

Grid Computing における MPEG-4 Encoder の並列化に関する 検討

† 柏木 文徳 † 三浦 康之 † 渡辺 重佳
† 湘南工科大学

現在の PC は高性能・低価格化が進み、マルチコア CPU が登場している。さらに、個人や企業で利用していない PC を所有していることが珍しくないものとなっている。そこで、それらを並列処理に利用することで、多大な時間を要する動画の圧縮処理を高速化する。また、並列処理では、ネットワークを介して複数のコンピュータを接続し、仮想的に高性能なコンピュータを作る Grid Computing が注目されている。Grid Computing では、接続されるコンピュータに性能の違いがあり、利用していない PC を接続することも可能となる。しかし、MPEG-4 のような依存関係が多く発生する処理では、Grid Computing で単純に並列処理を行うと性能の低いコンピュータがボトルネックになる可能性がある。動画像圧縮の並列化の方法として、MPEG-4 の B-VOP の特性を利用した「コンピュータ間並列化」と空間分割による「コンピュータ内並列化」を Grid Computing において組み合わせる方法を検討した。

キーワード：グリッドコンピューティング, MPEG-4, 並列処理

A Parallelization of MPEG-4 Encoder on Grid Computing

† Fuminori Kashiwagi † Yasuyuki Miura † Shigeyoshi Watanabe
† Shonan Institute of Technology

Abstract

In recent days, the cost of PCs (personal computers) have become down in price and become lower and the performance has become higher. Moreover, Multi-Core CPUs has spread and some people have a number of computers. The grid computer can be constructed by using those computers. However, low performance computer might become bottleneck of the encoder because MPEG-4 Encoding has many dependencies. So, we considered the method to combine the inter-PC parallelization B-VOP parallelization and intra-PC parallelization.

Keyword : Grid Computing, MPEG-4, Parallel Processing

1. はじめに

現在の PC は高性能・低価格化が進み、マルチコアの CPU も登場している。さらに個人や企業で利用していない PC を所有していることが珍しくないものとなっている。

この利用していない PC やマルチコアの CPU を並列処理に利用することで、高性能な PC でも多大な時間を要する動画の圧縮処理を高速化する。

また、並列処理の分野では、ネットワークを介して複数のコンピュータを結ぶことで仮想的に高性能なコンピュータを構築する Grid Computing が注目されている。Grid 環境を用いて、MPEG-4 Encoder の並列化を行う場合、接続されているコンピュータの性能に差があることが多く、そのようなケースでは、性能の低いコンピュータがボトルネックになることが多い。

その問題を解決する方法として B-VOP の特性を利用した並列化を行うことで依存関係が多く発生する動画の圧縮処理に性能の低いコンピュータを組み込むことを検討した

2. MPEG-4

MPEG-4 の動画の圧縮は I-VOP、P-VOP、B-VOP の 3 つから成っており、I-VOP は単体で符号化、復号化ができる VOP であり、P-VOP は時間的に自身よりも前にある I-VOP または P-VOP に基づいて順方向予測符号化される。B-VOP は、自身よりも前にある I-VOP または P-VOP と自身よりも後にある I-VOP または P-VOP に基づいて双方向予測される。

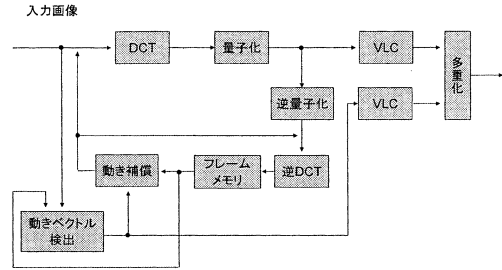


図 1 MPEG-4 のブロック図

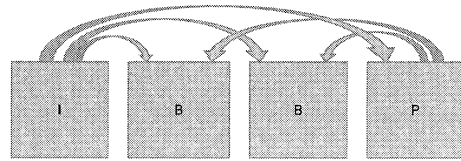


図 2 各 VOP の依存状況

3. 並列化環境

3.1. Cluster

図 3 に、クラスタの構成例を示す。現在あるスーパーコンピュータはいわゆるクラスタ型であることが多い。クラスタは、大抵の場合、同程度の性能のコンピュータで構築され、ネットワークも専用の通信網を構築している場合などは、通信にかかる速度も考慮しやすく、動画の圧縮処理のように依存関係が多く含まれるものでは有利である。しかし、今回は、利用されていない PC を利用するなど性能に差があることを前提としているのでこの方式は用いない。

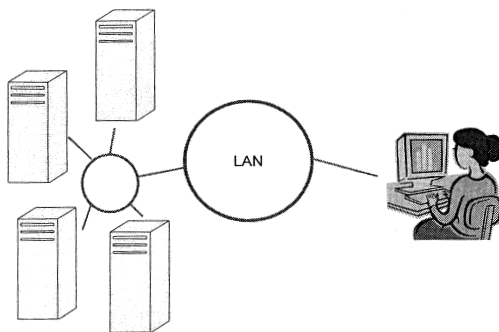


図 3 Cluster の構成例

3.2. Grid Computing

図 4 に Grid Computing の構成例を示す。Grid Computing では、接続されるコンピュータが多く、それぞれのコンピュータの性能に差があることが多く、通信速度についても考慮しにくく、動画の圧縮処理に用いるにはあまり向いていないと考えられる。しかし、今回は性能に差があるコンピュータや CPU がマルチコアであるかどうか等、環境に差があることを前提として検討を行うため、この方式を選択した。

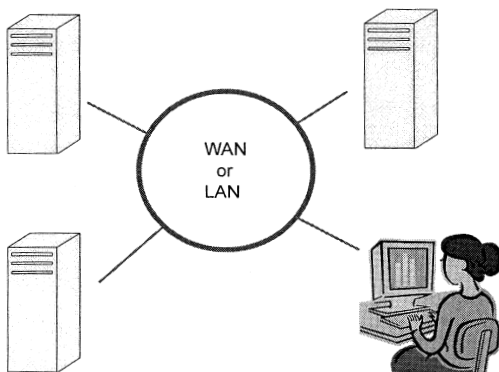


図 4 Grid Computing の構成例

4. 並列化方法

4.1. コンピュータ内並列化

コンピュータ内の並列化は、マルチコアの CPU を利用して行う。ここでは、通信が同じコンピュータ内で行われるため、依存関係のある処理でもあまり問題にならないということもあり、処理ブロック内の並列化と単純な空間分割による並列化を行い比較した。

処理ブロック内の並列化は、DCT ブロック内の並列化を行い、単純な空間分割による並列化は、横方向に短冊状に 4 分割した。

コンピュータ内並列化における 1 フレームあたりの処理時間を図 5 に示す[4],[5]。この結果から、空間分割による並列化の速度向上率が高く、実装も比較的容易な事からこの方式を採用した。

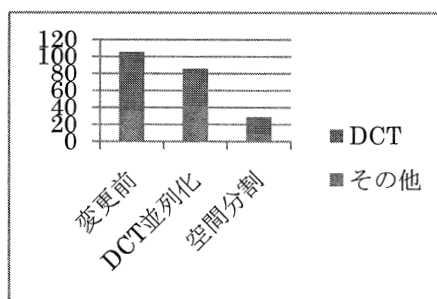


図 5 コンピュータ内並列化

4.2 コンピュータ間並列化

4.2.1. GOP 並列化

MPEG 規格では、I-VOP から成る一塊の VOP のグループを GOP(Group of Picture) と呼び、扱っている。これを 4 台のコンピュータで並列に処理させる。この方法は単純なだけに効率が良いが、圧縮終盤の GOP が性能の低いマシンに割り当てられると圧

縮処理の終了時間が遅くなるという欠点がある。

また、各 GOP の処理時間は変わらないため、リアルタイム処理にはあまり向いていない。

4.2.2. B-VOP 並列化

図 6 に B-VOP 並列化の流れを示す。B-VOP 並列化は、B-VOP が直前と直後の I-VOP または P-VOP に基づいて符号化を行い、自身は他の I-VOP、P-VOP、B-VOP に影響しないため、この部分を並列化することで、MPEG-4 の圧縮処理の依存関係に影響されることがなく並列化ができる。

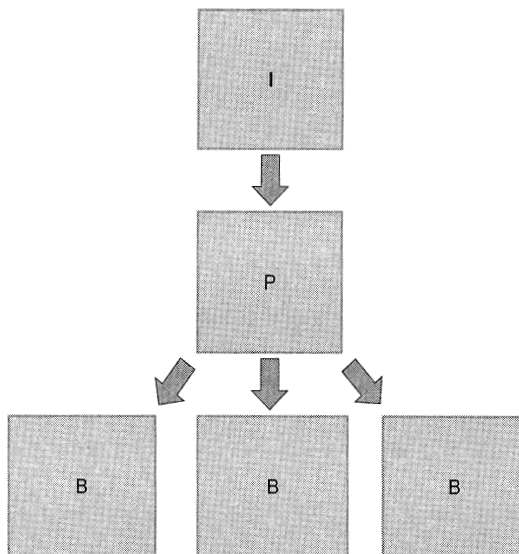


図 6 B-VOP の並列化方法

図 7 に実際の B-VOP 並列化の流れを、図 8 に B-VOP 並列化の例を示す。実際には、次の P-VOP とも並列処理できるので実際にはこれも並列処理が可能となる。また、B-VOP 並列化のスケジュール例は図 8 のよ

うになる。しかし、I-VOP または P-VOP と P-VOP または I-VOP の間に入る B-VOP の数が増えるほど、動きベクトルの計算などにかかる時間が増加、それにより P-VOP の処理に要する時間が増加したりする。さらに、B-VOP を大量に入れることによる画質の大幅な劣化など問題もあるため、B-VOP の特性を利用した並列化には問題も多い。

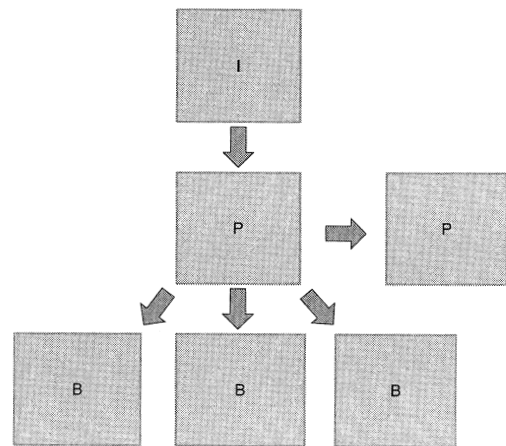


図 7 実際の B-VOP 並列化

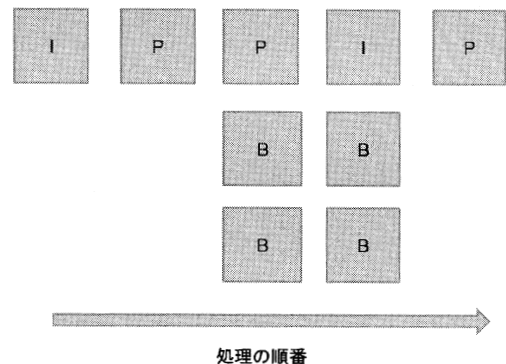


図 8 B-VOP 並列化の例

5. 実験

5.1. 実験方法

GOP 並列化と B-VOP 並列化についてシミュレータを作成して所要時間を調べる。シミュレータは、Core2 Quad2.66GHz において、各 I-VOP、P-VOP、B-VOP についての所要時間の平均の近似によって求めており、圧縮処理を行う台数は 4 台としてある。また、総フレーム数は 300、GOP は、一般的な 15 フレーム単位、B-VOP は連続して 3 枚ずつ挿入している。さらに、性能の低いコンピュータを利用した時のことを考え、3 台のコンピュータが 2 倍の処理時間を要する場合について結果を調べた。

なお、処理の重要度から B-VOP 並列化では、I-VOP と P-VOP を高速なコンピュータで処理しているため、処理時間が 2 倍になっているのは B-VOP のみとなっている。

5.2. 結果と考察

GOP 並列化と B-VOP 並列化の結果を表 1 に示す。

結果の表は、各方式でそれぞれの GOP の処理を行うのに要した時間である。合計の数値は、GOP 方式なら処理の終わったコンピュータにすぐさま次の GOP を割り当て、それぞれのコンピュータが要した時間のうち、最も処理に時間がかかったものを、B-VOP では、前の GOP と後ろの GOP の I-VOP の処理が被るため、合計の数値は B-VOP の欄を全て足し合わせた場合よりも少なくなっている。

B-VOP 並列化では、1GOP 辺りの処理は速いが総合では GOP 並列化に大きく劣っているのがわかる。

表 1 GOP、B-VOP 並列化の結果

GOP	GOP	GOP 2 倍	B-VOP	B-VOP 2 倍
1	11110	11110	9009	10282
2	10637	21274	8909	10086
3	11144	22288	8935	10157
4	10091	20182	8739	9776
5	11025	11025	8856	10028
6	11313	22626	8811	9931
7	10472	20944	8602	9505
8	11004	11004	8853	10014
9	11099	22198	8915	10058
10	10927	10927	8941	10157
11	11352	22704	8896	10039
12	10977	21954	8650	9642
13	11319	11319	8900	10098
14	11396	22792	8757	9839
15	11766	11766	9045	10400
16	11089	22178	8995	10302
17	10868	21736	8802	9919
18	10632	10632	8863	10015
19	10656	21312	8844	9980
20	10943	10943	8830	9132
合計	55834	88726	155589	156550

しかしながら、GOP 並列化では、性能の低いコンピュータをシステム中に加えた際に、処理にかかる時間が大きく上昇しているのがわかる。

逆に、B-VOP 並列化は、性能の低いコンピュータを混ぜても全体の所要時間がほとんど変わらないのがわかる。これは、高性能なコンピュータに行わせている P-VOP の処理が、性能の低いコンピュータで処理

している B-VOP よりも遅いため、この方式ではここがボトルネックとなっている。

6. まとめ

B-VOP 並列化による MPEG-4 圧縮は GOP 並列化に比べ、低速であることがわかった。しかし、GOP 並列化は 1GOP 辺りの処理時間は変わらず、リアルタイム性を求める場合には B-VOP 並列化の方が向いているということがわかった。また、B-VOP 並列化は GOP 並列化よりも小さい単位で分割しているので両者を組み合わせ、状況に応じて選択できるようにすれば、さらに効率の良い圧縮が出来るのではないかと考えられる。

参考文献

- [1] 池上広済, “ハイブリッド並列プログラミングによる MPEG2 エンコーダの高速化”, 立命館大学大学院 理工学研究科 情報システム学専攻, 修士論文, 2006
- [2] 山村周史, 平田博章, 新實治男, 紫山潔, “並列処理による MPEG エンコーダの高速化”, 電子情報通信学会, CPSY96-64(1996-08)
- [3] グリッド協議会
<http://www.jpgrid.org/index.html>
- [4] 柏木文徳, 三浦康之, MPEG-4 符号化の並列処理に関する検討, 第 70 回情報処理学会全国大会, 3Y-5, 2008.3.
- [5] 三浦康之, 柏木文徳, マルチコアプロセッサ向き MPEG 符号化の並列処理に関する検討, 第 7 回情報科学技術フォーラム(FIT2008), M-012, 2008.9.