幅が大きくなるものと考えられる。

R_N の変化後の過渡的な状態では $Y(n)$ は上記で述べた安定領域から外れて、 R_N の増加に対して $Y(n)$ は増加し、 R_N の減少時には $Y(n)$ も減少する。そして、変動幅が大きいほど $Y(n)$ の変化量も大きいことがわかる。 R_N の変化直後の時点ではトランスコーダの目標変換ビットレートの更新制御へ反映されていないので、出力ビットレートは R_N に対して大きく乖離する。例えば、図 6-8(b) の 473 から 1092[kbps] への変化直後では、トランスコーダ出力ビットレートは 1092[kbps] を大きく下回る。このときの $t(n)$ と $\tau(n)$ の関係は図 6-3(e) の $R_N = 1092$ [kbps] のプロット () に近い特性となる。変化直後で $R_T = 473$ [kbps] とすると $R_N/R_T = 1092/473 = 2.308$ となり、図 6-4 から $Y(n) = 2.05$ となる。一方、図 6-10(b)

において $Y(n)$ は 2.0 ~ 2.5 の領域まで上昇することが確認でき、6.3.3 の考察と一致する。

したがって、伝送速度の変動が $Y(n)$ の安定領域からの乖離として観測可能であり、提案方式により $Y(n)$ を指数関数的に安定領域へ戻す制御を行うことで、 R_T を R_N へ収束させる制御が実現されることが確認された。

6.5.3 復号バッファ内滞留時間に関する評価実験

本実験では、復号器内における各ピクチャの復号バッファの滞留時間に関する評価実験を行う。本研究で着目する伝送能力に応じた最適なビットレートへの変換方式の目的は、伝送から復号および再生までに時間的制約がある状況における復号器内のストリームの連続性の確保であり、そのためには各ピクチャの復号時刻より前に当該ピクチャの全ての符号化データが受信端末に到着済である必要がある。そこで、本実験では 6.5.1 に示す R_T の制御結果の妥当性を各ピクチャの復号器内バッファの滞留時間の観点から評価する。

復号器内バッファ滞留時間の導出

各ピクチャの復号処理はフレームレートから求められる等しい時間間隔で行われ、復号処理自体は瞬時に完了すると仮定する。このときの、復号器における各ピクチャの復号バッファ滞留時間を導出する。

フレームレートを $FrameRate$ [Frames/sec]、 k 番ピクチャ ($k \geq 0$) の復号時刻を $T_{dec}(k)$ とおく。0 番ピクチャの入力開始時刻を基準時刻とすると $T_{dec}(k)$ は式 (6.15) で与えられる。

$$T_{dec}(k) = T_s + \frac{k}{FrameRate} \quad (6.15)$$

式 (6.15) において T_s を 0 番ピクチャの復号時刻とする。これは、はストリーム受信開始から再生までのバッファリングによる初期遅延量を決定するパラメータである。

次に、 n 番ピクチャの入力終了時刻 $T_{in}(k)$ について考察する。 $n-1$ 番ピクチャの入力終了時刻から n 番ピクチャの入力処理に要する時間が経過した時点での時刻が、 n 番ピクチャ自身の入力完了時刻となるから、 $T_{in}(k)$ は式 (6.16) に示す漸化式により表現される [81]。

$$T_{in}(k) = T_{in}(k-1) + \frac{d(k)}{R_C(k)} \quad (6.16)$$

式 (6.16) において $d(k)$ は k 番ピクチャの符号量、 $R_C(k)$ は k 番ピクチャ受信時点での受信速度である。

$T_{in}(0) = d(0)/R_C(0)$ であることに着目して式 (6.16) に示す漸化式を解くことで $T_{in}(k)$ は式 (6.17) として求められる。

$$T_{in}(k) = \sum_{i=0}^k \frac{d(i)}{R_C(i)} \quad (6.17)$$

式 (6-15), 式 (6-17) より k 番ピクチャの復号バッファ内滞留時間 $T_B(k)$ を式 (6-18) に導出する .

$$T_B(k) = T_s + \frac{k}{FrameRate} - \sum_{i=0}^k \frac{d(i)}{R_C(i)} \quad (6-18)$$

$T_B(k)$ の変動に関する評価

$T_B(k) > 0$ は復号時刻より前に全てのピクチャデータが到着済であることを意味し, 復号における連続性の確保が可能となる . ただし, $T_B(k)$ が非常に大きい場合には復号器内のバッファリングに起因した遅延量が大きいことを意味し, 遅延に対する要求が高いアプリケーションでは $T_B(k)$ の上限値には注意を要する . 逆に, $T_B(k) < 0$ の場合には, ピクチャデータの到着が再生時刻より遅れた状態となるので, ストリーム再生の連続性を確保できなくなる . そのため, $T_B(k) > 0$ を最低条件として, 適切な $T_B(k)$ を確保することが R_T の制御の目的となる . また, R_T の変動域の重心が R_N にある場合には各セグメントにおける R_T と R_N の差分の総和 $\sum_{i=0}^n R_T(i) - R_N(i)$ は相殺され, 伝送能力に適した発生符号量の制御が実現されている . 式 (6-17) よりバッファへの入力完了時刻は各ピクチャの符号量の総和として表されるから, このときには平均的に $T_B(k)$ は一定値を維持する .

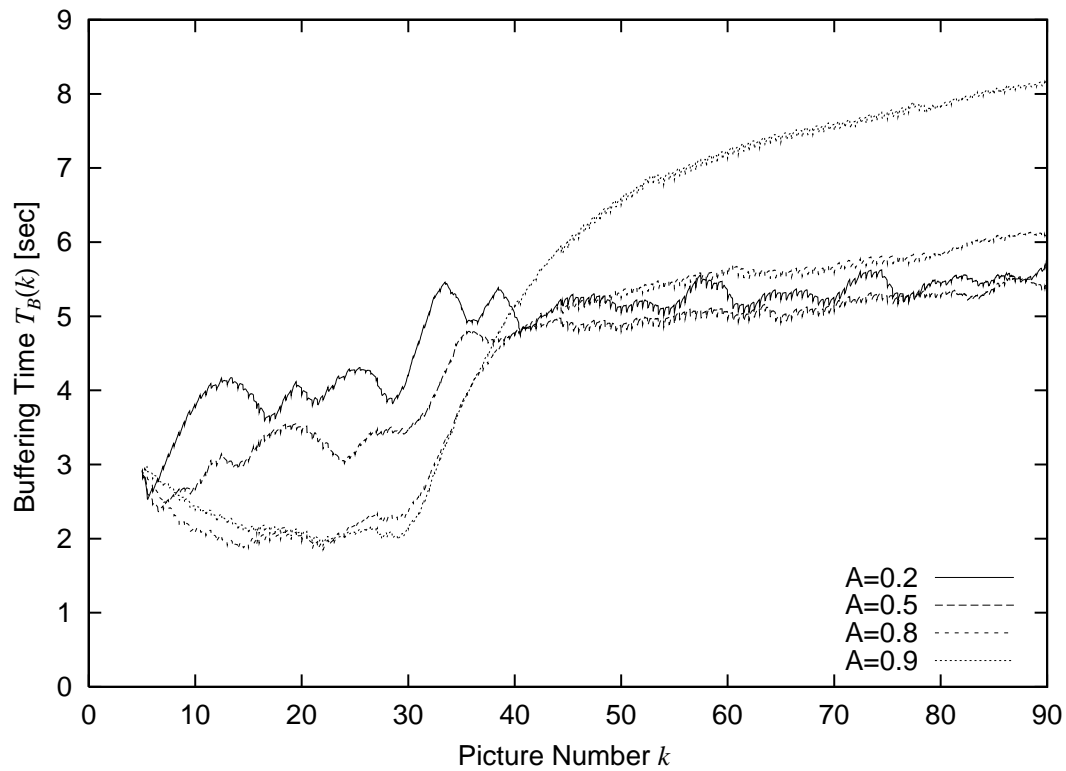
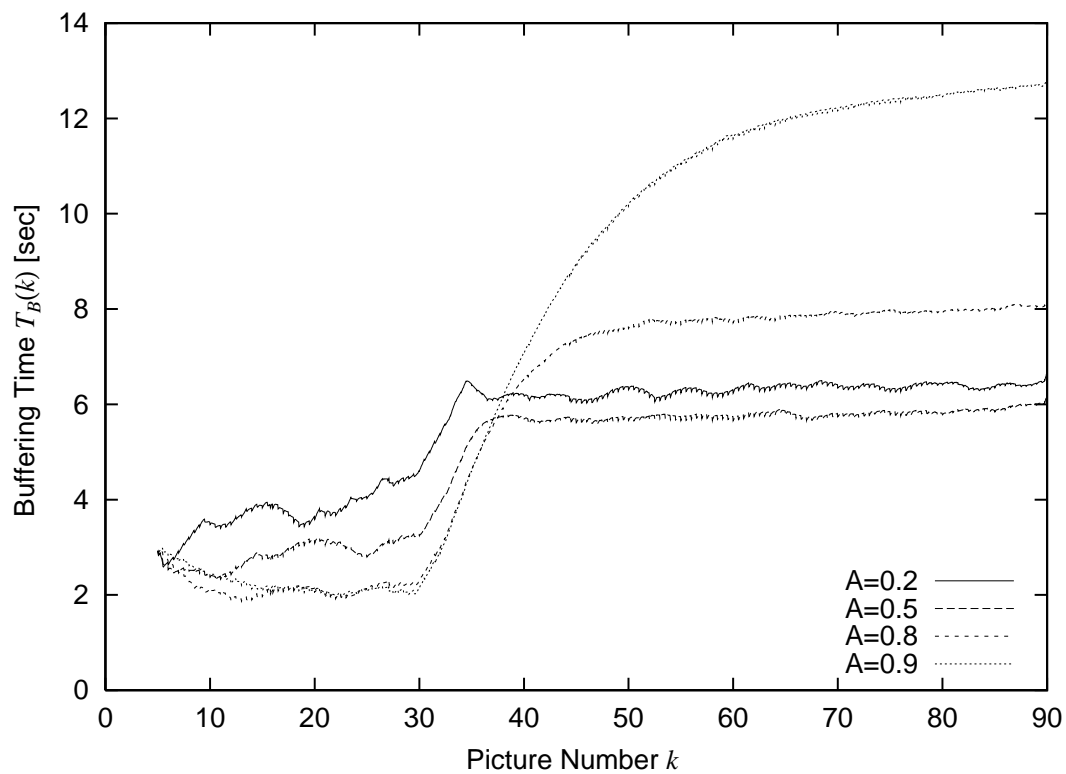
(1) 定常状態における $T_B(k)$ の安定性

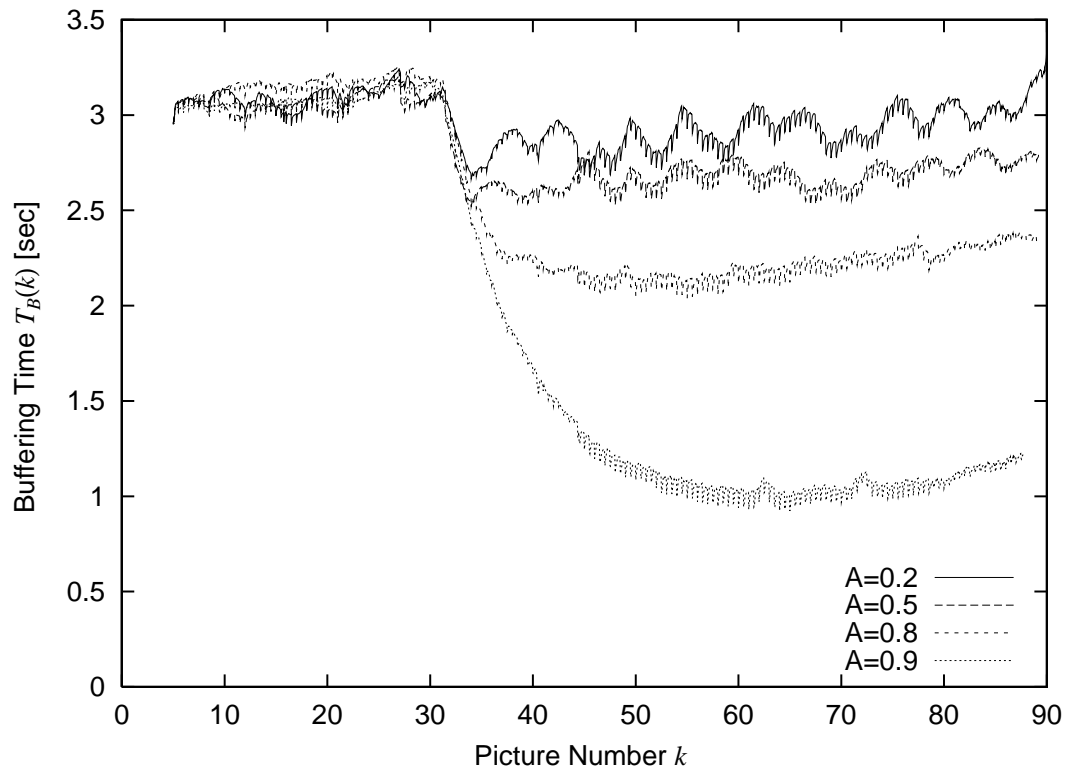
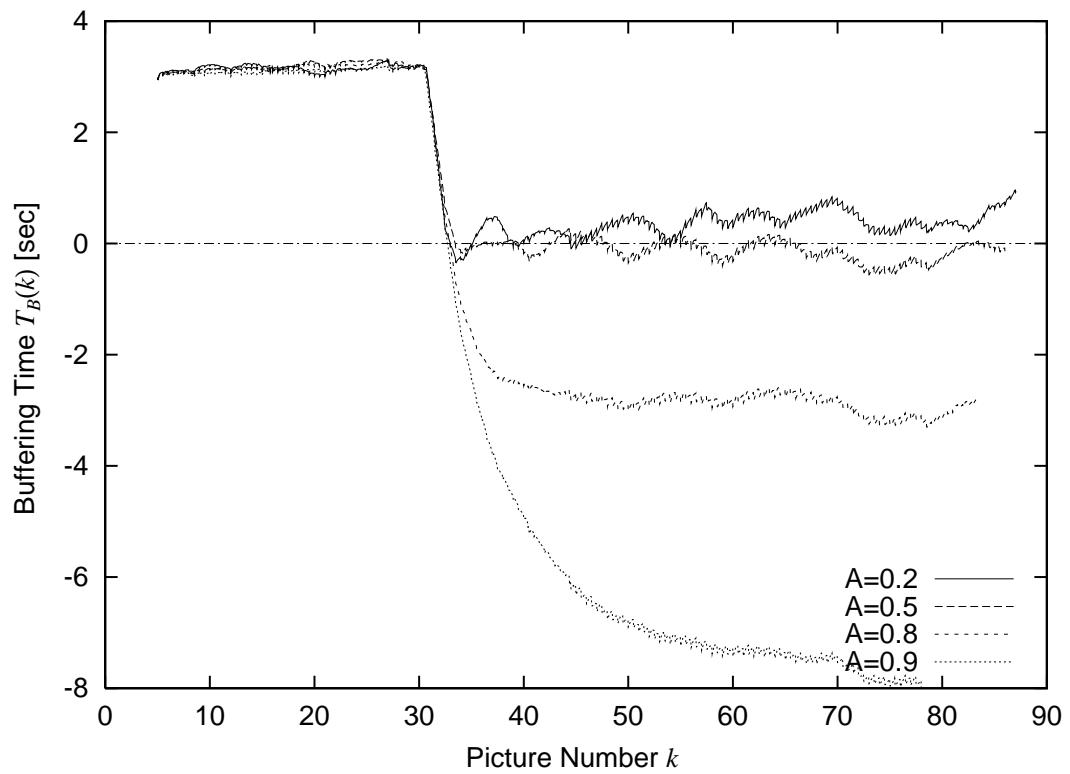
$T_s = 3.0[\text{sec}]$ として初期遅延量を設定したときの各フレームにおける $T_B(k)$ の変化を図 6-12, 図 6-13 に示す . 定常状態では, R_N が低い場合 (図 6-12 の $n < 30$ および図 6-13 の $n > 30$ の領域) では $T_B(k)$ の変動幅が大きいものに対して, R_N が高い領域 (図 6-12 の $n < 30$ および図 6-13 の $n > 30$ の領域) では $T_B(k)$ はほとんど変化せず一定となりバッファの安定性を確保可能であることが分かる . これは, 6.5.2 で考察したように R_N が低い領域では制御の感度が高くなる傾向にあるために R_T への変動が大きく, これが $T_B(k)$ の変動として現れていると考えられる .

(2) 過渡状態における $T_B(k)$ の応答特性

R_N の変動に対する応答特性を評価する . R_N の変化直後で $T_B(k)$ が大きく変動し, R_T と同様に時定数 A によってその特性は異なることが確認される . $A = 0.2, 0.5$ の場合には R_T の制御の応答性が良いことに起因して $T_B(k)$ の安定化が速いものに対して, $A = 0.8$ では安定化に $A = 0.2, 0.5$ の場合の 2 倍以上のピクチャ数を要し, $A = 0.9$ では本実験での観測区間内 (1800 フレーム) では $T_B(k)$ は安定化しなかった . $A = 0.9$ は $T_B(k)$ の安定化に要する時間が多大であることから, 復号器バッファ状態の観点からは適切な応答性能の確保は困難といえる . 一方, $A = 0.2, 0.5$ の場合には R_N の上昇時には約 240[Frame], R_N 下降時には 120[Frame] の遷移間隔となる . 実験用ビットストリームのフレームレートは 29.97[Frame/sec] であるから, バッファの安定化のためには約 4 ~ 8[sec] の応答時間を必要とする .

図 6-8, 図 6-9 における $A = 0.2, 0.5$ のような, 定常状態での R_T の安定性が十分でない場合でも, 図 6-12, 図 6-13 より R_N が一定の領域での $T_B(k)$ の変動は高々 $\pm 0.5[\text{sec}]$ である . 一方, R_N 変化時の過渡状態での応答性能には明確な差が確認されているので, A

(a) $R_N = 473 \rightarrow 689$ [kbit/sec](b) $R_N = 473 \rightarrow 1092$ [kbit/sec]図 6-12: 各ピクチャの復号バッファ内滞留時間 $T_B(k)$ の変動 (その 1)

(a) $R_N = 1092 \rightarrow 872$ [kbit/sec](b) $R_N = 1092 \rightarrow 473$ [kbit/sec]図 6-13: 各ピクチャの復号バッファ内滞留時間 $T_B(k)$ の変動 (その 2)

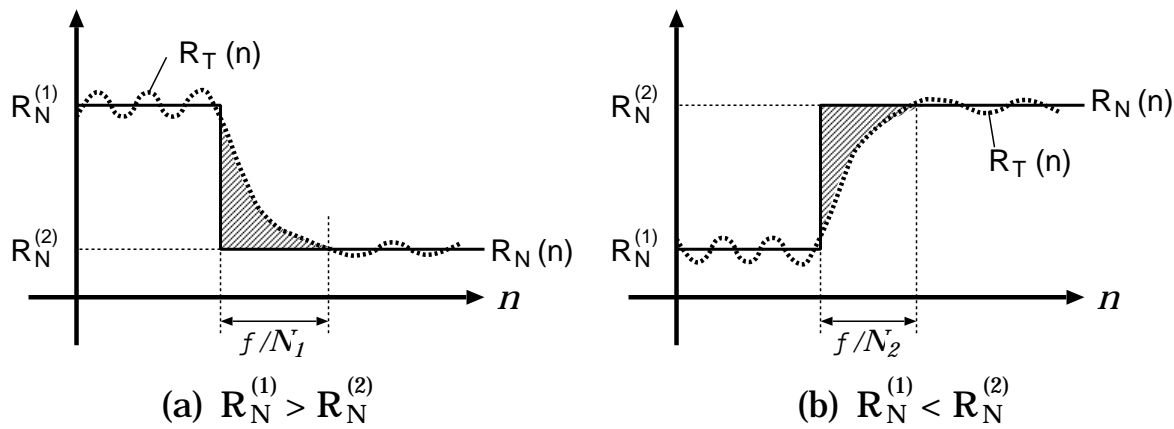


図 6-14: 過渡状態における R_N と R_T の不均衡領域

の設定値を低く設定しての収束速度を速めた方が復号バッファ内の定常化の確保の観点では適切と言える。

ただし、 $T_B(k)$ の定常化後の変動域の重心が R_N 変動に伴って上下にシフトし、変化前の状態に復元しないことが確認される。 R_N の変動直後で R_T の遷移中の過渡的な状態では、一時的に R_N と R_T の均衡が取れない状態が発生する。図 6-14 は R_N の変化直後における R_T の応答を表しており、(a) では ΔN_1 の区間内では常に $R_T(n) > R_N(n)$ であり、(b) の場合には ΔN_2 の区間で常に $R_T(n) < R_N(n)$ となる。両ケースともに、当該区間内における R_N と R_T の差分の積算量 $\sum_{i=0}^n R_T(i) - R_N(i)$ は相殺されず、これが以降の全てのピクチャのバッファ内の滞留時間に影響する。例えば、図 6-14(a) の場合には、 ΔN_1 の区間で常に伝送能力を上回る符号量が発生しており、本区間内での各ピクチャの符号化データの受信に要する時間が引き延ばされる。これにより、安定化した以降の全てのピクチャの到着が全体的に後方へシフトし、結果として復号開始までのバッファリング遅延時間が短縮されて $T_B(k)$ が減少する。逆に、図 6-14(b) では、過渡区間内で各ピクチャの受信に要する時間が短縮されて、以降の全てのピクチャの到着時刻が早くなり $T_B(k)$ が増大する。

本方式は、 $\tau(n)$ と $t(n)$ の変化率のみに着目しており、変化率を一定値へ収束されることで伝送能力が未知の場合でも適切な変換レートの制御を実現したが、 $\tau(n)$ と $t(n)$ の絶対量自身を用いた制御を適用していないために、過渡状態での R_N と R_T の乖離状態が以降へ及ぼす影響を相殺する制御とはなっていない。そのため、今後は一定のバッファ内滞留時間 $T_B(k)$ の確保の観点でリアクション係数 r の最適値を検討して Step 2 を含めた制御方式を確立することが課題となる。

6.6 まとめ

本章では、伝送速度に適したビットレートへの変換を行うトランスコーダにおいて、変動する伝送帯域に応じて適応的に変換レートの制御を行う目標変換レート動的制御方式について検討した。伝送プロトコルとして TCP/IP を想定し、TCP/IP 上での輻輳制御に起因するストリーム送出処理のボトルネックがもたらす処理能力への影響をトランスコーダ側で観

測することで帯域変動を推定する方式を検討した。

入力ストリーム内の同期制御情報に対するトランスコード処理の経過時間の関係に関する基礎実験により、トランスコードの処理性能は伝送速度の影響を受けて変動し、変換レートと伝送速度間の比率が、SCR 復号値に対するトランスコード処理経過時間の増加率によって定式化可能であることを明らかにした。それに基づき、処理経過時間増加率、トランスコード出力ビットレート、伝送速度との関係を表す特性関数を導出した。次に、本特性関数に基づく変換レート動的制御方式を提案し、目標変換ビットレート、処理経過時間増加率、復号バッファ内滞留時間の各観点から評価を行った。その結果、伝送速度が未知であっても、自動的に適切な変換レートへの制御が実現可能であり、伝送速度の変動にも追従可能な制御方式であることを明らかにした。特に、処理経過時間増加率の収束制御に用いる時定数により定常状態での安定性と帯域変動時の応答性に大きな特性の違いが存在することを述べた。両者はトレードオフの関係にあり、最適な時定数の導出は今後の課題となる。また、帯域変動時の過渡状態において各ピクチャの復号バッファ滞留時間は大きく変動することから、復号バッファの安定化の課題も残されている。

提案方式では伝送帯域の推定を実験的に行っており、ネットワークとのインタフェース部分の配信制御機構の状態、送信及び受信バッファの状態、TCP/IP プロトコルの実装方式 [82] [83] [84] [85] の特性等を考慮したものではない。そのため、提案方式は TCP/IP の伝送制御に基づいて厳密に変換レートを制御する方式ではない。したがって、帯域変動に応じた動的レートスケージング行うストリーミング配信制御に本方式を適用するためにはさらなる検討を必要とする。具体的には以下に示す項目が挙げられる。

- トランスコードとインタフェースの非同期動作時の帯域推定パラメータの変動対策
提案トランスコードではレート制御部分と伝送路とのインタフェース部分が同期して動作し、輻輳制御による速度変動現象が比較的早期にレート制御へ反映されることを前提としている。しかし、両者を分離して別スレッドとして動作させて、トランスコードの出力をバッファリングする場合、両スレッド間のタスク切り替えやバッファメモリへのアクセス制御により処理経過時間の観測が影響を受けることが考えられる。そこで、このような実装方式に依存しない制御方式の実現が課題となる。
- インターネット上での映像配信実験
本研究では受信端末側で受信速度を制限して帯域制御を擬似的に再現した。本研究は、インターネットなどの帯域変動が発生するネットワーク上での映像配信制御方式への適用を目的としたものである。そのため、実在のネットワークシステムで実験を行い有効性を明らかにする必要がある。
- 処理能力の変動への対応
提案トランスコードでは処理の経過時間の観測を行って制御を行うため、PC の処理能力に依存した方式となる。本研究では、処理時間は TCP/IP の輻輳制御によるのみ影響を受けることを前提としたが、実際には、ハードウェア自体の処理性能、ディスクへのアクセスやカーネルへのシステムコールの性能、他プロセスの負荷変動に起

因するトランスコードの処理優先度の変化等によりトランスコードの処理速度は変動する．そのため，式(6.6)に示す特性関数 $F[\cdot]$ の比例定数 C はPC環境に依存した値であり一意に決められるものではない．そこで，トランスコードの動作環境に基づいて比例定数 C を自動的に取得する方式が必要となる．

第7章

再量子化/再符号化処理共有によるマルチレート出力ビデオトランスコーダ

7.1 はじめに

本章では、1本の入力ストリームを品質の異なる複数ストリームへの同時変換機能(以下、マルチレート変換機能と呼ぶ)を実現するマルチレート変換トランスコーダにおいて、出力レート数増加に対しても処理量を収束可能な高い処理効率性を有するアーキテクチャを提案する。まず、マルチレート変換の従来方式を考察し、従来方式において要求される処理量は出力数に比例的に増大するために同時出力可能なストリームの数に限界が存在することを述べる。次に、同一の再量子化ステップ値を有する再量子化器/再符号化器を共有化してマルチレート変換における処理量を削減したマルチレート出力トランスコーダを提案する。さらに、提案方式における再量子化/再符号化処理量の特性を理論的に考察する。最後に、評価実験に基づき提案方式の処理コストを評価し有効性を示す。なお、出力数や出力レートの選択方法は検討対象外とし、トランスコーダアーキテクチャに焦点を当てて論ずる。

7.2 マルチレート出力トランスコーダの従来方式

マルチレート変換機能の実現方法として、1入力1出力型のトランスコーダ(以下、シングルレート出力トランスコーダと呼ぶ)をユーザの品質要求ごとに適用する方法が考えられる。しかし、変換処理に要する処理量がユーザ数に比例して増大し、トランスコーダを実装する配信サーバや中継ノードの負荷が高まる。これを解決するには、処理量増加の少ない効率的なマルチレート変換アルゴリズムが必要である。

そこで、本節では、マルチレート出力トランスコーダを実現する従来方式として、以下の2方式について処理量の観点から考察する。

1. シングルレート出力トランスコーダ並列型
2. 復号/逆量子化器共用型 [52] [59]

なお、トランスコード処理方式として、動き補償予測ループを持たないDCT係数領域内の再量子化方式 [27] [74](以下、DCT係数領域トランスコーダと呼ぶ)を対象とするが、復号器と符号化器のカスケード接続型トランスコーダでも同様の議論が成立する。

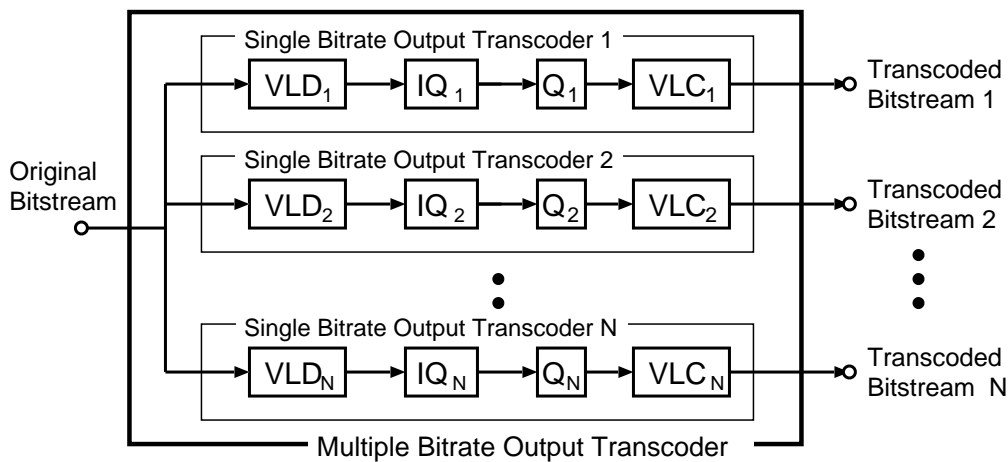


図 7-1: シングルレート出力トランスコーダ並列型

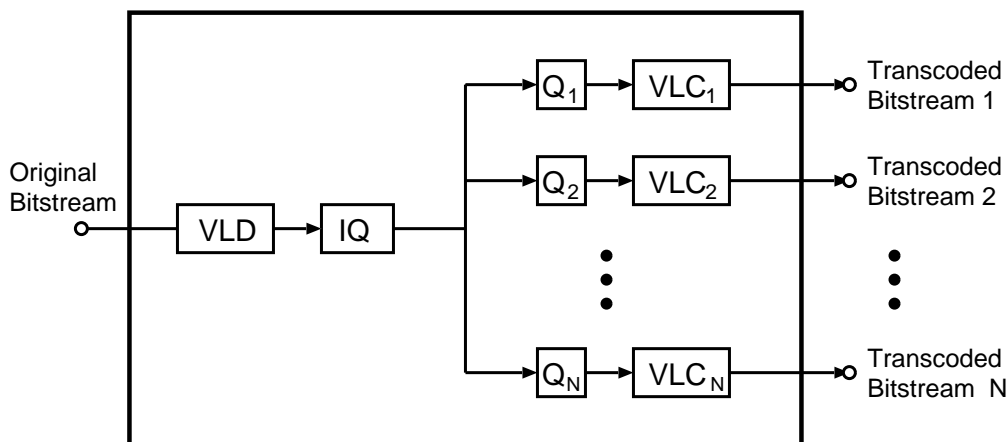


図 7-2: 復号/逆量子化器共用型

7.2.1 シングルレート出力トランスコーダ並列型

本方式は図 7-1 のようにシングルレート出力トランスコーダを出力数だけ並列接続してマルチレート変換機能を実現する。各トランスコーダは1つの出力レートへの変換に対応し、全体として複数ストリームを出力する。これは、任意のトランスコーダを適用できるが、出力レート数と同数のトランスコーダを必要とする。

7.2.2 復号/逆量子化器共用型

復号と逆量子化は入力ビットストリーム(以下、単にストリームと呼ぶ)に対する処理で、出力レートに関係なく共通である。そこで、図 7-2 に示すように、一組の復号器と逆量子化器を共用する。本方式は、逆量子化後の DCT 係数を各出力レートに対応する再量子化器へ分配して、複数ストリームへ変換する。

7.2.3 処理コストの比較

7.2.1, 7.2.2 の両方式について出力数に対する処理量を比較する．出力数を N , 復号系処理 (復号, 逆量子化処理) の処理量を X_{dec} , 符号化系処理 (再量子化, 再符号化処理) の処理量を X_{enc} とする．シングルレート出力トランスコーダ並列型の処理量 $F_1(N)$ と復号/逆量子化器の共用型の処理量 $F_2(N)$ は, それぞれ式 (7.1), 式 (7.2) の通りである．

$$F_1(N) = N \cdot (X_{dec} + X_{enc}) \quad (7.1)$$

$$F_2(N) = X_{dec} + N \cdot X_{enc} \quad (7.2)$$

式 (7.1), 式 (7.2) より, 両方式の処理量は $O(N)$ で増加する．よって, トランスコーダを実装するハードウェア性能が向上しても同時出力のストリーム数には限界がある．この問題を解決するために, DCT 係数領域内トランスコーダの特徴に着目し, $O(N)$ 以下の処理量でマルチレート変換を実現する方式を 7.3 で提案する．

7.3 再量子化/再符号化器共用によるマルチレート出力トランスコーダ

トランスコーダにおいて, 高品質性と高速性はトレードオフの関係にあり, システム要件に適したアーキテクチャ選択が必要である．本論文では, リアルタイム伝送時のマルチレート変換器を対象としており, 特に, 多出力レートへの同時変換が重要な要件である．文献 [27] では, 使用アプリケーションごとに適したアーキテクチャについて考察しており, このような用途においては, 処理の簡略性, フレームメモリの不要性などの理由で動き補償予測ループを省いた DCT 係数領域トランスコーダの有効性が示されている．

そこで, 本節では DCT 係数領域トランスコーダを拡張して, マルチレート変換機能を実現する方式を提案し, その詳細を述べる．

7.3.1 基本原理

DCT 係数領域トランスコーダはマクロブロック (以下, MB と呼ぶ) ごとに, 再量子化ステップ値を変換パラメータとした再量子化演算によりレート削減する．この方式では予測誤差信号を再計算しないため, 再量子化出力結果は再量子化ステップ値によって一意に決定される．したがって, 1 つの入力 MB に対する再量子化の結果は有限であり, 同一再量子化/再符号化結果を異なるストリームへ出力できる場合がある．よって, 7.2.2 を元にして, 同一の出力符号化データを有するストリーム間で再量子化/再符号化処理を共用して, さらに処理量を低減できる．

この場合, 1 つの MB につき必要な再量子化/再符号化回数は, 各出力レートごとに算出する再量子化ステップ値の中の相異なる値の個数と同数となる．そのため, この再量子化ステップ値の中に重複した値が多く含まれるほど処理量削減効果が高い．筆者らはレート歪特性に基づき再量子化ステップ値に禁止領域を設ける量子化制御方式を提案した [34] ．本方式では再量子化ステップ値を特定値へ制限するために重複が発生する確率が高くなる．すな

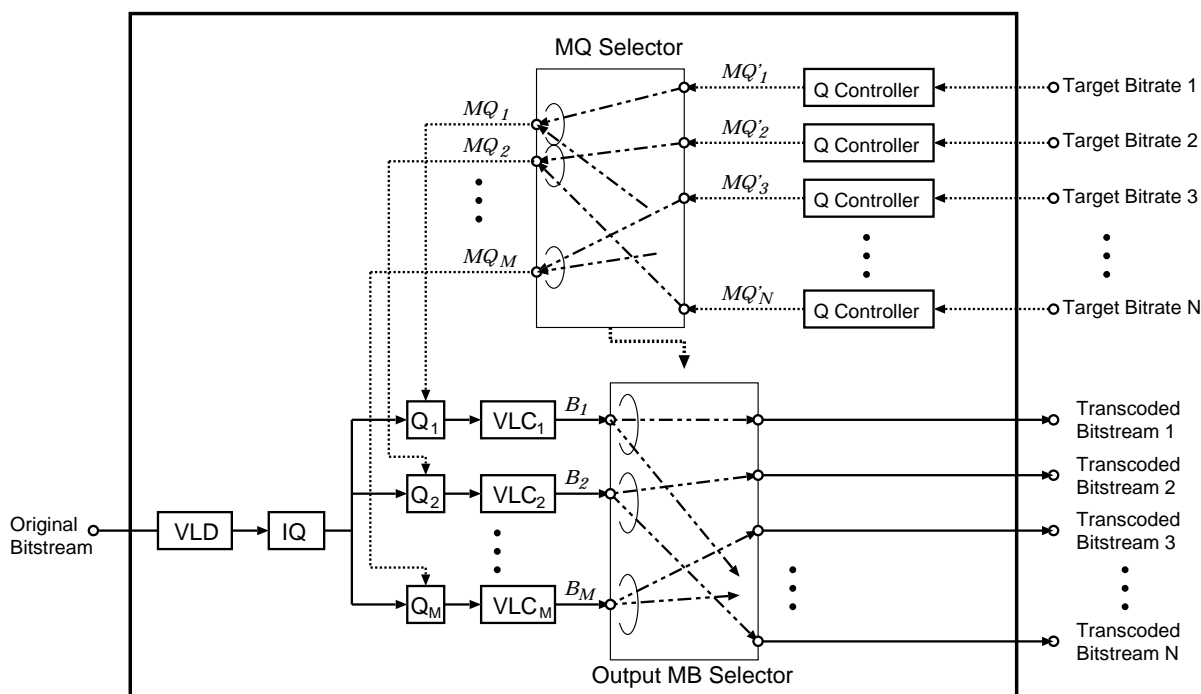


図 7-3: 提案マルチレート出力トランスコーダブロック図

わち，本制御を量子化制御器へ追加することで再量子化/再符号化処理回数の削減効果をさらに期待できる．

7.3.2 処理器の構成

図 7-3 に提案マルチレート出力トランスコーダのブロック図を示す．本トランスコーダは，復号処理器と逆量子化処理器を 1 組と，再量子化処理器と再符号化処理器を M 組で構成する．また， N 個の量子化制御器 (Q Controller) を有し，各量子化制御器はそれぞれの目標レートに応じ再量子化ステップ値を出力する．次に，再量子化ステップ値選択器 (MQ Selector) により，複数の再量子化ステップ値から重複を除外し再量子化器で使用する再量子化ステップ値を選択する．選択した再量子化ステップ値の個数は再量子化/再符号化回数 M と同値である．すなわち，再量子化/再符号化器数 M は MB ごとに異なり N 個の量子化制御器から生成される再量子化ステップ値に依存する．

本方式は，DCT 係数信号を逆量子化後に M 組の再量子化/再符号化器へ分配する．さらに， M 種の実出力 MB 符号化データを，出力 MB 符号選択器 (Output MB Selector) によって等しい再量子化ステップ値を有した複数の出力ストリームへ振り分けて， N 本の出力ストリームを生成する．

逆量子化後の DCT 係数信号は M 組の再量子化/再符号化器へ分配される．

7.3.3 MB 符号変換処理手順

1 個の入力 MB から出力 MB 符号化データを生成する処理は以下の手順に従う。

- 入力 MB 符号化情報の復号および逆量子化
- 候補量子化ステップ値の算出制御
- 再量子化ステップ値選択処理
- 再量子化，再符号化処理
- 出力 MB 符号化信号の分配処理

以下に，各処理の詳細を述べる。

1. 入力 MB 符号化情報の復号および逆量子化

入力ストリームから MB1 個分に相当するストリーム素片を復号・逆量子化して DCT 係数を復元する。

2. 候補量子化ステップ値算出処理

量子化制御器 (Q Controller) は N 種の出力レート毎に対象 MB の再量子化ステップ値を算出する。算出した再量子化ステップ値 (候補再量子化ステップ値) のうち第 n 番出力ストリームへの値を MQ'_n とする。

3. 量子化ステップ値選択処理

再量子化ステップ値選択器 (MQ Selector) により N 種の候補再量子化ステップ値 $\{MQ'_1, MQ'_2, \dots, MQ'_N\}$ の中から重複を排除して相異なる値を選択する。選択した M 組 ($M \leq N$) の再量子化ステップ値 $\{MQ_1, MQ_2, \dots, MQ_M\}$ を各再量子化器で使用する。

4. 再量子化/再符号化処理

3. より当該 MB で必要な再量子化/再符号化回数は M となる。1. で復元された DCT 係数信号を M 個の再量子化器へ分配して再量子化処理を行う。第 m 番再量子化器に MQ_m なる再量子化ステップ値を適用して再量子化演算を行い M 通りの再量子化係数信号を生成する。また，それぞれをラン-レベルの二次元ハフマン符号により符号化し出力 MB 符号化信号 $\{B_1, B_2, \dots, B_M\}$ を生成する。ここで， MQ_m で再量子化/符号化した出力 MB 符号は B_m に対応する。

5. 出力 MB 符号化信号の分配処理

出力 MB 符号選択器 (Output MB Selector) において，第 n 番出力ストリームに適用する出力 MB 符号化信号を $\{B_1, B_2, \dots, B_M\}$ の中から選択する。3. の再量子化ステップ値選択処理の結果を参照して， n 番出力ストリームの MQ'_n が MQ_m である場合には B_m を選択し，出力 MB 符号をストリームの部分符号として出力する。

7.4 再量子化/再符号化処理回数の解析

本章では，提案方式における1つのMBあたりに必要な再量子化/再符号化処理回数を理論的に解析し，再量子化/再符号化処理器の統合化の性能を考察する．そして，提案方式がMBあたりの再量子化/再符号化処理回数を削減可能であることを証明する．

7.4.1 平均再量子化，再符号化処理回数の導出

復号および逆量子化処理器を共用するときの出力数 N に対する処理量 $F(N)$ は式 (7.3) で表せる．

$$F(N) = X_{dec} + M_E(N) \cdot X_{enc} \quad (7.3)$$

式 (7.3) において $M_E(N)$ は平均再量子化/再符号化処理回数を表す．ここで，従来方式と提案方式における $M_E(N)$ を以下に導出する．

1. 従来方式の場合

出力数 N と同数の再量子化/再符号化処理を行うので $M_E(N) = N$ である．

2. 提案方式の場合

N 番出力ストリーム用に算出した再量子化ステップ値 MQ_N が， $1 \sim N - 1$ 番出力用の全ての再量子化ステップ値と異なる場合にのみ $M_E(N)$ は $M_E(N - 1)$ より増加する．そこで MQ_N がこの条件を満たして $M_E(N)$ が増加する確率を $P_{inc}(N)$ とすると， $M_E(N)$ に関して式 (7.4) の漸化式を得る．

$$M_E(N) = M_E(N - 1) + 1 \times P_{inc}(N) \quad (7.4)$$

一方， $P_{inc}(N)$ は式 (7.5) に導ける．

$$P_{inc}(N) = 1 - C_p \times M_E(N - 1) \quad (7.5)$$

ここで， C_p は MQ_N が第 m 番再量子化器用の再量子化ステップ値 MQ_m と等しくなる確率であり式 (7.6) に導ける (式 (7.5)，式 (7.6) の導出は次節を参照)．

$$C_p = \sum_{l=MQ_{min}}^{MQ_{max}} \sum_{k=MQ_{min}}^{MQ_{max}} \{p_{rq}(k, l)\}^2 \cdot p_q(l) \quad (7.6)$$

ここで， MQ_{min} ， MQ_{max} をそれぞれ量子化ステップ値の最小値，最大値とする． $p_q(l)$ は逆量子化ステップ値が l となる確率， $p_{rq}(k, l)$ は逆量子化ステップ値が l の条件で再量子化ステップ値が k となる確率である．式 (7.4)，式 (7.5) から $M_E(N)$ を式 (7.7) に導く．

$$M_E(N) = \frac{1 - (1 - C_p)^N}{C_p} \quad (7.7)$$

再量子化回数増加確率 $P_{inc}(N)$ および C_p の導出

式 (7.5), 式 (7.6) に示した $P_{inc}(N)$ および C_p の導出方法を述べる. 1 から $N-1$ 番出力ストリームの MB 符号生成に用いる再量子化ステップ値の集合を $\mathbf{X}_M^{(N-1)}$ と定義し式 (7.8) に示す.

$$\mathbf{X}_M^{(N-1)} = \{MQ_1, MQ_2, \dots, MQ_M\} \quad (7.8)$$

ただし, $\mathbf{X}_M^{(N-1)}$ の各要素は相異なる値であり, 要素数 M は 1 から $N-1$ 番出力ストリーム生成時において必要となった再量子化処理回数である.

再量子化回数は, $\mathbf{X}_M^{(N-1)}$ の要素数が m であることが知られている場合には, MQ_N と等しい要素値が集合 $\mathbf{X}_M^{(N-1)}$ に存在しない場合に増加する. これは, M が与えられたときの条件付き確率であり $P[MQ_N \notin \mathbf{X}_M^{(N-1)} | M = m]$ として表現できる. したがって, $P_{inc}(N)$ は式 (7.9) に表現される. ただし, $P[M = m]$ は $\mathbf{X}_M^{(N-1)}$ の要素数が m である確率を表す.

$$P_{inc}(N) = \sum_{m=1}^{N-1} P[MQ_N \notin \mathbf{X}_M^{(N-1)} | M = m] \times P[M = m] \quad (7.9)$$

$P[MQ_N \notin \mathbf{X}_M^{(N-1)} | M = m]$ を式 (7.10) により求める.

$$\begin{aligned} & P[MQ_N \notin \mathbf{X}_M^{(N-1)} | M = m] \\ &= 1 - P[MQ_N = MQ_1 \cup MQ_N = MQ_2 \dots \cup MQ_N = MQ_m] \\ &= 1 - \sum_{i=1}^m P[MQ_N = MQ_i] \end{aligned} \quad (7.10)$$

一方, 逆量子化ステップ値を $MQ^{(in)}$ と表すと, $p_q(l), p_{rq}(k, l)$ はそれぞれ式 (7.11), 式 (7.12) のように定義される.

$$p_q(l) = P[MQ^{(in)} = l] \quad (7.11)$$

$$p_{rq}(k, l) = P[MQ_N = k | MQ^{(in)} = l] \quad (7.12)$$

$MQ^{(in)}$ が l であるときに, MQ_N と MQ_i が等しくなる確率を $P[MQ_N = MQ_i | MQ^{(in)} = l]$ として表現する. 式 (7.10) において $P[MQ_N = MQ_i]$ は MQ_N が MQ_i と等しくなる確率 C_p を表し, 式 (7.13) のように求められる.

$$\begin{aligned} C_p &= P[MQ_N = MQ_i] \\ &= \sum_{l=MQ_{min}}^{MQ_{max}} P[MQ_N = MQ_i | MQ^{(in)} = l] \cdot p_q(l) \end{aligned} \quad (7.13)$$

$P[MQ_N = MQ_i | MQ^{(in)} = l]$ は $MQ^{(in)} = l$ が成立しているときに $MQ_N = k$ かつ $MQ_i = k$ となる条件付き確率を, k の値域 $MQ_{min} \leq k \leq MQ_{max}$ の全域で合計した

値となり式 (7.14) に表される .

$$\begin{aligned} & P \left[MQ_N = MQ_i \mid MQ^{(in)} = l \right] \\ &= \sum_{k=MQ_{min}}^{MQ_{max}} P \left[MQ_N = k \cap MQ_i = k \mid MQ^{(in)} = l \right] \end{aligned} \quad (7.14)$$

MQ_N と MQ_i は独立と仮定し, 式 (7.12), 式 (7.13), 式 (7.14) から, C_p を式 (7.15) に導く .

$$C_p = \sum_{l=MQ_{min}}^{MQ_{max}} \sum_{k=MQ_{min}}^{MQ_{max}} \{p_{rq}(k, l)\}^2 \cdot p_q(l) \quad (7.15)$$

式 (7.9) 式 (7.10) 式 (7.15) より, $P_{inc}(N)$ は式 (7.16) のように表される .

$$P_{inc}(N) = 1 - C_p \times \sum_{m=1}^{N-1} m \cdot P \left[M = m \right] \quad (7.16)$$

式 (7.16) において $\sum_{m=1}^{N-1} m \cdot P[M = m]$ は出力数が $N - 1$ のときの再量子化回数の期待値を意味するので $M_E(N - 1)$ として表せる . 以上から, $P_{inc}(N)$ は式 (7.17) のように導出される .

$$P_{inc}(N) = 1 - C_p \times M_E(N - 1) \quad (7.17)$$

7.4.2 平均再量子化, 再符号化処理回数の特性解析

式 (7.6), 式 (7.7) より, 提案方式での $M_E(N)$ の特性は逆量子化および再量子化ステップ値の確率分布から得られる . そこで, $p_q(l)$, $p_{rq}(k, l)$ について以下の 3 通りの確率分布を用いて $M_E(N)$ を評価する .

分布 1: 一様分布

分布 2: テストストリームを 4~14[Mbps] ヘレート変換して実測した分布

分布 3: 再量子化ステップ値禁止領域制御 [34] を適用し, 分布 2 と同条件のレート変換により実測した分布

ただし, テストストリームは ITU-R BT.601 フォーマットの 7 種類の標準画像 (Bus, Table Tennis, Football, Flower Garden, Cheer Leader, Bicycle, Mobile& Calendar) を MPEG-2 MP@ML の 15[Mbps] で符号化したものである . ここで, 出力レートの範囲が広いほど再量子化ステップ値の分布における偏りは減少し, 提案方式に厳しい条件となる . そこで, 分布 2, 3 では出力レートを MPEG-2 MP@ML の条件を満たす範囲で最大限にとり, この条件下で $M_E(N)$ を低く保ち従来方式に対して十分な利得があることを明らかにする . また, MPEG で規定される量子化ステップ値の定義域から $MQ_{min} = 1$, $MQ_{max} = 31$ である [2] [3] .

再量子化ステップ値禁止領域制御 [34] は, 再量子化処理特有のレート歪み特性において, 符号量はほぼ一定でありながら誤差電力のみが増大することで符号化効率が低下する領域を禁止領域に設定し, 再量子化ステップ値のとりうる値の領域を制限する . 本制御は, 逆量子

化ステップ値 $MQ^{(in)}$ と整数値 k を用いて式 (7-18) により MQ_N を算出する。ただし、上式がイントラ MB、下式がインター MB の場合であり、 $[x]$ は x を越えない最大の整数を表す。

$$\left. \begin{aligned} MQ_N &= 2MQ^{(in)} \cdot k + 1 \\ MQ_N &= \left\lfloor \left(k + \frac{1}{2}\right) MQ^{(in)} \right\rfloor + 1 \end{aligned} \right\} \quad (7-18)$$

従来方式と提案方式の $P_{inc}(N)$ 、 $M_E(N)$ の特性を図 7-4、図 7-5 にそれぞれ示す。

提案方式では、出力数 N が多いほど新たな再量子化/再符号化処理器を追加せずに出力 MB 符号を生成可能な確率は高くなり、 $M_E(N)$ は収束する。これは、小さい N では、 $MQ_1 \sim MQ_{N-1}$ 中に MQ_N と等しい値を含む確率は低いが、 N の増大とともにその確率は高くなるためである。

MPEG では 31 通りの量子化ステップ値が定義されているので、最大でも 31 種類の再量子化ステップ値で出力 MB 符号を生成すればあらゆる出力レートに対応可能である。しかし、図 7-5 は、分布 1 でも 1MB あたり 20 回程度の再量子化処理で十分な出力レート数を提供可能なことを示している。また、分布 2 では図 7-5 から $N = 15$ 付近で $M_E(N)$ が収束し、さらに、分布 3 では $M_E(N)$ が低減する。これらは、選択可能な再量子化ステップ値が制限されることで MQ_N と等しい値が見つかる確率が増加するためである。以上より、提案方式は、十分に少ない再量子化/再符号化処理回数のもとで多くの出力レートに対応可能である。

7.4.3 平均再量子化、再符号化処理回数の極限值

式 (7-7) より $M_E(N)$ は単調増加するので、平均再量子化/再符号化回数の極限值は式 (7-19) に表される。

$$\lim_{N \rightarrow \infty} M_E(N) = \frac{1}{C_p} \quad (7-19)$$

式 (7-19) より、 $M_E(N)$ の極限值は再量子化ステップ値の分布特性によって決定し、容易に見積もれる。分布 1 ~ 分布 3 における極限值は式 (7-20) に算出される。

$$\lim_{N \rightarrow \infty} M_E(N) = \begin{cases} 31.00 & (\text{分布 1}) \\ 4.38 & (\text{分布 2}) \\ 2.86 & (\text{分布 3}) \end{cases} \quad (7-20)$$

式 (7-20) から提案方式の再量子化/再符号化処理器の削減効果に関して、以下の特性を得る。

- 一様分布では、 $\lim_{N \rightarrow \infty} M_E(N) = 31$ として任意の数の出力レートを生成可能
- 実効上は 5 個程度の再量子化出力 MB で任意の出力レート数を得られる。
- 再量子化ステップ値禁止領域制御は $M_E(N)$ を削減する効果を有し、3 個程度の再量子化出力 MB で任意の出力レート数を得られる。

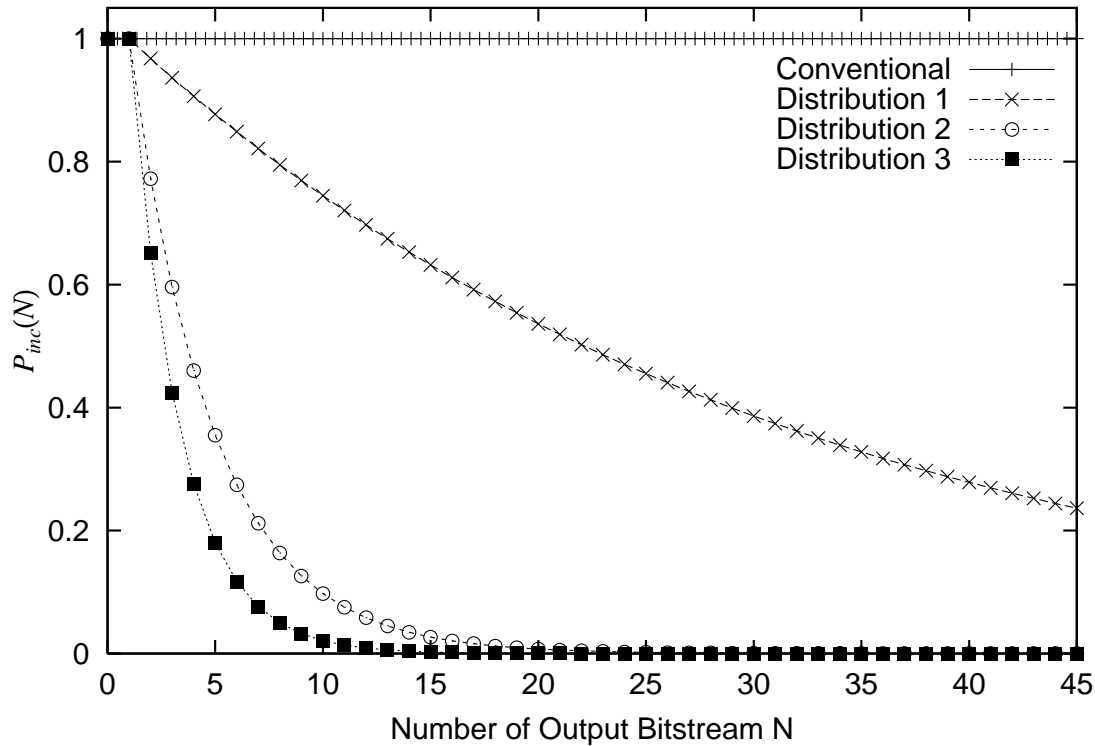


図 7-4: 出力数 N に対する再量子化回数の増加確率 $P_{inc}(N)$ の変動

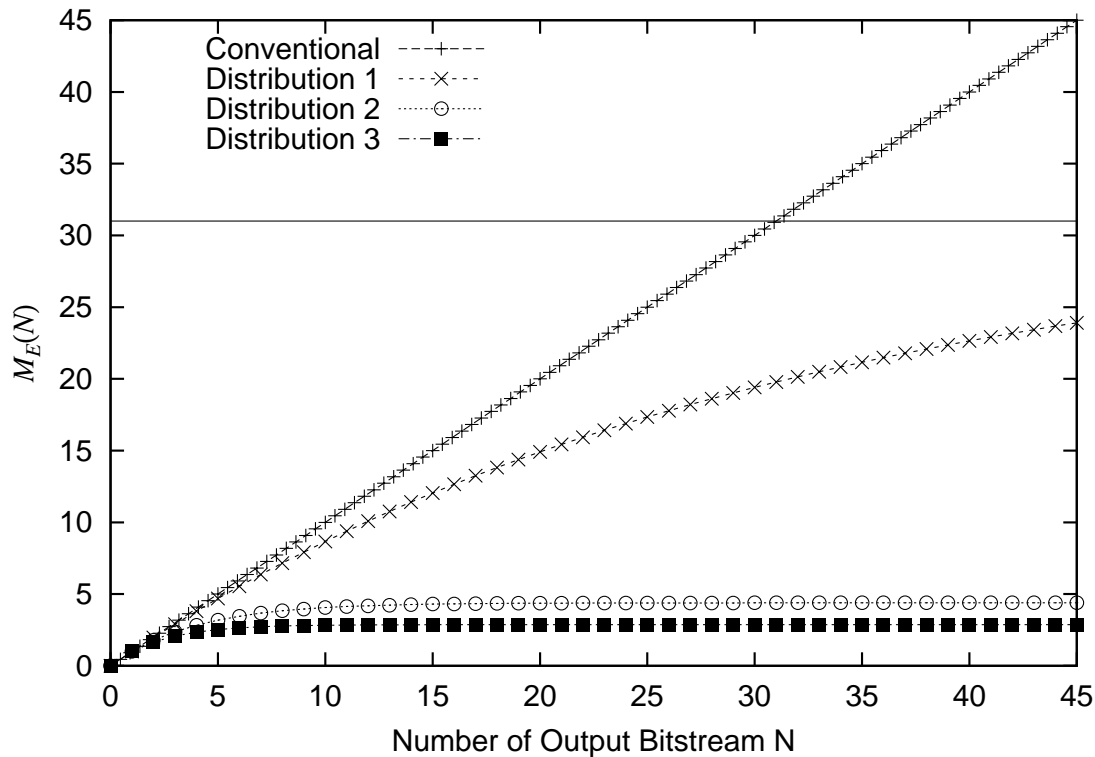


図 7-5: 出力数 N に対する平均再量子化回数 $M_E(N)$ の変動

7.4.4 再量子化，再符号化処理回数の最悪値

本節では，再量子化/再符号化処理回数の最悪値 $M_{worst}(N)$ について考察する．

提案方式において，各量子化制御器より得られる候補再量子化ステップ値 $\{MQ'_1, MQ'_2, \dots, MQ'_N\}$ 内の各要素値が全て異なる値となるときには，提案方式の効果が得られず再量子化/再符号化回数は最悪値をとる．このときには，出力数と同数の再量子化/再符号化処理を行うので最悪値は $M_{worst}(N) = N$ となる．

一方，再量子化ステップ値は $[MQ_{min}, MQ_{max}]$ の範囲内の D 通りの値をとるので，出力数 N がいくら大きくなっても N 通りの再量子化ステップ値の中に含まれる相異なる値の個数は D 個以下に抑えられる．したがって， $N > D$ のときには，最も多くの再量子化/再符号化処理を必要とする場合でもその回数は D 回に飽和して， $M_{worst}(N) = D$ となる．

以上の考察より，再量子化/再符号化回数の最悪値 $M_{worst}(N)$ は， D により場合分けして式 (7.21) のように求められる．

$$M_{worst}(N) = \begin{cases} N & N \leq D \\ D & N > D \end{cases} \quad (7.21)$$

ただし， D は $[MQ_{min}, MQ_{max}]$ の範囲内の整数値の個数として与えられるため，

$$D = MQ_{max} - MQ_{min} + 1 \quad (7.22)$$

である．MPEG の場合には $D = 31$ となる．

7.5 評価実験

提案方式について処理量の観点から評価する．比較対象方式を表 7-1 に示す 4 方式とする．各方式の処理量の評価式を導出し，復号/逆量子化/再量子化/再符号化の各処理単体での処理量を実験的に求めて，処理量算出に必要な基礎実験を行う．次に，導出された評価式を用いて処理量を算出し各方式を比較，評価する．

7.5.1 トランスコード処理コストの導出

本節では，各方式の評価に用いる処理量を定式化する．処理量を，1本のストリーム全体のトランスコード処理に要する時間と定義する．復号/逆量子化/再量子化処理は入力 MB 内の有意係数に対する処理であり，再符号化処理は再量子化出力 MB 内の有意係数に対する処理なので，処理量を MB 内の有意係数の個数と 1 係数あたりの平均処理時間を用いて定式化する．

j 番目 ($1 \leq j \leq MB_{cnt}$) の入力 MB および再量子化出力 MB 内の有意係数の個数をそれぞれ $C_{in}(j)$ ， $C_{out}(j)$ と定義する．また，復号，逆量子化，再量子化，再符号化の各処理における 1 個の有意係数あたりの平均処理時間をそれぞれ K_{VLD} ， K_{IQ} ， K_Q ， K_{VLC} と定義する．1MB あたりの復号，逆量子化，再量子化，再符号化の処理コストをストリーム全体の MB

表 7-1: 比較対象方式

方式	トランスコーダアーキテクチャ	再量子化ステップ 禁止領域制御
1	トランスコーダ並列型 (図 7-1)	—
2	復号/逆量子化器共用型 (図 7-2)	—
3	提案方式 (図 7-3)	OFF
4	提案方式 (図 7-3)	ON

で総和することにより，処理量 T を式 (7-23) に導く．

$$T = \sum_{j=1}^{MB_{cnt}} \left\{ (K_{VLD} + K_{IQ} + K_Q) \cdot C_{in}(j) + K_{VLC} \cdot C_{out}(j) \right\} \quad (7-23)$$

次に，表 7-1 に示す 4 方式について， N 本の出力ストリームへ変換したときの処理量を導出する．ただし，第 n 番出力ストリームの j 番 MB に含まれる有意係数個数を $C_{out}(j, n)$ とする．

1. 方式 1

トランスコーダ並列型アーキテクチャにおける出力数 N に対する処理量を $T_1(N)$ として式 (7-24) に示す．

$$T_1(N) = N \sum_{j=1}^{MB_{cnt}} (K_{VLD} + K_{IQ}) \cdot C_{in}(j) + N \sum_{j=1}^{MB_{cnt}} K_Q \cdot C_{in}(j) + \sum_{j=1}^{MB_{cnt}} \sum_{n=0}^{N-1} K_{VLC} \cdot C_{out}(j, n) \quad (7-24)$$

2. 方式 2

復号/逆量子化器共用型アーキテクチャにおける処理量を $T_2(N)$ と定義する．本方式では，復号/逆量子化器数は出力数に関係なく 1 であるから， $T_2(N)$ は式 (7-24) において右辺第 1 項の係数 N を 1 とした以下の式 (7-25) により定式化される．

$$T_2(N) = \sum_{j=1}^{MB_{cnt}} (K_{VLD} + K_{IQ}) \cdot C_{in}(j) + N \sum_{j=1}^{MB_{cnt}} K_Q \cdot C_{in}(j) + \sum_{j=1}^{MB_{cnt}} \sum_{n=0}^{N-1} K_{VLC} \cdot C_{out}(j, n) \quad (7-25)$$

3. 方式 3，方式 4

提案方式では，再量子化/再符号化回数が MB ごとに変化する． j 番 MB における再量子化/再符号化回数を $M(j)$ として，方式 3，方式 4 の処理量をそれぞれ $T_3(N)$ ， $T_4(N)$

表 7-2: 実験用ストリーム符号化条件

符号化方式	MPEG-2 Video MP@ML
符号化レート	5, 8, 10, 15 [Mbps]
フレームレート	29.97[frame/sec]
符号量制御方式	TM5 方式
GOP 構造	N=15, M=3 (固定)
テストシーケンス	Bus
画像フォーマット	ITU-R BT.601 4:2:0 Format 輝度信号 704pel × 480 line 色差信号 352pel × 240 line
符号化フレーム数	150 [Frame]

と定義する．方式 3 と方式 4 は同一のアーキテクチャであるから，両者共に 式 (7-26) として定式化される．

$$T_3(N) = T_4(N) = \sum_{j=1}^{MB_{cnt}} (K_{VLD} + K_{IQ}) \cdot C_{in}(j) + \sum_{j=1}^{MB_{cnt}} M(j) \cdot K_Q \cdot C_{in}(j) + \sum_{j=1}^{MB_{cnt}} \sum_{m=0}^{M(j)-1} K_{VLC} \cdot C_{out}(j, m) \quad (7-26)$$

7.5.2 トランスコード内部処理時間分布

本節では，復号/逆量子化/再量子化/再符号化の各処理単体における処理量を評価し，シミュレーション実験から K_{VLD} , K_{IQ} , K_Q , K_{VLC} を導出する．

内部処理時間とビットレートの関係

表 7-2 に試験用ビットストリームの符号化条件を示す．実験には，CPU は Pentium III 800[MHz]，メモリは SDRAM 256[MByte]，OS は Vine Linux 2.5 で構成される PC を使用した．ビットストリーム全体 (150 フレーム) にトランスコード処理を行ったときの復号，逆量子化，再量子化，再符号化のそれぞれの処理に要した時間を T_{VLD} , T_{IQ} , T_Q , T_{VLC} [sec] と定義し，処理時間とビットレートとの関係について考察する．

復号，逆量子化，再量子化処理は，入力ストリーム中の係数符号あるいは復号された有意係数に対する処理である．そのため，これらの各処理量は入力ビットストリームのビットレート R_{in} [Mbps] に依存すると考えられる．一方，出力ビットレート R_{out} が高いほど細かな再量子化ステップによって有意係数が多く発生すると考えられるから，再符号化処理量は出力ビットレート R_{out} [Mbps] に依存すると考えられる．そこで， R_{in} に対する T_{VLD} , T_{IQ} , T_Q の関係を図 7-6 に， R_{out} に対する T_{VLC} の特性を図 7-7 にそれぞれ示す．

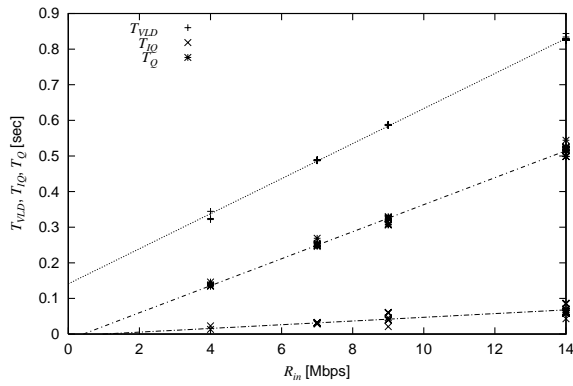
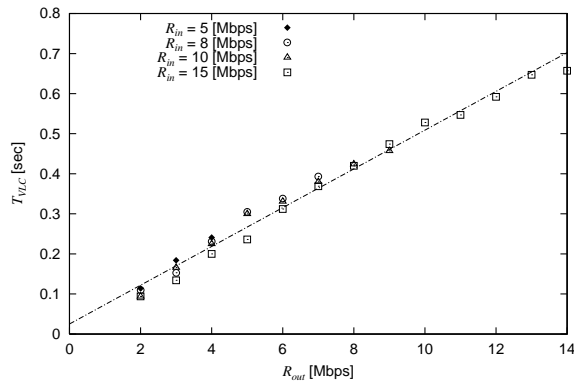
図 7-6: T_{VLD}, T_{IQ}, T_Q と R_{in} の関係図 7-7: T_{VLC} と R_{out} の関係

図 7-6, 図 7-7 より処理時間はビットレートに対して直線的に変化し両者が直線近似可能であることがわかる。これは、ビットレートが高いほど多くの有意係数符号が存在し、有意係数に対する処理自体は個数に関係なく固定的なためと考えられる。ここで、 T_{VLD} および T_{VLC} はオフセットが確認される。逆量子化と再量子化処理は単純な四則演算から構成されるため、処理時間は演算回数から容易に見積もることができる。それに対して、復号および符号化処理は可変長の符号語を扱うための特別な処理を行う。また、MPEG のシンタックスに依存した分岐処理なども伴うため処理時間を単純に見積もることは難しい。そこで、このような実装方法による影響がオフセット値として現れたものと考えられる。

また、図 7-8 に示すようにストリーム中に含まれる有意係数の総数 C_{total} とビットレート R の関係も同様にして直線近似が可能であることがわかる。

以上より、所要処理時間、ビットレート、有意係数の個数のそれぞれが一次関数で近似可能であり、7.5.1 の処理量の評価式の導出において $T_i(N)$ と $C_{in}(j)$, $C_{out}(j)$ を比例関係で近似することの妥当性が明らかとなった。そこで、本性質を利用して比例係数 K を実験的に導出する方法を以下に示す。

比例係数 K_x の導出

上記考察から、処理時間 T_x ($x \in \{VLC, IQ, Q, VLC\}$) とビットレート R の関係は一次関数で近似可能であるので、傾きと切片をそれぞれ α_x, β_x と表して T_x と R の関係を式 (7-27) により表す。

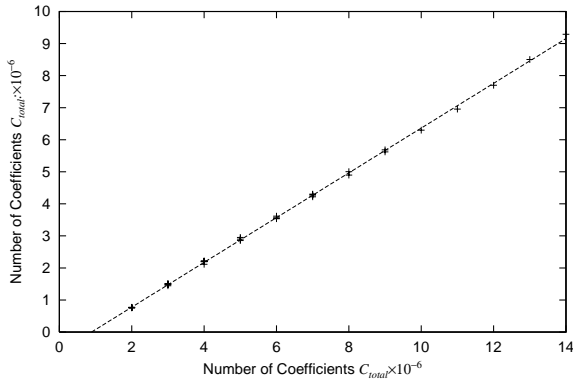
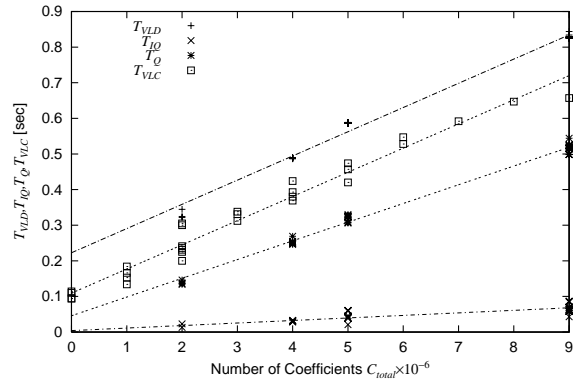
$$T_x = \alpha_x R + \beta_x \quad (7-27)$$

同様にして有意係数の総数を C_{total} と R の関係は、図 7-8 の傾きを γ 、切片を δ とそれぞれ表すことで式 (7-28) のように近似する。

$$C_{total} = \gamma R + \delta \quad (7-28)$$

式 (7-27), 式 (7-28) より T_x と C_{total} は式 (7-29) の関係を持つ。

$$T_x = \frac{\alpha_x}{\gamma} (C_{total} - \delta) + \beta_x \quad (7-29)$$

図 7-8: 有意係数個数 C_{total} と R との関係図 7-9: 処理時間 T_x と C_{total} の関係

R_{in}, R_{out} を媒介変数にして T_x を縦軸に, C_{total} を横軸にプロットして求めた, C_{total} と T_x の特性を図 7-9 に示す. ここで, 変化率 (傾き) α_x/γ は 1 係数あたりの所要時間であり式 (7-23) ~ 式 (7-26) の各式における比例係数 K_x に相当する. したがって, 式 (7-23) ~ 式 (7-26) における比例係数 K_x は α_x と γ を用いて式 (7-30) に表される.

$$K_x = \frac{\alpha_x}{\gamma} \quad (7-30)$$

図 7-9 のプロットから 1 次近似直線の傾きを求めて得た K_x を式 (7-31) に示す.

$$\begin{cases} K_{VLD} = 6.736 \times 10^{-8} \\ K_{IQ} = 1.282 \times 10^{-8} \\ K_Q = 5.299 \times 10^{-8} \\ K_{VLC} = 6.600 \times 10^{-8} \end{cases} \quad (7-31)$$

7.5.3 総処理量に基づく評価実験

7.5.1 で求めた処理量の評価式を用いて各方式のマルチレート変換処理性能を評価する. 試験用ストリームの符号化条件および実験で使用する PC の環境は 7.5.2 と同様である.

出力レート設定条件

出力ストリーム数を N 本として, 出力レート $R_n (n = 0, 1, \dots, N-1)$ を, 4[Mbps] から入力ストリームのレート R_{in} [Mbps] の区間で等間隔にとり, 式 (7-32) のように設定する.

$$R_n = R_0 + \frac{R_{N-1} - R_0}{N-1} \times n \quad [Mbps] \quad (7-32)$$

ただし, $R_0 = 4$, $R_{N-1} = R_{in}$ [Mbps] である.

提案トランスコーダは, 動き補償予測ループを省略した簡略化方式をベースにしているため, 符号化器/復号器間の参照画像の不一致に基因するドリフト誤差による画質低下が発生する. そして, 入出力間のレートの差が大きいほどこの影響は顕著に現れる. 筆者らは, 文

表 7-3: $T_3(N), T_4(N)$ の収束値のシングルレート出力に対する比率

R_{in} [Mbps]	方式 3	方式 4
5	1.76	1.31
8	3.08	1.76
10	3.48	2.03
15	3.92	2.72

献 [33] において MPEG-2 MP@ML 15[Mbps] で符号化されたビットストリームを 5.0, 7.5, 10.0[Mbps] へレート変換した場合における, 動き補償予測ループの ON/OFF 方式間の比較を処理時間と SN 比の観点で行った. その結果, 予測ループ OFF 方式は予測ループ ON 方式に対して, SN 比は 10.0[Mbps] の場合には 0.5~0.6dB, 5.0 および 7.5[Mbps] の場合に 0.9~1.0dB それぞれ低下するが, 処理時間は約 1/30~1/40 となることを明らかにした. 要した処理時間に対する画質の差を考慮すると, 予測ループ OFF 方式は本論文で想定するリアルタイム性が要求されるアプリケーションに対しては有効なアーキテクチャと考えられる.

出力数に対する総処理コスト特性

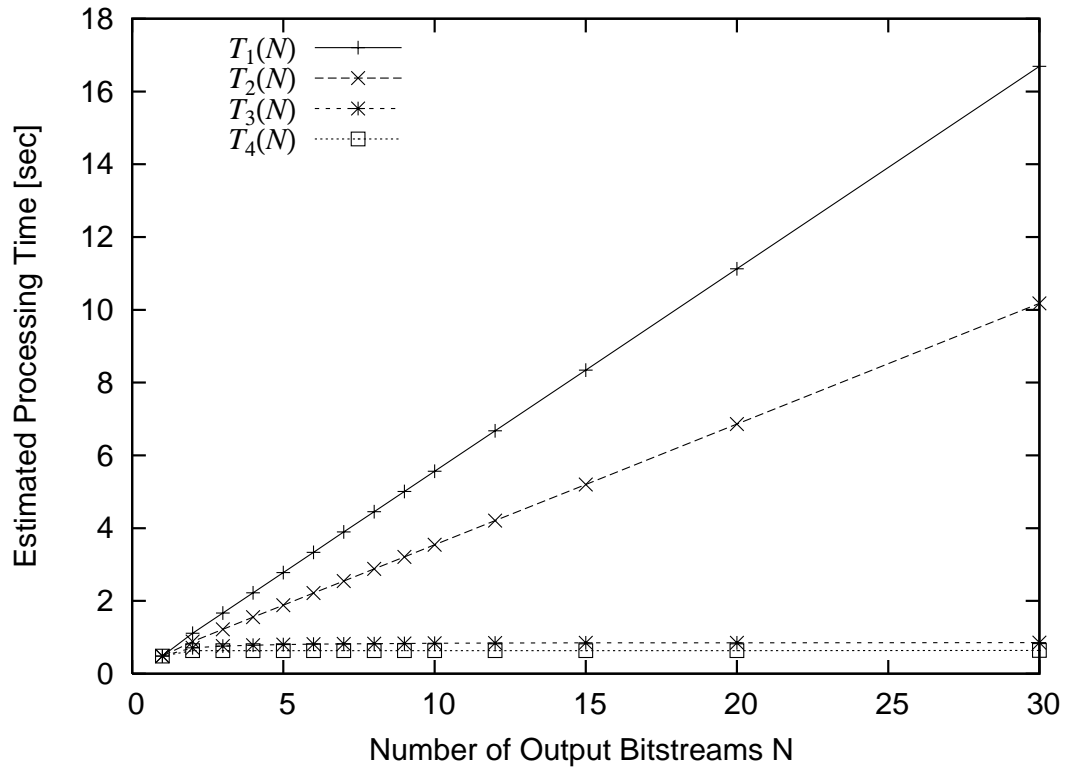
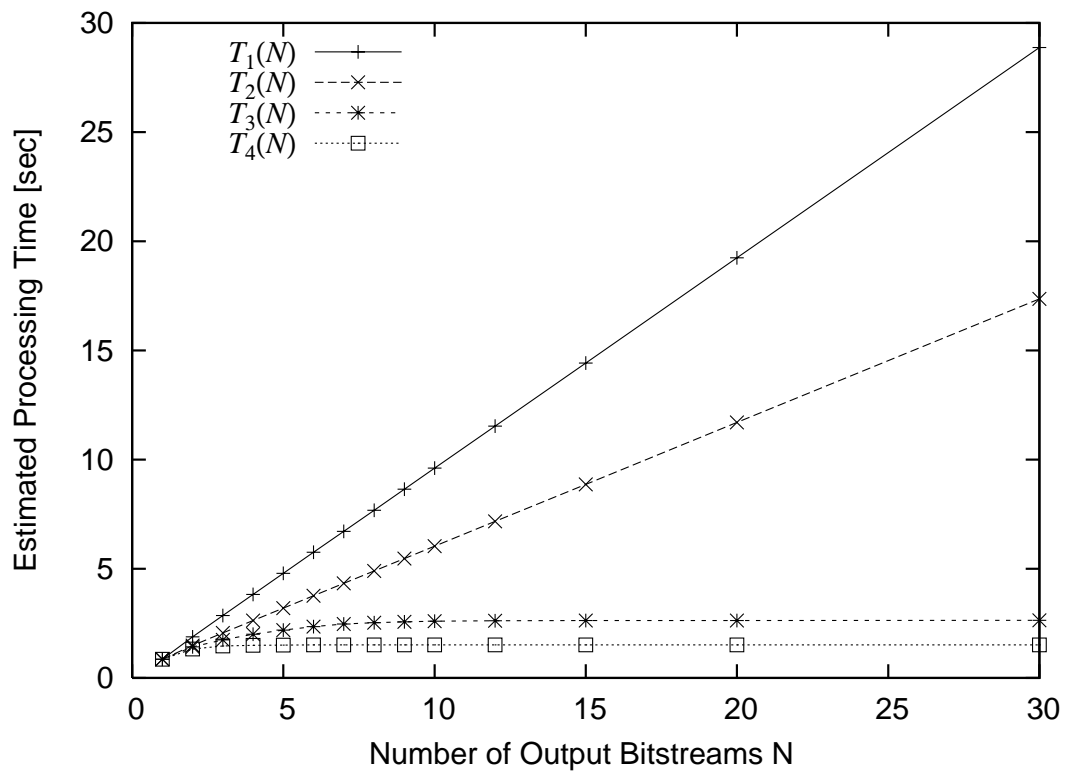
$R_{in}=5,8,10,15$ [Mbps] のそれぞれにおいて, 出力ストリーム数 N に対する $T_i(N)$ ($i = 1, 2, 3, 4$) の特性を図 7-10, 図 7-11 に示す.

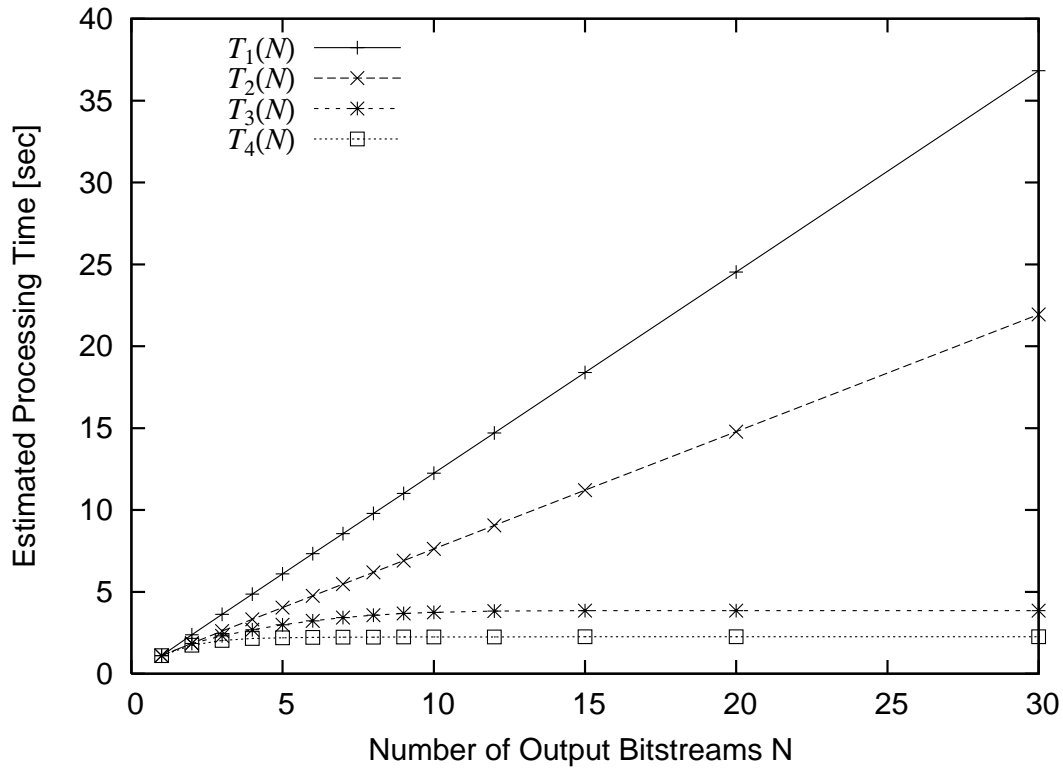
図 7-10, 図 7-11 より, 方式 1, 方式 2 では処理量は出力数 N に比例して増大し, 特に, $R_{in}=8$ [Mbps] の場合に方式 1 では $N = 5$ で 5[sec] を上回りリアルタイム性を保証できなくなる*. 方式 2 では復号, 逆量子化処理がそれぞれ 1 つへ統合されることで, 方式 1 に対して約 50~60% にまで処理量の低減が実現される. その結果, リアルタイム性を維持可能な出力数は $N = 10$ 程度まで拡張される. 方式 1 と方式 2 の比較から, 復号/逆量子化処理器の共用により 2 倍程度の処理スループット向上が確認された.

方式 3, 方式 4 では, 処理量 $T_3(N), T_4(N)$ は共に $N \geq 20$ でほとんど増加しなくなり, 収束する特性を示す. そこで, 処理量が収束したときのシングルレート出力時に対する比率を表 7-3 に示す. 提案方式の処理量は出力数 N には直接比例せず, 再量子化/再符号化処理回数 $M(j)$ に依存した増加特性となる. $M(j)$ は 7.4.2 の $M_E(N)$ と同様の特性であり, 表 7-3 より方式 3 は 3~4 倍, 方式 4 は 2~3 倍の処理量で収束する. これは, 7.4.3 に示した結果と一致する. すなわち, 再量子化/再符号化処理器の共有により処理量は収束し, シングルレート出力トランスコーダの約 2~4 倍の処理量で任意の出力数に対応可能となる.

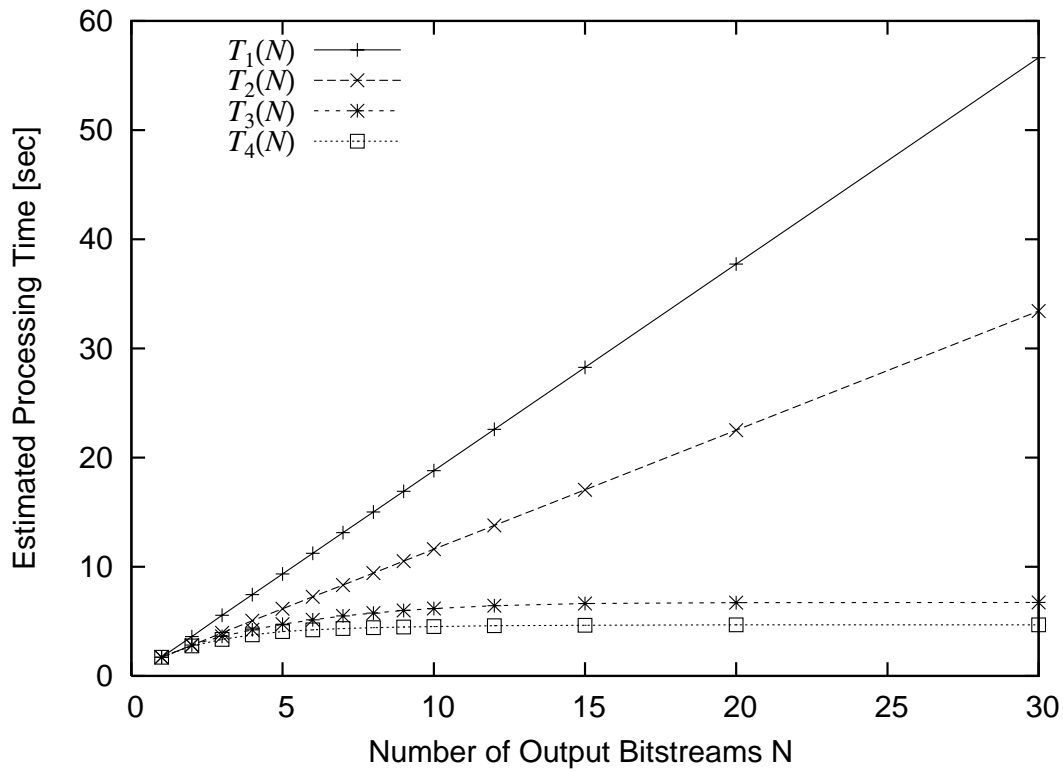
また, 再量子化ステップ値禁止領域制御の再量子化/再符号化回数の低減効果により, 方式 4 ではさらに処理量を削減している. その結果, 本実験で用いた PC 環境での処理量の見積りの上では, 方式 4 は $R_{in}=5, 8, 10$ [Mbps] の場合に出力数をどれだけ大きくとっても

*実験用映像はフレームレートが 29.97[Frame/sec] で 150 フレームで構成されているので $150/29.97 \approx 5$ [sec] 以内で処理が完了すればリアルタイム変換が実現可能である.

(a) $R_{in} = 5$ [Mbps](b) $R_{in} = 8$ [Mbps]図 7-10: 出力数 N に対する総処理コスト $T_i(N)$ の特性 (その 1)



(a) $R_{in} = 10$ [Mbps]



(b) $R_{in} = 15$ [Mbps]

図 7-11: 出力数 N に対する総処理コスト $T_i(N)$ の特性 (その 2)

処理時間を 5[sec] 以内に抑えリアルタイム処理が可能という結果が得られた。つまり、トランスコーダ特有のレート歪み特性に基づく量子化制御が、画質向上だけでなく、提案マルチレート出力トランスコーダの処理量の削減に有効に作用する。

提案方式の性能は出力数 N 以外にも出力レートの範囲や R_n の設定値そのものにも依存し、 R_n に偏りがある場合にはその特性は変化する。出力レートの範囲がせまく各々の R_n が互いに近い値に集中している場合には、各量子化制御器から算出された再量子化ステップ値の分布が局所化されるため、 N が大きくなっても低い $M(j)$ を維持可能であり提案方式の効果は大きい。逆に、 N が低くてもレートの範囲が広い場合には量子化ステップ値の重複が発生しにくくなるので、従来方式との明確な差が現れなくなる。したがって、 N が固定されていても出力レートの範囲が広いほど $M(j)$ が最悪値をとる確率が上昇し、提案方式には厳しい条件となる。

再量子化ステップ値禁止領域制御による処理量削減率評価

再量子化ステップ値禁止領域制御による処理量削減効果について考察する。図 7-10、図 7-11 における $T_4(N)/T_3(N)$ の比率を算出した結果を図 7-12 に示す。

図 7-12 より、方式 4 の処理量は $R_{in} = 5,15$ [Mbps] の場合には方式 3 の約 80% で収束し、 $R_{in} = 8,10$ [Mbps] では最大で約 60% にまで低減する。本量子化制御では、逆量子化ステップ値 $MQ^{(in)}$ から与えられる間隔で再量子化ステップ値の値域を間引くため、選択可能な値の個数は $MQ^{(in)}$ に反比例する [34]。 R_{in} が低いストリームでは平均的に高い $MQ^{(in)}$ が多く分布するので、多くの MB において選択可能な再量子化ステップ値の数が大幅に削減される。その結果、各量子化制御器から算出された候補再量子化値の中に重複が多く含まれることになり、再量子化ステップ値選択処理で少数の量子化器へ束ねられる。ただし、 $R_{in} = 5$ [Mbps] の場合には、 R_n が 4~5[Mbps] の領域に集中しているために、再量子化ステップ値禁止領域制御を適用しなくても多くの再量子化/再符号化回数が削減されている。すなわち、方式 3 で十分な効果が得られているために両方式間の性能差は縮まったものと考えられる。以上より、再量子化ステップ値禁止領域制御による再量子化/再符号化回数の低減効果により最大で約 40% の処理量を削減可能となる。

7.6 むすび

本章では、異なるレートのストリームへの同時複数変換を行うマルチレート変換機能に着目し、出力数に対する処理量を収束させることが可能なマルチレート出力トランスコーダを提案した。そして、再量子化/再符号化処理回数を理論的に解析し、MB あたりの平均再量子化/再符号化処理回数は出力数に対して収束する特性を明らかにした。最後に、評価実験から、提案方式では再量子化/再符号化処理器の共用により負荷の増加を抑制することで処理量を収束させ、結果として、シングルレート出力時の約 3~6 倍の処理量で任意の出力数に対応可能であることを述べた。さらに、再量子化ステップ値禁止領域制御が再量子化/再

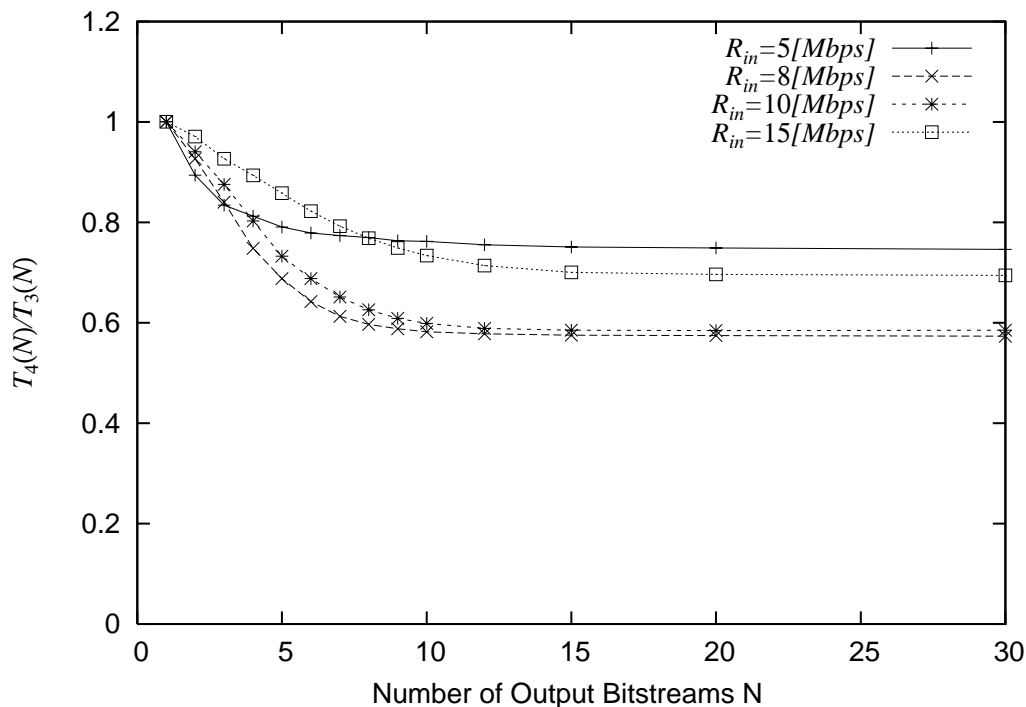


図 7-12: 方式 3,4 の処理量比率 $T_4(N)/T_3(N)$ の特性

符号化器の統合化に効果的に作用し、提案トランスコーダの処理量削減に有効であることを述べた。

ただし、本章では復号、逆量子化、再量子化、再符号化の各処理の処理量に基づき提案方式の有効性を示したが、提案トランスコーダの実装においてはストリームの入出力インタフェースの負荷と出力 MB 符号の選択処理が加わる。よって、これらを考慮に入れた総合的な性能評価は今後の課題である。

本提案方式は、再量子化ステップ値に基づき必要な再量子化回数の削減を行うため、量子化処理をベースにした符号化標準全般に適用可能である。また、提案方式の有効性は、配信サーバや中継ノードへの負担を低減しながら最適な品質のビデオストリームを多くの端末へ同時に提供可能な方式であることを意味する。よって、異種帯域混在型ネットワーク上でのリアルタイム映像配信システムにおける柔軟な QoS 制御方式として実用化可能である。

第8章

結論

本論文では、ブロードバンドネットワーク上での動画像情報の高機能な流通形態の実現を目指して符号化動画像情報の変換技術に着目した。特に、符号変換技術を基盤とした動画像情報の階層的伝送により狭帯域伝送路上で高品質な符号化情報を効率的に伝送するための変換アーキテクチャ及び制御方式について研究した。以下に各章で得られた結論をまとめる。

8.1 各章の結論

第2章では、階層符号化方式による映像のスケーラビリティの実現方式に着目し、トランスコーダの拡張によって階層符号化と同種の機能を実現することで、両手法を融合した新たなスケーラブル映像通信方式を提案した。主な結果は以下の通りである。

1. 階層符号化方式の特徴および実現される機能を整理した。
2. 階層符号化方式には、符号化/復号器の処理構造の複雑化、汎用性および階層化に伴う画質劣化に関する課題があることを述べた。
3. トランスコーダにおける変換前後の差分映像信号を付加信号として利用することで、変換後映像を基本階層、差分信号を高位階層とした階層符号化機能を、トランスコーダの拡張によって実現する方法を提案した。
4. 本手法が非階層のビットストリームに対する変換処理であることに起因して、階層化のための専用の処理構造を持たない汎用の符号化器により階層符号化機能を実現可能という特徴を有することを述べた。
5. 非階層 MPEG ビットストリームを階層構造へ変換する処理器をビットストリーム分離器と定義した。
6. 階層符号化情報を非階層 MPEG ビットストリームへ変換する処理器をビットストリーム合成器と定義した。
7. ビットストリーム合成器を復号器の前段に接続することで、汎用の復号器のもとで階層符号化情報の復号処理が可能であることを述べた。
8. 従来からの符号化/復号器による階層符号化方式として SNR スケーラビリティ、更新スケーラビリティについて取り上げ、これらの各種方式と提案方式の特徴を比較した。

9. SNR スケーラビリティでは、高位階層復号時に非階層と同等の画質が保証されないのに対し、提案方式では、階層化処理の前段で非階層符号化情報を生成するので、階層化処理を行っても非階層符号化時と同等の画質を保証可能であることを述べた。
10. 復号側においては、提案方式と更新スケーラビリティと全く同様の機能を実現する。しかし、符号化側での処理量、汎用性に関しては明確な差があることを述べた。
11. 以上より、提案階層符号化方式の特徴を以下にまとめる。
 - 汎用の符号化器/復号器により実現可能
 - 非階層時と同等の画質を保証

第 3 章では、第 2 章で検討した符号化器/復号器の外部機能による階層符号化機能の実現する具体的な符号化方式を提案した。主な結果は以下の通りである

1. DCT 係数領域内再量子化処理によりビットレート削減を行う MPEG トランスコーダに、再量子化前後間の差分信号の符号化処理器を追加することで階層符号化機能を実現する方法について検討した
2. ビットレート削減後の符号化情報を基本階層、差分量子化係数信号の符号化情報を高位階層として 2 階層の階層符号化方式を実現する方法を検討した。
3. 再量子化演算において、トランスコーダ特有のレート歪み特性を考慮した符号化効率の劣化領域回避のための再量子化ステップ値禁止領域制御の適用を前提としたときの、再量子化前後間での係数値の変化の特性について考察した。
4. 再量子化ステップ値算出に再量子化ステップ値禁止領域制御を適用することで、イントラモード、インターモードともに、切り上げ演算によって再量子化前と等しい係数値をとなることはなく、再量子化演算後には必ず係数値は変動し差分信号を発生させることを明らかにした。
5. デッドゾーン領域に関する考察より、再量子化係数値が 0 になる入力係数値の範囲は、再量子化ステップ値算出制御に用いるパラメータにより一意に決定されることを明らかにした。
6. 再量子化後が非零係数となる場合には、再量子化係数ブロック内の非零係数をジグザグスキャン順に参照することで、差分量子化係数値に対応する再量子化係数値のブロック内の位置が特定可能であり、ブロック内位置を表現するための符号が不要であることを述べた。
7. 上記考察に基づき、入力係数を再量子化後に零係数になるものとそれ以外として分類して、それぞれを別々の方法により符号化する方法を提案した。
8. 再量子化後に零係数となる場合には、入力係数ブロック内の連続する零係数の個数に関する 2 次元ランレングスにより符号化を行う方法を述べた。
9. 再量子化後が非零係数となる場合の差分信号については係数値のみを符号化対象とし、再量子化処理における除算で発生した剰余値を符号化する方法を述べた。

10. 基本階層と高位階層の合計符号量に関する評価実験により、単純にブロック間差分をとり2次元ランレングス符号化した場合には、符号量が大きく増大するのに対して、提案方式では階層化処理に伴う符号量の増加を回避可能であり、符号化効率の点で有効性があることを示した。
11. DCT係数をラプラス分布としてモデル化して提案方式の符号化効率について理論的に考察し、MPEG-2での2次元ランレングス符号化方式と比較した。
12. 本考察により、提案方式は、全く同一の係数情報をMPEG-2での2次元ランレングス符号化方式よりも少ない符号量で符号化可能な再圧縮の効果が得られ、符号化効率の有効性を理論的に明らかにした。特に、単にトランスコーダによるスケラビリティの実現手段としてだけでなく、DCTと量子化をベースとした各種符号化方式における可変長符号化方式の新たな手法としても有用な方式であることを述べた。

第4章では、第3章で提案した2階層の階層符号変換方式を拡張し、高位階層の符号化信号に任意の品質を取得可能な情報構造を持たせることで多階層/多品質の品質取得に対応する階層符号変換方式を提案した。主な結果は以下の通りである。

1. 高位階層符号を、内部的に階層構造を持たせた部分階層符号により構成し、部分階層符号の再送による映像情報の順次伝送方式について述べた。
2. 本方式は、受信側では基本的な品質を持った映像情報のみを先行受信して大まかな映像シーンを早期に把握する。その後、伝送帯域、要求伝送時間に応じて高位階層符号を数回に分けて行う。受信側では受信した部分階層成分を逐次合成して階層を積み上げていくことで受信映像の高画質化を行う。これにより、受信可能な情報量のみを伝送可能となり、高位階層の伝送にスケラビリティを実現する。
3. 再量子化ステップ値禁止領域制御を前提として、2種類のステップ値で再量子化された再量子化出力ブロック間での有意係数の分布の違いについて考察した。その結果、有意係数分布の差は再量子化ステップ値算出のための制御パラメータである整数値と関連性を有することを明らかにした。すなわち、再量子化後が零係数となる入力有意係数を係数レベル値ごとに分類して再量子化ブロックに加算して行くと、中間の再量子化ステップで再量子化したブロックと同等の有意係数分布を持つ量子化係数ブロックが生成可能であることを明らかにした。
4. 上記考察に基づき、階層構造を実現するための量子化係数情報の構成方法として、再量子化前後間の差分係数情報を係数レベル値により階層化して、それぞれをランレングス符号化することで、部分的に復号/合成して中間的な品質の取得を可能とする高位階層符号を符号化方式を提案した。
5. 具体的な差分信号の符号化/復号アルゴリズムについて説明し、係数レベル値ごとに別々にランレベル系列が構成されることで、復号/合成時に任意のランレベル系列を選択可能となり、これにより、再量子化前と後との中間的な係数ブロックを合成可能となることを示した。

6. 基本階層と高位階層の総符号量に関する評価実験より，2 階層方式とほぼ同等の総符号量が達成されていることから，提案符号化方式は，高位階層の多階層化による符号化効率の変動はほとんど無く 2 階層方式と同等の符号化効率を実現可能な方式あることを明らかにした．
7. 中間映像の合成実験から，再量子化演算によりほぼ同等の品質を生成した場合と比較して中間合成映像の符号量は符号量が大きく増大することから，中間合成映像の符号化効率に課題があることを述べた．

第 5 章では，基本階層との差分信号を，互いに独立した情報構造を有する階層符号として符号化処理をすることで，階層間の依存関係を排した多階層の階層符号化変換方式を提案した．主な結果は以下の通りである．

1. 従来の多階層の階層符号化方式では，中間の階層が欠落した場合それより上の全ての階層の符号化情報が無効となることから，各階層において階層に応じたエラー対策機構を必要とし，階層数の増加に伴う優先制御の複雑化の課題があることを述べた．
2. ゼロ係数へ再量子化される入力量子化係数を抽出した量子化係数列において，係数のインデックス値を高位階層の数で除算した剰余値で各係数を分類する方式を提案した．
3. 分類後の量子化係数列をそれぞれ独立にスキャン，エントロピ符号化することで，互いに依存しない情報構造を持った高位階層符号化情報を出力する方式を提案した．
4. 評価実験から，階層間の依存構造の排除により若干の符号化効率の低下が確認されたが，大幅な性能低下は回避可能であることを示した．
5. 複数の蓄積，伝送デバイスを利用することで効率性と損失の耐性を両立した映像蓄積，伝送方式への応用の可能性について言及し，提案方式の実用面での有効性を述べた．

第 6 章では，階層間のビットレート配分制御の柔軟性の特徴に基づいて，提案階層符号変換における変換制御方式に着目した．特に，伝送帯域の時間的変動に対して適応的に変換ビットレート制御を行う動的制御方式を提案した．主な結果は以下の通りである．

1. 伝送プロトコルとして TCP/IP を想定し，伝送帯域の不足時に働く TCP/IP の輻輳制御に起因するトランスコード処理速度への影響に着目した．
2. 処理速度の評価基準としてストリーム中の映像と音声の同期制御のための基準時刻を与える SCR 信号に着目し，トランスコード処理の経過時間を SCR を用いて評価することで，伝送帯域の推定を行う方法を検討した．
3. 基礎実験により，SCR 復号値と処理経過時間は一定の比率を保ちながら増加し，両者の比率 (処理経過時間増加率) は伝送帯域と変換レートの関数で定式化可能であることを示した．
4. トランスコード処理経過時間と SCR 復号値を利用して伝送帯域に応じたビットレート制御を行うための，トランスコード目標変換レート制御方式を提案した．

5. 提案制御方式は、(1) 処理経過時間増加率導出制御、(2) 処理経過時間増加率収束目標値制御、(3) 変換レート更新制御、の3つの Step で構成されることを述べた。
6. Step 1 では、処理経過時間と SCR 復号値を基準に算出されたストリーム時刻の観測点からの最小二乗近似により、処理経過時間増加率を求める制御を述べた。
7. Step 2 では、処理経過時間とストリーム時刻の比較により、Step 3 における収束目標値を求める方法を述べた。ストリーム時刻が先行する場合にはビットレートを上昇させる制御を行い、処理経過時間が先行する場合には符号量を抑制する制御を行う。
8. Step 3 では、基礎実験で定式化された特性関数を用いて処理経過時間増加率から伝送帯域を推定し、処理経過時間増加率を、伝送帯域と変換レートの均衡点へ所定の時定数で指数関数的に収束させるように、目標変換レートの制御を行う方法を述べた。
9. シミュレーション実験から、提案方式により伝送帯域が未知の場合でも目標変換レートを適切な値へ制御することが可能であり、伝送帯域の変動にも追従可能な制御であることを示した。
10. Step 3 で設定された時定数により提案方式の特性は大きく異なり、定常時の安定性と帯域変動に対する応答性能のトレードオフがあることを述べた。すなわち、時定数を大きくとりフィードバックが弱いほど高い安定性を有するが応答速度が遅く、逆に時定数を小さな値へ設定すると制御の感度が高くなり、定常時の目標変換レートの変動幅が増大するが応答性能が向上することを示した。
11. トランスコーダの処理の経過時間と入力ストリーム中の時刻の増加比率を用いることで、伝送帯域が変動する場合においても目標変換レートをそれに追従して制御可能であることを示した。

第7章では、1本の入力ストリームを品質の異なる複数のストリームへ同時に変換を行う1入力N出力型の変換(マルチレート変換機能)を実現するマルチレート変換トランスコーダに着目し、出力ストリーム数の増加に対しても処理量を収束可能な高い処理効率を実現可能なトランスコーダアーキテクチャを提案した。主な結果は以下の通りである。

1. マルチレート変換の従来方式として、1入力1出力型のトランスコーダ(シングルレート出力トランスコーダ)を出力数だけ並列動作させる方式(シングルレート出力トランスコーダ並列型方式)、復号および逆量子化処理を共通化した方式(復号/逆量子化器共用型方式)の2方式について取り上げた。
2. 処理コストに関する考察より、従来方式では処理量は $O(N)$ で増加するので、トランスコーダを実装するハードウェア性能が向上しても同時出力可能なストリームの数には限界が存在することを述べた。
3. DCT 係数領域内再量子化処理によるトランスコーダではMBの再符号化出力は再量子化ステップ値により一意に決定可能であることから、1つの入力MBに対する再符号化結果は有限であり、同一の再量子化/再符号化結果を異なるストリームへ出力可能な場合があることを述べた。

4. 各出力レートごとに算出される再量子化ステップ値の中から重複を除外して相異なる値のみを抽出して再量子化器で使用する再量子化ステップ値を選択することで、共通な再量子化/再符号化処理を共用した再量子化/再符号化演算回数を削減可能なマルチレート変換トランスコーダのアーキテクチャを提案した。
5. 提案トランスコーダでは再量子化/再符号化処理の回数が性能に大きく影響する。そこで、出力数に対する平均再量子化/再符号化処理回数の特性について評価したところ、出力数がどれだけ増加しても実効上は 1MB あたり 5 回程度の再量子化/再符号化処理で任意の出力レートに対応可能であることを明らかにした。
6. トランスコーダ特有のレート歪み特性に基づく再量子化ステップ値の禁止領域制御(再量子化ステップ値禁止領域制御)の適用によって、再量子化/再符号化処理回数は 3 回程度へ削減可能であることを明らかにした。
7. 最悪値に関する考察より、再量子化/再符号化処理回数は、最悪の場合でも出力数と同数あるいは量子化ステップ値の定義個数以下に抑えることが可能であることを明らかにした。
8. ストリーム全体の有意係数の個数、復号/逆量子化/再量子化/符号化の各処理における 1 係数あたりの平均処理時間および再量子化/再符号化処理回数により処理量を定式化して、提案方式の処理量の特性を評価した。その結果、提案方式における処理量は、出力数が 20 以上の領域でほとんど増加しなくなり、シングルレート出力の 2~4 倍の処理量で収束することを明らかにした。
9. 再量子化ステップ値禁止領域制御の適用による再量子化/再符号化回数の低減効果が最大で約 40% の処理量を削減可能であることを示した。

8.2 階層型動画像符号変換方式に関する総合評価

本研究では、ビットレート変換を行うトランスコーダを拡張して階層符号化機能を実現する動画像符号変換方式について検討した。特に、ビットレート変換時の再量子化演算の特性に着目して、(i) 基本階層と高位階層から構成される 2 階層の階層符号変換方式、(ii) 高位階層符号化情報の内部構造を階層化して任意の品質を持つ合成映像を取得の可能とした方式、(iii) 独立した情報構造を持つ N 通りの階層符号化情報を用いた高位階層間の依存性を排除した方式、の 3 方式に関する階層化方式を提案した。その結果、非階層符号化時と同等の符号化効率を達成可能な階層化方式を実現し、符号化効率の課題に対する解を示した。各種方式の要点を表 8-1 に、性能比較を表 8-2 にそれぞれ示す。ただし、表 8-2 は、本論文の各章の評価・検討結果を整理したものであり、定性的な比較のための指標と考えて頂きたい。

本論文で提案する階層化手法は、ビットストリーム生成と別の段階で階層化を行うことで、専用機能を持たない汎用の符号化/復号器のもとで階層型動画像伝送を実現する。そのため、符号化および復号処理に高い汎用性を有する点を特徴とする。ただし、高位階層情報の復号には復号器の前段に非階層符号への復元を行うビットストリーム合成処理器を必要とする。しかし、復号器に専用機能を要求する従来の方式と比較すると十分に高い利用価値を

表 8-1: 各種階層符号変換方式の特徴のまとめ

方式	階層数	品質数	各階層の符号化方法		特徴
			基本階層	高位階層	
SNR スケーラビリティ	2	2	入力信号を粗く量子化	入力信号と基本階層との差分信号を量子化	受信者を主体とした階層選択による品質選択性を実現
ストリーム分離合成 (第 3 章)	2	2	復号量子化係数を粗く再量子化	再量子化前後での量子化係数の差分を符号化	汎用符号化器のもとで 2 階層の階層符号化を実現
多階層符号変換 (第 4 章)	2	N	復号量子化係数を粗く再量子化	量子化係数の差分を係数値レベル値ごとに別々にスキランして可変長符号化	任意の品質を取得可能
階層間独立方式 (第 5 章)	N	N	復号量子化係数を粗く再量子化	量子化係数の差分を復号ブロック内位置を階層数で除算した余りにより分離し、それぞれを別々にスキランし可変長符号化	階層同士の依存構造を排除

持つと考えられる。また、符号化処理段階で品質が決定されることはないために、伝送チャネルの帯域状況に応じて柔軟にビットレートの制御が可能でありトランスコーダが持つ柔軟性の特徴を継承する。

次に、変換処理方式に着目すると、各方式に共通して復号器と階層型符号化器の従属接続方式 [44] [45] [46] ではなく、DCT 係数領域内再量子化方式による密結合型トランスコーダアーキテクチャを拡張した復号量子化係数のハンドリングにより階層符号変換を実現する点を特徴とする。基本階層では動き補償処理を省略し予測誤差信号の再計算を省略しているので復号画像にはドリフト誤差による画質劣化が発生する。本特徴により、基本画像の画質においては提案方式は不利となる。しかし、階層構造への変換処理は量子化係数領域で完結し、非階層の DCT 係数信号に完全一致することを前提とした非可逆な方式となっている。これにより、高位階層映像の画質は非階層符号化時と同等となることが保証される。特に、階層符号変換時に非階層符号化処理時と同等の符号化効率を実現可能な点は、従来の階層型の MC-DCT ループによる符号化アーキテクチャに対する明確な性能差となる。

符号化器の設計技術、制御技術は、標準仕様が勧告化された当時と比較して飛躍的に向上

表 8-2: 各種階層符号変換方式の性能比較と応用分野

方式	符号化効率 (画質)		符号化器 汎用性	復号器 汎用性		品質 選択性	柔軟性	応用分野
	基本 階層	高位 階層		基本 階層	高位 階層			
SNR スケーラビリティ		×	×	×	×		×	デジタル放送の受信強度改善策策
ストリーム分離合成 (第 3 章)								IP 網を用いた映像監視システム
多階層符号変換 (第 4 章)								ニュース映像編集システム
階層間独立方式 (第 5 章)								パケットロスをする IP 伝送

しており、MPEG-2 では 20[Mbit/sec] 以上のビットレートが必要と言われていた HDTV 信号を 10[Mbit/sec] 前後のビットレートで符号化処理が可能な MPEG-2 符号化制御技術が実現されている [86]。符号化/復号器との外部機能により実現し、なおかつ階層化に伴う符号化効率の低下を回避可能であることから、十分に設計された高画質性/高符号化能率性を有する符号化器に対しても、その性能を損なうことなく階層型動画像伝送を実現可能となる。したがって、本研究で検討した量子化係数領域内での変換処理による階層化方式は、昨今の符号化器の性能向上に対して親和性の高い方式といえる。

本研究は、ビットレート変換を行うトランスコーダを起点として、従来よりスケーラブル符号化方式 (SNR スケーラビリティ) として検討されている階層符号化機能を実現したものである。これは、図 8-1 に示す両種の機能を継承した従来とは別の分類体系 (図中の網掛け部分) に位置する。各章で議論した一連の方式は全てビットストリーム変換処理を基点としたものであることから、本論文の成果を、符号化パラメータの変換、符号化フォーマット変換に続く第 3 の形態の変換処理体系として位置付ける。

8.3 本論文の総括および今後の課題

最後に、本研究の成果を総括し、将来の動画像流通における基盤技術としての適用可能性を展望する。

動画像信号の符号化/伝送技術は、計算機の処理能力の向上、伝送環境の広帯域化などのハードウェア環境の高性能化を源動力として発達していった。プラットフォームの高性能化はさらなる動画像流通への需要を喚起し、流通コンテンツの増加、高精細化、高ビットレート化をもたらす。そのため、広帯域網が普及してもハンドリングすべき情報量も同時に増大し、帯域資源が相対的に不足する状況は今後も継続すると考えられる。その一方で、高精細化/高品質性への要求は高まるばかりであり、映像情報のハイビジョン化や MPEG-4 AVC/H.264、Windows Media Video 9 などの最新の高能率符号化技術により高画質化が実

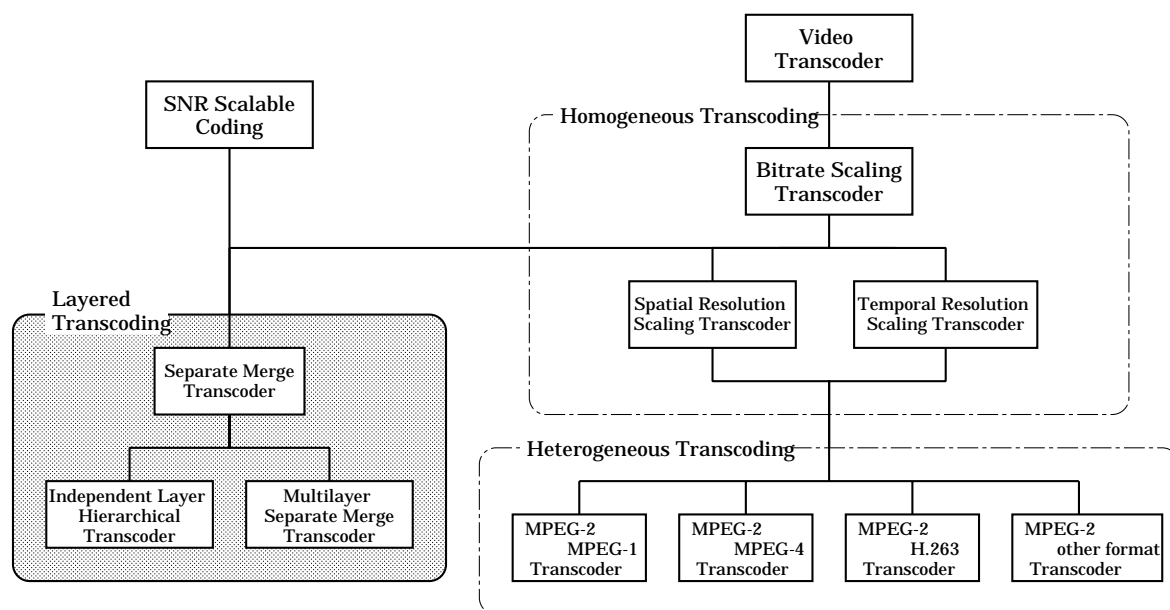


図 8-1: 階層符号変換方式の研究の位置付け

現される反面，コンテンツ自身の増大によりこの状況をさらに加速させられると思われる．計算資源および帯域資源の相対的な不足に対して，情報量のスケール変換を行うトランスコーダや低レート符号化による解は，品質を制限する以上は高画質化への要求に対する根本的な解とはなり得にくい．本論文では，伝送帯域の有効利用と高品質化という相反する要求に対して符号変換処理のアプローチによる解を与えることを目指したものであり，この成果はブロードバンド環境での映像流通における符号化処理技術として有効的と考えている．複雑化する周辺環境の中で符号化動画のハンドリングに関する問題は今後も継続的に議論すべき課題であり，そのためには本論文の成果だけでなく，帯域資源の配分，レート制御などの次世代の映像伝送ネットワークシステム的设计からメディア情報の圧縮符号化技術までを含めた多角的側面からの総合的な検討が必要となるであろう．

謝辞

本研究は早稲田大学理工学部コンピュータネットワーク学科(旧 電子・情報通信学科)及び早稲田大学大学院国際情報通信研究科 富永 英義教授の御指導のもとに進められました。富永教授には、素晴らしい研究の機会と充実した環境を与えて頂き、日頃から終始熱心に御指導を賜り、また温かく御激励頂きました。ここに改めて厚く御礼申し上げます。

また、渡辺 裕 教授、亀山 涉 教授、花村 剛 客員助教授(メディアグルー, 工学博士)をはじめとする早稲田大学大学院国際情報通信研究科の諸先生方、及び浅谷 耕一 教授(工学院大学)、小宮 一三 教授(神奈川工科大学)の各先生方には、筆者の博士後期課程在学中に適切な御指導を賜り、また本論文をまとめるに際して懇切な御教示を賜りました。ここに深く感謝の意を表します。

特に、花村 剛 助教授には、筆者の博士後期課程在学中に熱心な御指導、御鞭撻を頂き、本論文をまとめるにあたり構想段階から懇切な御指導を頂きました。また、本研究の全般に渡って有益な御討論、御助言および御協力頂き、各章のアルゴリズム開発、実験等に貴重な御助力頂きました。重ねて深く感謝の意を表します。

笠井 裕之氏(現 NTT ドコモ, 工学博士)には、筆者の博士後期課程在学中に熱心な御指導、御鞭撻を頂き、研究室配属時より有益な御教示、御討論を頂き、本論文をまとめるにあたり懇切な御助言を頂きました。また、第7章における討論、実験等に御助力頂きました。ここに深く感謝の意を表します。

本研究における第5章は、涌井 道子女史(修士課程2年)の平成15年度修士論文の研究を筆者と共同で行ったものであり、研究を進める上で有益な御討論、実験等に御助力を頂きました。また、アルゴリズム開発およびシミュレーション実験のためのソフトウェア製作・実験環境の整備等に御助力頂きました。ここに厚く御礼申し上げます。

本研究における第4章は、角田 福美女史(修士課程1年)と筆者との共同で行った平成14年度の卒業論文の研究を基礎として発展させたものです。本研究を進める上で有益な御討論、御助力を頂きました。ここに厚く御礼申し上げます。

本研究における第6章は、中平 航太氏(現、インクリメントP)と筆者との共同で行った平成14年度修士論文の研究を基礎として、高岡 崇靖氏(学部4年)の平成15年度卒業論文の研究として発展させたものであり、筑波 健史氏(学部4年)、松尾 翔平氏(学部4年)らとともに、研究を進める上で有益な御討論、御助言を頂き、シミュレーション実験等に御助力頂きました。ここに厚く御礼申し上げます。

以上の方々をはじめとして、研究だけでなく様々な活動を共にし、日頃より多方面にわたり筆者を支えて下さいました榊 直昭氏(メディアグルー)をはじめとするメディアグルーの方々および富永研究室の皆様ならびに関係者各位に心から感謝申し上げます。

2004年3月

永吉 功

参考文献

- [1] 横山 徹: “ブロードバンドへむけたストリーミング技術の動向と課題”, 映像情報メディア学会誌, Vol. 57, No.5, pp. 555–556 (2003).
- [2] ISO/IEC 11172-2, IS: “Coding of moving pictures and associated audio for digital storage media at a up to about 1.5 Mbits/s” (1993).
- [3] ISO/IEC 13818-2, IS: “Generic Coding of Moving Picture and Associated Audio, Recommendation H.262” (1995).
- [4] ISO/IEC 14496-2, IS: “Generic Coding of Audio-Visual Objects, Part-2 Visual” (2001).
- [5] Joint Video Team (JVT) of ISO/IEC and ITU-T, IS: “Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 — ISO/IEC 14496-10 Advanced Video Coding)” (2003).
- [6] Windows Media 9 Series Audio and Video Codecs: “<http://www.microsoft.com/windows/windowsmedia/9series/codecs.aspx>”.
- [7] DigitalStream-USA: “http://www.ds-usa.com/xvd_ov.html”.
- [8] The On2 VP6 Video Codec: “<http://www.on2.com/vp6.php3>”.
- [9] Windows Media Technologies: “<http://www.microsoft.com/windows/windowsmedia/default.aspx>”.
- [10] RealNetworks: “<http://www.realnetworks.com/products/index.html>”.
- [11] 柳原 広昌, 菅野 勝: “コンテンツオーサリング技術”, 映像情報メディア学会誌, Vol. 57, No.5, pp. 553–554 (2003).
- [12] 笠井 裕之: “スケーラブル伝送実現のための動画像符号化制御方式の研究”, 早稲田大学大学院理工学研究科 博士学位論文 (2000).
- [13] 小林 直樹: “映像配信技術の動向”, 映像情報メディア冬期大会 講演予稿集, pp. 39–44 (2000).
- [14] 小貝 昌彦, 市川 俊一, 中里 秀則, 横田 英俊, 浦野 義頼, 富永 英義: “Diff-serv におけるマルチキャスト映像配信に関する検討”, 信学技報, IN2000-51 (2000).
- [15] 三宅 洋一郎, 中里 秀則, 浦野 義頼, 富永 英義: “Diff-serv を用いた範囲制御マルチキャストに関する検討”, 信学技報, IN2000-51 (2003).
- [16] 中島 康之, 米山 暁夫, 菅野 勝, 柳原 広昌: “MPEG オーディオデータの符号化レートスケーラビリティ”, 信学論 D-II, J82-D-II, No.9, pp. 1355–1364 (1999).

- [17] A. Vetro, C. Christopoulos and H. Sun: “Video Transcoding Architectures and Techniques: An Overview”, *IEEE Signal Processing Magazine*, pp. 18–29 (2003).
- [18] 児玉 明: “機能化画像の構造化方式とスケーラビリティに関する研究”, 早稲田大学大学院理工学研究科 博士学位論文 (1997).
- [19] DivX.com: “<http://www.divx.com/>”.
- [20] Canopus X-Transcoder: “http://www.zdnet.co.jp/news/0306/18/nj00_mtvfx.html”.
- [21] I/O Data GV-XVD/PCI: “<http://www.iodata.jp/prod/multimedia/tv/2003/gv-xvdpci/>”.
- [22] T.Shanableh and M.Ghanbari: “Heterogeneous Video Transcoding to Lower Spatio-Temporal Resolutions and Different Encoding Formats”, *IEEE Trans. Multimedia*, **Vol.2**, pp. 101–110 (2000).
- [23] J.Xin, M.T.Sun, K.Chun and B.S.Choi: “Motion Re-estimation for HDTV to SDTV Transcoding”, *Proc. IEEE Int. Symp. Circuits and Systems*, Vol. 4, pp. 715–718 (2002).
- [24] 阿久津 隆史, 木全 英明, 清水 淳, 八島 由幸, 小林 直樹: “低ビットレート用 MPEG-2 リアルタイムソフトウェアトランスコーダ「Trampeg」の開発”, *信学論 D-II*, **J84-D-II**, No.6, pp. 1084–1093 (2001).
- [25] 辻 裕之, 阪谷 徹, 八島 由幸, 小林 直樹: “DCT ドメインにおける MPEG トランスコードの検討”, *画像符号化シンポジウム資料*, PCSJ2000 P-O3.02, pp. 129–130 (2000).
- [26] G.Keesman, R.Hellinghuizen, F.Hoeksema and G.Heideman: “Transcoding of MPEG Bitstreams”, *Signal Processing:Image Communication*, **Vol.8**, (1996).
- [27] H. Sun, W. Kwok and J. W.Zdepski: “Architecture for MPEG Compressed Bitstream Scaling”, *IEEE Trans. on Circuits & Syst. for Video Technology*, **Vol.6**, No.2, pp. 191–199 (1996).
- [28] Y.Nakajima, H.Hori and T.Katoh: “Rate conversion of MPEG coded video by re-quantization process”, *Proc. IEEE Int. Conf. Image Processing*, pp. 408–411 (1995).
- [29] N.Bjork and C.Christopoulos: “Transcoder architectures for video coding”, *IEEE Trans. Consumer Electron.*, **Vol.44**, pp. 88–98 (1998).
- [30] J.Wang and S.Yu: “Dynamic rate scaling of coded digital video for IVOD applications”, *IEEE Trans. Consumer Electron.*, **Vol.44**, pp. 743–749 (1998).
- [31] P. A. A. Assuncao and M. Ghanbari: “A Frequency-Domain Video Transcoder for Dynamic Bit-Rate Reduction of MPEG-2 Bit Streams”, *IEEE Trans. on Circuits & Syst. for Video Technology*, **Vol.8**, No.8, pp. 953–967 (1998).

- [32] 笠井 裕之, 富永 英義, 花村 剛, 亀山 涉: “低遅延 MPEG-2 ビデオトランスコーダ符号量制御方式”, 信学論 B, **J83-B**, No.2, pp. 151–164 (2000).
- [33] 笠井 裕之, 永吉 功, 亀山 涉, 富永 英義: “ドリフト誤差伝搬を考慮した MPEG ビットストリームレートスケーリング符号量制御方式”, 信学論 D-II, **J83-D-II**, No.2, pp. 509–524 (2000).
- [34] 笠井 裕之, 永吉 功, 花村 剛, 亀山 涉, 富永 英義: “MPEG-2 ビデオトランスコーダのための再量子化パラメータ禁止領域制御方式”, 信学論 B, **J85-B**, No.2, pp. 278–286 (2002).
- [35] W.Zhu, K.H.Yang and M.J.Beacken: “CIF-to-QCIF video bitstream down-conversion in the DCT domain”, Bell Labs Tech. Journal, **Vol.3**, No.3, pp. 21–29 (1998).
- [36] J. Xin, M. T. Sun, B. S. Choi and K. W. Chun: “An HDTV-to-SDTV Transcoding”, IEEE Trans. on Circuits & Syst. for Video Technology, **Vol.12**, No.11, pp. 998–1008 (2002).
- [37] P.Yin, M.Wu and B.Lui: “Video transcoding by reducing spatial resolution”, Proc. IEEE Int. Conf. Image Processing, Vol. Vol. 1, pp. 717–720 (2000).
- [38] 清水 淳, 嵯峨田 淳, 渡辺 裕, 小林 直樹: “参照フレーム変更によるビットストリームスケーリング”, 信学論 D-II, **J83-D-II**, No.3, pp. 967–976 (2000).
- [39] J. Hwang, T. Wu and C. Lin: “Dynamic frame-skipping in video transcoding”, Proc. IEEE Workshop Multimedia Signal Processing, pp. 616–621 (1998).
- [40] K.T.Fung, Y.L.Chan and W.C.Siu: “New architecture for dynamic frame-skipping transcoder”, IEEE Trans. on Image Processing, **Vol.11**, pp. 886–900 (2002).
- [41] A.Vetro, P.Yin, B.Liu and H.Sun: “Reduced spatio-temporal transcoding using an intra-refresh technique”, Proc. IEEE Int. Symp. Circuits and Systems, Vol. Vol.4, pp. 723–726 (2002).
- [42] M.J.Chen, M.C.Chu and C.W.Pan: “Efficient motion estimation algorithm for reduced frame-rate video transcoder”, IEEE Trans. on Circuits Syst. for Video Technology, **Vol.12**, pp. 269–275 (2002).
- [43] N.Feamster and S.We: “An MPEG-2 to H.263 transcoder”, Proc. SPIE Conf. Voice, Video Data Communications (1999).
- [44] Y. Lin, C. Wang, T.Chiang, A.Vetro and H.Sun: “Efficient FGS-to-single layer transcoding”, Proc. IEEE Int. Conf. Consumer Electron., pp. 134–135 (2002).
- [45] Y. Tan and Y. Liang: “Methods and needs for transcoding MPEG-4 fine granularity

- scalability video”, Proc. IEEE Int. Symp. Circuits and Systems, Vol. Vol. 4, pp. 719–722 (2002).
- [46] E.Barrau: “MPEG video transcoding to a fine-granular scalable format”, Proc. IEEE Int. Conf. Image Processing, Vol. Vol. 1, pp. 717–720 (2002).
- [47] H.Kato, H.Yanagihara, Y.Nakajima and Y.Hatori: “A fast motion estimation algorithm for DV to MPEG-2 conversion”, Proc. IEEE Int. Conf. Consumer Electronics., pp. 140–141 (2002).
- [48] W.Lin, D.Bushmitch, R.Mudumbai and Y.Wang: “Design and implementation of a high-quality DV50-MPEG-2 software transcoder”, Proc. IEEE Int. Conf. Consumer Electronics., pp. 142–143 (2002).
- [49] J.Youn, M.T.Sun and C.W.Lin: “Motion vector refinement for high performance transcoding”, IEEE Trans. Multimedia, **Vol.1**, pp. 30–40 (1999).
- [50] 笠井 裕之, M. Nilsson, 富永 英義: “Audiovisual Service への Mobile Access のための Multimedia Transcoding System の研究開発”, 信学技報, CS2001-114, IE2001-141, pp. 19–24 (2001).
- [51] 笠井 裕之, 富永 英義: “H.323-SIP ゲートウェイシステムの研究開発”, 情処研報, AVM36-9, pp. 53–58 (2002).
- [52] H. Kasai, M. Nilsson, T. Jebb, M. Whybray and H. Tominaga: “The Development of a Multimedia Transcoding System for Mobile Access to Video Conferencing”, IEICE Trans. Comms., **E85-B**, No.10, pp. 2171–2181 (2002).
- [53] O. Werner: “Requantization for Transcoding of MPEG-2 Intraframes”, IEEE Trans. on Image Processing, **Vol.8**, No.2, pp. 179–191 (1999).
- [54] S.Kadono, M.Etoh and N.Yokoya: “Rationality of Restricted Re-Quantization for Efficient MPEG Transcoding”, IEEE Int. Conf. Image Processing ICIP2000, MP-11-1885 (2000).
- [55] A. Leventer and M. Porat: “On Bit Allocation in Video Coding and Transcoding”, Proc. IEE Int. Conf. Visual Information Engineering, pp. 254–257 (2003).
- [56] D. Morrison: “Video Transcoders with Low Delay”, IEICE Trans. Comms., **E80-B**, No.6, pp. 963–969 (1997).
- [57] L. Wang, A. Luthra and B. Eifrig: “Rate Control for MPEG Transcoders”, IEEE Trans. on Circuits & Syst. for Video Technology, **Vol.11**, No.2, pp. 222–234 (2001).
- [58] 永吉 功, 笠井 裕之, 花村 剛, 亀山 渉, 富永 英義: “削減符号量による歪み増加特性を

- 考慮した MPEG-2 ビデオトランスコーダ再量子化符号量制御方式に関する基礎検討”, 映情学誌, **Vol.55**, No.12, pp. 1669–1674 (2001).
- [59] Jeongnam Youn and Jun Xin and Ming-Ting Sun and Ya-Qin Zhang: “Video Transcoding for Multiple Clients”, Visual Commun. and Image Processing VCIP2000, Vol.4067-08 (2000).
- [60] 花村 剛, 富永 英義: “スケーラブルビデオ符号化”, テレビ学技報, ITE17-57, pp. 25–30 (1994).
- [61] 映像情報メディア学会: “総合マルチメディア選書 MPEG”, オーム社 (1997).
- [62] 牛久保 宏徳, 大津 徹, 石田 則明, 佐々木 信幸, 三好 匠, 浅谷 耕一, 富永 英義: “デジタル衛星放送を考慮した圧縮階層符号化方式の検討”, 信学技報, SANE2001-131, SAT2001-153 (2002).
- [63] 笠井 裕之, 児玉 明, 富永 英義: “階層符号量配分率の動的更新制御による動画像階層符号化符号量制御方式”, 信学論 B, **J82-B**, No.8, pp. 1475–1488 (1999).
- [64] W. Li: “Overview of Fine Granularity Scalability in MPEG-4 Video Standard”, IEEE Trans. on Circuits and Syst. for Video Technology, **Vol.11**, No.3, pp. 301–317 (2001).
- [65] X. Sun, F. Wu, S. Li, W. Gao and Y.-Q. Zhang: “Macroblock-Based Progressive Fine Granularity Scalable Video Coding”, Proc. IEEE Int. Conf. Multimedia Expo., pp. 461–464 (2001).
- [66] C. Buchner, T. Stockhammer, D. Marpe, G. Blattermann and G. Heising: “Efficient Fine Granular Scalable Video Coding”, Proc. IEEE Int. Conf. Image Processing, pp. 997–1000 (2001).
- [67] W.-S. Peng and Y.-K. Chen: “Mode-Adaptive Fine Granularity Scalability”, Proc. IEEE Int. Conf. Image Processing, pp. 993–996 (2001).
- [68] 児玉 明, 花村 剛, 富永 英義: “Result of an SNR Scalability Experiment and Syntax Modifications to it”, ISO-IEC JTC1/SC2/WG11, MPEG 92/269 in RIO (1992).
- [69] 児玉 明, 花村 剛, 富永 英義: “Simulation Results of Core Experiments on Frequency Scalability - Layer Rate Control(Slice vs MB Rate Control)”, ISO-IEC JTC1/SC2/WG11, MPEG 92/439 in TARRY (1992).
- [70] 花村 剛, 亀山 涉, 富永 英義: “Hierarchical Coding Scheme of Video Signal with Scalability and Compatibility”, EURASIP:Signal Processing, IMAGE Communication, **Vol.5**, No.1, pp. 159–184 (1993).
- [71] 児玉 明, 富永 英義: “動画像ビットストリーム更新スケーラビリティとその評価”, 信学論 B-I, **J80-B-I**, No.2 (1997).

- [72] 宮地 悟史: “携帯電話による家庭内映像監視システム”, 映像情報メディア学会誌, Vol. 57, No.9, pp. 1073–1075 (2003).
- [73] 小沢 慎治: “サベーランスの現状と今後”, 映像情報メディア学会誌, Vol. 57, No.9, pp. 1056–1057 (2003).
- [74] 西村 敏, 笠井 裕之, 高屋 和幸, 亀山 渉, 榊 直昭, 花村 剛, 富永 英義: “リアルタイム MPEG-2 ビデオトランスコーダソフトウェアの開発”, 情処研報, AVM25-5, pp. 25–30 (1999).
- [75] I. Nagayoshi, H. Kasai, T. Hanamura, W. Kameyama and H. Tominaga: “A Study on the Rate Control Scheme for MPEG-2 Video Transcoder considering Re-Quantization Error”, IEICE Technical Report, IE-99-75, pp. 9–16 (1999).
- [76] 笠井 裕之, 永吉 功, 花村 剛, 亀山 渉, 富永 英義: “再量子化特性を利用した MPEG-2 トランスコーダ量子化制御方式の検討 (1)”, 信学春季全大, D-11-59 (2000).
- [77] 角野 眞也, 栄藤 稔, 横矢 直和: “MPEG-2 再符号化における再量子化誤差の検討”, 信学技報, IE-99-32 (1999).
- [78] ISO/IEC 15444-1, FDIS: “JPEG2000 Part-1 Core Coding System” (2000).
- [79] 市川 俊一, 浅谷 耕一, 富永 英義: “誤り訂正符号を用いた損失耐性のある階層化マルチキャストレート制御”, 信学論 B, J86-B, No.2, pp. 187–199 (2003).
- [80] 酒澤 茂之, 滝嶋 康弘, 和田 正裕: “ビデオマルチキャストにおける符号化伝送制御方式”, 映像情報メディア学会誌, Vol. 52, No.6, pp. 863–870 (1998).
- [81] 涌井 道子, 永吉 功, 花村 剛, 笠井 裕之, 富永 英義: “時間的変動を有するネットワーク適用のためのトランスコーダ符号量制御方式に関する基礎検討”, 信学技報, DSP2002-1, IE2002-1, MI2002-1, pp. 1–6 (2002).
- [82] V.Jacobson: “Congestion Avoidance and Control”, Proc. SIGCOMM Synposium on Communications Architectures and Protocols, pp. 314–329 (1988).
- [83] W. Sterens: “詳解 TCP/IP Vol.1”, ピアソン・エデュケーション (2000).
- [84] S.Floyd and T.Henderson: “The NewReno Modification to TCP’s Fast Recovery Algorithm”, Network Working Group RFC2018 (1996).
- [85] L.S.Brakmo and L.L.Peterson: “TCP Vegas : End to end congestion avoidance on a global internet”, IEEE J.Selected Areas in Commun., Vol.13, No.8, pp. 1465–1480 (1995).
- [86] 現行DVDへの記録を可能とするHDTV超圧縮技術: “<http://www.mitsubishielectric.co.jp/news-data/2002/release/0206-a6-1.htm>”.

図一覧

1-1	ビデオトランスコーダの分類体系	5
2-1	トランスコーダを用いた階層符号化への変換	19
3-1	ストリーム分離器ブロック図	25
3-2	ストリーム合成器ブロック図	26
3-3	係数ブロックの变化成分情報の符号化方法	31
3-4	Bitsream C のビットレート R_C の変化 (入力符号化レート 15[Mbps])	34
3-5	ビットレート合計値 $R_B + R_C$ の変化 (入力符号化レート 15[Mbps])	36
3-6	ビットレート合計値 $R_B + R_C$ の変化 (入力符号化レート 10[Mbps])	37
3-7	ストリーム分離器二段接続実験図	38
3-8	ストリーム分離器二段接続構成におけるビットレート合計値 $R_C + R_D + R_E$ の変化	39
3-9	m に対する各種符号量の変化 ($Q_1 = 2$)	43
3-10	(R_1, m) に対する出力総符号量の変化	46
4-1	多階層構造を有する高位階層符号による階層型動画像伝送	48
4-2	係数差分符号化情報の生成	53
4-3	分離処理フロー	55
4-4	合成方式	56
4-5	合成処理フロー	58
4-6	差分ストリームの抽出による部分合成	60
4-7	R_B に対する合計ビットレート $R_B + R_C$ の変化 (入力符号化レート 15[Mbit/sec])	62
4-8	R_B に対する合計ビットレート $R_B + R_C$ の変化 (入力符号化レート 10[Mbit/sec])	62
4-9	高位階層数 m に対する階層符号化情報総符号量特性 (上) $MQ_1 = 2$, (中) $MQ_1 = 4$, (下) $MQ_1 = 6$	64
4-10	中間合成ビットストリーム符号化特性, (上) $MQ_1 = 2$, (中) $MQ_1 = 4$, (下) $MQ_1 = 6$	65
5-1	独立性を有する階層符号への変換方法	69
5-2	合成方法	71
5-3	高位階層損失時の合成処理	72
5-4	高位階層数 N に対する階層符号総符号量の特性	74

5-5	中間合成出力符号量と画質の関係 (入力符号化レート 10[Mbit/sec])	75
5-6	中間合成出力符号量と画質の関係 (入力符号化レート 15[Mbit/sec])	75
6-1	実験系の構成	79
6-2	実験用ビットストリーム作成のための多重化ヘッダ構造の変換	81
6-3	トランスコード処理時間と SCR 復号値の関係	82
6-4	処理経過時間増加率 Y と R_N / R_T の関係	84
6-5	トランスコード処理経過時間と SCR 復号値の関係 (伝送速度変動あり)	85
6-6	最小二乗近似による $Y(n)$ の導出	86
6-7	処理経過時間増加率を用いた目標レートの算出	88
6-8	目標変換ビットレート R_T の変化 (方式 I, その 1)	89
6-9	目標変換ビットレート R_T の変化 (方式 I, その 2)	90
6-10	処理経過時間増加率 $Y(n)$ の変動 (方式 I, その 1)	91
6-11	処理経過時間増加率 $Y(n)$ の変動 (方式 I, その 2)	92
6-12	各ピクチャの復号バッファ内滞留時間 $T_B(k)$ の変動 (その 1)	96
6-13	各ピクチャの復号バッファ内滞留時間 $T_B(k)$ の変動 (その 2)	97
6-14	過渡状態における R_N と R_T の不均衡領域	98
7-1	シングルレート出力トランスコーダ並列型	102
7-2	復号/逆量子化器共用型	102
7-3	提案マルチレート出力トランスコーダブロック図	104
7-4	出力数 N に対する再量子化回数の増加確率 $P_{inc}(N)$ の変動	110
7-5	出力数 N に対する平均再量子化回数 $M_E(N)$ の変動	110
7-6	T_{VLD}, T_{IQ}, T_Q と R_{in} の関係	114
7-7	T_{VLC} と R_{out} の関係	114
7-8	有意係数個数 C_{total} と R との関係	115
7-9	処理時間 T_x と C_{total} の関係	115
7-10	出力数 N に対する総処理コスト $T_i(N)$ の特性 (その 1)	117
7-11	出力数 N に対する総処理コスト $T_i(N)$ の特性 (その 2)	118
7-12	方式 3,4 の処理量比率 $T_4(N)/T_3(N)$ の特性	120
8-1	階層符号変換方式の研究の位置付け	129

表一覧

1-1	トランスコーダの変換機能とアーキテクチャ	6
1-2	トランスコーダアーキテクチャの特性比較	8
3-1	比較対象方式	33
3-2	実験用ビットストリーム符号化条件	35
4-1	実験用ビットストリーム符号化条件	61
4-2	比較対象方式	61
4-3	MQ_1 に対する m_{max} の関係	62
5-1	実験用ビットストリーム符号化条件	73
5-2	高位階層符号の合成順序	74
6-1	実験用 PC の諸元	80
6-2	実験用ビットストリーム符号化条件	80
7-1	比較対象方式	112
7-2	実験用ストリーム符号化条件	113
7-3	$T_3(N), T_4(N)$ の収束値のシングルレート出力に対する比率	116
8-1	各種階層符号変換方式の特徴のまとめ	127
8-2	各種階層符号変換方式の性能比較と応用分野	128