

H.264 のデコード処理における 2パス限定投機方式の並列性能評価

川上憂騎[†] 大津金光[†] 横田隆史[†] 馬場敬信[†][†]宇都宮大学大学院工学研究科情報システム科学専攻

1 はじめに

動画像圧縮符号化方式の H.264 は非常に高い圧縮率を達成することができ、ワンセグ放送、携帯端末やインターネット上での動画配信など、様々な分野で利用されている。しかし、H.264 のデコード処理には非常に多くの演算量が必要であり、デコーダの高速化が重要な課題となっている。この課題に対し、Venezia[1] を始めとする組み込み向けプロセッサでは消費電力の増加を避けるためにスレッドレベル並列性および命令レベル並列性を利用した並列化手法の適用による高速化のアプローチがなされているが、H.264 の可変長符号の復号処理など十分な効果が得られない部分が存在する。

我々はプログラムの実行経路 (パス) に着目し、パスの投機的並列実行によるプログラム高速化手法として 2パス限定投機方式 [2] を提案している。2パス限定投機方式は実行頻度が高い上位 2 つのパスを投機的に並列実行することで高速化を達成する実行方式である。ただし、投機失敗時は回復処理を行う必要があるためにオーバーヘッドが増加するため投機実行するパスの予測成功率が性能に大きく影響する。

我々はこれまでに H.264 デコーダプログラム中のいくつかのループに対してパスの実行割合を評価し、パスの予測成功率から適用可能性を示した段階であり、2パス限定投機方式を適用することで実際にどの程度の速度向上が見込めるのか、並列性能の見積りが必要となる。本稿では 2パス限定投機方式に基づいてデコーダプログラムを並列化し、2パス限定投機方式の適用後の並列性能を見積る。

2 2パス限定投機方式

2パス限定投機方式とは、プログラム実行中に実行されるパスのうち実行頻度の高い上位 2 つのパスが支配的な割合を占めることに着目し、これらのパスに特化した最適化を施したコードを投機的に並列実行することでスレッド間並列性やスレッド自身の実行性能の向上を目指したプログラム高速化手法である。同方式では投機対象のパスに不要な命令を削除し、分岐命令に代わってパス実行の整合性を保つための `assert` 命令を用いることで基本ブロック間をまたいだ命令スケジューリングを可能とし、同期待ちの緩和が可能となる。

2パス限定投機方式ではループの 1 イテレーションを単位として実行されるパスを予測し、予測パスに対応する最適化されたコード (投機スレッドコード) を

実行する。予測ミスにより投機実行が失敗した場合は実行を正しく保つために回復処理を行い、もう一方のパスを投機実行する。二度目の投機が失敗した場合は回復処理の後に元のループ構造に沿った非投機のコードを実行する。

3 評価

本評価ではデコーダプログラムを対象に、2パス限定投機方式に基づいて並列化した場合の並列性能の見積りを行う。並列性能の見積りは 2パス限定投機方式適用前後の評価対象部分の実行にかかる所要サイクル数を比較することで行う。

3.1 評価ツール

2パス限定投機方式はパスの予測や投機失敗時の回復処理など、特有の機能を使用する。評価ツールでは実行されるパスの実行サイクル数や投機失敗が発覚するまでの時間を予め計算しておき、#1パスと#2パスのパス情報やコア数、パスの予測や回復処理にかかるペナルティなどをパラメータとして入力することで、2パス限定投機方式の実行動作に基づいて入力から実行にかかる所要サイクル数を計算することができる。

本評価ではパスの予測、回復処理は 1 サイクルで完了するものと仮定した。また、先行スレッドからのレジスタ値の受渡しができるものとし、1 サイクルで完了できるものとした。さらに命令パイプラインのシミュレートは行わず、1 命令 1 サイクルで実行できるものとした。

3.2 評価用入力映像

標準画像の Football, Susie, Tempete の 3 つの映像を H.264 のエンコーダを用いてエンコードしたものをを用いる。符号化条件は解像度は 176x144、走査方式はプログレッシブ、フレーム数は 10 フレームである。

3.3 対象ループとパスの集計結果

本評価では H.264 のデコード処理にかかる所要サイクル数が大きい関数からループ構造の命令数が少なく構造が簡単な `zigzagscan_8x8_cabac` 関数を選び、関数中のパスが複数存在するループを評価対象のループとした。評価ツールを用いて並列性能評価をする前に、対象ループの実行パスの割合を示す。評価用映像をデコードしたときのパスの実行割合は図 1 の通りである。対象ループは実行パスが 2 通りしか存在しないループであり、1 度目の投機が失敗した場合でも必ず 2 度目の投機は成功し、予測成功率の点から 2パス限定投機方式が有効なループである。各入力データがパスの実行割合に与える影響は少なく、どの映像を入力した場合でも #1 パスが 51%、#2 パスが 49% の割合を占めている。このことから、2パス限定投機方式を適用し

Evaluation of Parallel Performance of Two-Path Limited Speculation Method by using H.264 Decoder

[†]Yuki Kawakami, Kanemitsu Ootsu, Takashi Yokota and Takanobu Baba

Department of Information Systems Science, Graduate School of Engineering, Utsunomiya University (†)

た場合、2度の投機実行のうち片方は必ず成功することがわかる。

Football	51.42%	48.58%
Susie	52.46%	47.54%
Tempete	51.33%	48.67%

□ #1 path ■ #2 path

図 1: 対象ループのパスの実行割合

3.4 並列性能の評価結果

図 2 は各入力映像について逐次実行時の実行にかかる所要サイクル数を 1 倍とした時の速度向上率のグラフである。グラフ中の prediction mode はパス予測方法を示しており、perfect は投機実行されるパス(#1 または #2) の予測を 100% 成功させる予測器である。対象ループではパスは #1 パスと #2 パスの 2 通りしかないため、perfect 予測器を用いることで投機を 100% 成功させることができる。#1, #2 はそれぞれ #1, #2 と予測結果を返す予測器となっている。グラフからわかるように、どの入力映像の場合でもコア数が増加するにつれ、速度向上が見られる。perfect 予測時の Football の場合、コア数 2 で 1.39 倍、コア数 4 で 1.57 倍、Susie の場合はコア数 2 で 1.41 倍、コア数 4 で 1.99 倍、Tempete の場合でコア数 2 で 1.38 倍、コア数 4 で 1.76 倍となった。

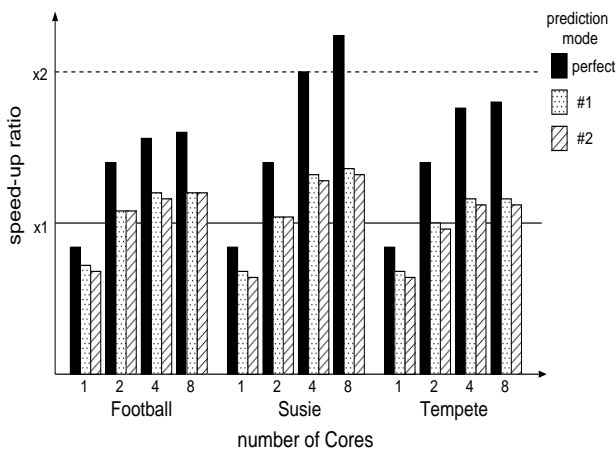


図 2: 方式適用後の速度向上率

3.5 考察

Football, Tempete ではコア数 2, 4 のときはコア数を増やすことで性能が向上した。コア数 8 のときはコア数を増やしても性能向上の幅が小さかった。Susie は 2 つに比較してコア数 8 以上でも速度向上が見られ、perfect 予測器を用いた場合でコア数 8 で最大約 2.2 倍の速度向上となった。ここで、対象ループの実行回数と各ループ実行あたりのイテレーション数の分布を表 1 に示す。2 パス限定投機方式ではループイテレーション単位で並列実行するため、ループのイテレーション数が大きいほど性能が上がる。表によれば、ループ実行の

うち多くは 1 回ないし 2 回程度のイテレーション数でループ実行が終わっている。また、Football, Tempete ではイテレーション数が 8 を超えるループ実行が全体の 1% 以下であることから、コア数に対して十分なスレッド数が生成できていないことが原因である。

表 1: 対象ループのイテレーション数の分布

number of iterations	Football	Susie	Tempete
1	159	97	135
2	213	96	185
3	37	71	92
4	16	42	33
5	6	24	13
6	2	13	4
7	1	11	1
8	0	5	1
9	0	2	0
10	0	2	0
11	0	0	0
12	0	1	0

4 おわりに

2 パス限定投機方式に基づいた並列化の結果、デコーダプログラム中の対象ループの並列実行により最大で約 2.2 倍程度の速度向上が期待できることがわかった。また、ループ実行あたりのイテレーション数が少ない場合、スレッドが十分に生成されないために速度向上率に影響することを示した。本評価で用いた評価ツールは並列性能見積りのための簡易的な評価環境であるため、命令パイプラインなどのプロセッサ内部の動作を正確にシミュレーションしていないという問題があった。そこで、より詳細な評価を行うためにプロセッサ内部を正確にシミュレートするマルチコアプロセッサ PALS[3] 上でデコーダプログラムの性能評価を行う予定である。

謝辞

本研究は、一部日本学術振興会科学研究費補助金 (基盤研究 (C)21500050, (C)21500049, 同 (C)20500047) および宇都宮大学若手萌芽的研究プロジェクトの援助による。また (株) 東芝セミコンダクター社より Venezia 開発環境ならびに H.264 コードを提供いただき実施しました。

参考文献

- [1] T. Miyamori “Venezia: a scalable multicore subsystem for multimedia applications”, in Proceedings of the 8th International Forum on Application-Specific Multi-Processor SoC, Aachen, Germany, June 2008.
- [2] 横田隆史, 齋藤盛幸, 大津金光, 古川文人, 馬場敬信, “2 パス限定投機方式の提案”, 情報処理学会論文誌: コンピューティングシステム, Vol.46, No.SIG 16(ACS-12), pp.1-13, 2005
- [3] 十鳥弘泰, 大津金光, 横田隆史, 馬場敬信, “2 パス限定投機方式を実現するマルチコアプロセッサ PALS の提案”, 信学技報, Vol.109, No.319, pp.19-24, (CPSY2009-46), 2009 年 12 月 3 日.