C-033

マルチメディア応用ヘテロジニアスマルチコアアーキテクチャのための最適メモリアロケーション

Optimal Memory Allocation for Heterogeneous Multicore Architecture for Multimedia Applications

松田 岳久† ハシタ ムトゥマラ ウィシディスーリヤ† 張山 昌論†
Takehisa Matsuda Hasitha Muthumala Waidyasooriya Masanori Hariyama
亀山 充隆†
Michitaka Kameyama

1. まえがき

近年，モバイル機器の普及により，低消費電力で高速な処理を行うことが可能なプロセッサが必要とされている．そのため，最近注目されているのがヘテロジニアスマルチコアプロセッサである．ヘテロジニアスマルチコアプロセッサとは異種のコアを複数もつプロセッサで，それぞれのコアの特徴を生かすことにより全体の性能を向上させることが可能である．特に，マルチメディア分野においては，全体の制御を行う汎用プロセッサと並列演算が可能なアクセラレータコアを組み合わせることで，低消費電力で高速な処理を行うことが可能になる [1]，[2]．しかしながら，画像などの大容量なデータを処理する場合に，アクセラレータコアとグローバルメモリとのデータ転送時間がボトルネックとなるという問題がある．本稿ではこのデータ転送時間を短縮するメモリアロケーションを提案する．

2. アーキテクチャモデル

本研究では図 1 に示すアーキテクチャモデルをもつヘテロジニアスマルチコアプロセッサを考える．プロセッサは複数の CPU コア，アクセラレータコア，メモリコア及びグローバルメモリを持つ．各コアは相互結合網で接続され，コア間において自由にデータ転送ができる．

次に，アクセラレータコアのアーキテクチャモデルを図 2 に示す．アクセラレータはコアの外部に容量の大きなグローバルメモリを持ち，コアの内部には並列アクセス可能なローカルメモリを持つ．各ローカルメモリは独自のアドレス生成回路を備える．ローカルメモリと演算器はクロスバーネットワークにより接続され，コントロールステップごとに必要に応じて接続を変えることができる．ローカルメモリは容量が小さく，画像すべては入り切らない．そのため，グローバルメモリに画像データを入れ，ここからデータの一部をローカルメモリに転送し，演算を行い，再び必要なデータを転送するという作業を繰り返すことにより処理を行う．

ローカルメモリのアドレス生成回路が複雑な場合，実装面積と消費電力が大きくなる問題がある．この問題を解決するためアドレッシング関数をシンプルなものとする必要がある．しかしながら，アドレッシング関数を制約すると任意のアドレスにアクセスできないため，ウインドウ演算などを行う場合に重複記憶が発生する．ローカルメモリに記憶するデータをすべて転送するため，この重複記憶が転送時間の増加につながる．そのため，転送量を抑えるためには，シンプルだが重複記憶を減らせるアドレッシング関数を用いる必要がある．この条件を満たすアドレッシング関数として式 (1) の関数を用いた．

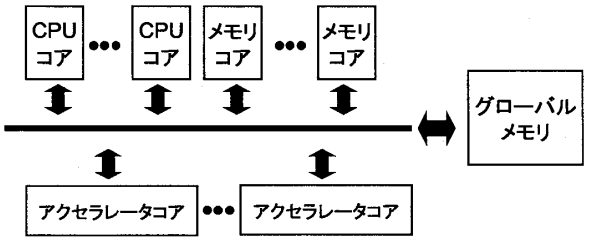

図 1: ヘテロジニアスマルチコアの構成

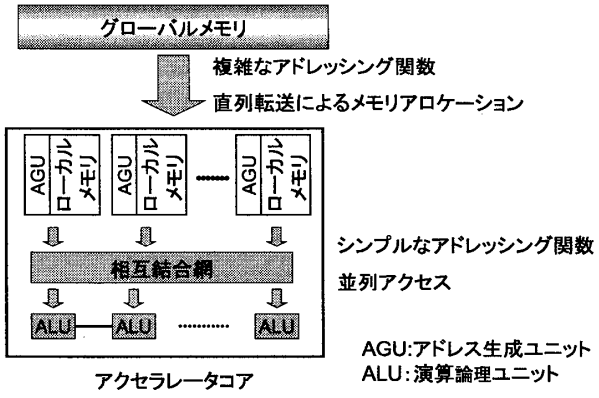

図 2: アクセラレータコアのアーキテクチャモデル

$$f(t) = m \times \mathrm{mod}(t, P) + n \times \left\lfloor \frac{t}{P} \right\rfloor \quad (1)$$

ここで t はコントロールステップ，m,n,P はそれぞれ，$m \geq 0$，$n \geq 0$，$P \geq 1$ を満たす整数である．この関数は概形は図 3 で与えられ，傾き m，周期 P のノコギリ波に 1 周期ごとに n が

† 東北大学大学院情報科学研究科 Graduate School of information Sciences TOHOKU University

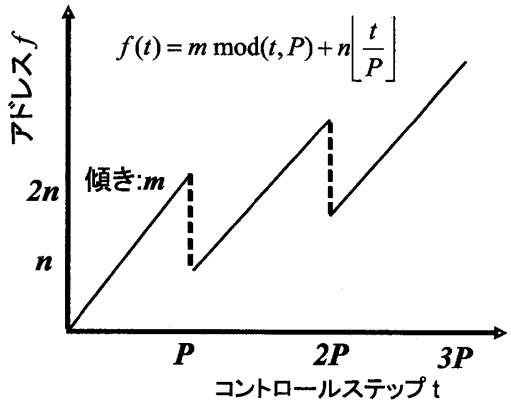

図 3: アドレッシング関数

(a) データシーケンス

コントロールステップ	0	1	2	3	4	5	6	···
データ	d0	d1	d2	d1	d2	d3	d2	···

(b) $f(t) = t$ の場合のメモリアロケーション

アドレス	0	1	2	3	4	5	6	···
データ	d0	d1	d2	d1	d2	d3	d2	···

(c) $f(t) = \mathrm{mod}(t,3) + \left\lfloor \frac{t}{3} \right\rfloor$ の場合のメモリアロケーション

アドレス	0	1	2	3	4	5	6	···
データ	d0	d1	d2	d3	d4	d5	d6	···

図 4: メモリアロケーションの例

加算された概形をしている．この関数はカウンターと加算器のみで実現可能であり，アドレス生成回路を小さくすることが可能である．この関数を用いたメモリアロケーションの例を図 4 に示す．図 4(a) はコントロールステップとそのコントロールステップで必要なデータをセットにしたデータシーケンスである．図 4(b) はシンプルなアドレッシング関数として典型的な線形関数 $f(t) = t$ の場合のデータ配置 (メモリアロケーション) である．この場合与えられたデータシーケンスの順にデータを読み出すため，d1 がアドレス 1 と 3，d2 がアドレス 2 と 4 にそれぞれ重複記憶されてしまっている．図 4(c) は $m = 1, n = 1, P = 3$ の場合のメモリアロケーションである．このアドレッシング関数では，アドレス 0,1,2 とアクセスした後，再びアドレス 1 に戻り，図 4(a) のデータシーケンスを実現できている．このように同一データが必要なときに同一アドレスにアクセスすることで重複記憶をなくすことができる．

3. データ転送量最小化を指向したメモリアロケーションの定式化と評価

メモリアロケーション最適化問題の概要を説明する．目的関数はグローバル・ローカルメモリ間のデータ転送量であり，最小化が目的である．上述のように，これはローカルメモリの記憶容量最小化に帰着される．入力は図 4(a) に示すようにデータシーケンスであり，このデータシーケンスに従って，データを読み出しできるようにローカルメモリにデータを配置する．制約条件はローカルメモリの個数とメモリのワード数である．自由度はアドレッシング関数のパラメータ m,n,P である．

表 1: 比較結果

メモリの数	線形 $f(t) = t$		最適解 $f(t) = \mathrm{mod}(t,4) + \lfloor t/4 \rfloor$		
	転送量	重複度	転送量	重複度	探索時間
1	144	4	72	2	0.031s
2	96	2.66	48	1.33	0.073s
4	72	2	36	1	27.0s

アドレッシング関数のパラメータ m,n,P と，メモリと演算器との配線の組み合わせを総当りで探索するプログラムを作成し，評価を行った．評価はサイズ 4×4 のウインドウ演算について行い，ローカルメモリの個数が 1 個，2 個，4 個の場合に対しデータ転送量を最小とするメモリアロケーションを求めた．画像サイズは 6×6，メモリのワード数の合計は 80 とした．ウインドウを横方向に 1 ピクセルずつ動かすスケジューリングに従ってデータシーケンスを作成した．プログラムは C 言語で作成し，コンパイラとして Microsof Visual Studio 2008 を用いた．CPU は Intel Core 2 Duo，周波数は 3.0GHz である．最適解 $m = 1, n = 1, P = 4$ のときの $f(t) = \mathrm{mod}(t,4) + \lfloor t/4 \rfloor$ と線形関数 $f(t) = t$ についての比較結果を表 1 に示す．表 1 における重複度は (転送量)/(画像サイズ) で定義され，一つの画素データあたりの平均記憶回数を表す．この結果より線形の場合と比較して最適解の方がデータ転送量を 50% にできていることがわかる．また，メモリの数が大きくなるほど，転送量が減り，重複度が下がっているのがわかる．プログラムの出力より，(メモリの数)>(ウインドウサイズの一辺) の場合はデータの重複なしにデータを転送できることを確認した．

4. まとめ

本稿では重複記憶を削減できるシンプルなアドレッシング関数を対象としたメモリアロケーションを提案した．総当り探索ではメモリの数が増加するほどに探索時間が指数関数的に増加してしまうため効率の良い解法が必要である．

参考文献

[1] H. Kondo, M. Nakajima, N. Masui, S. Otani, N. Okumura, Y. Takata, T. Nasu, H. Takata, T. Higuchi, M. Sakugawa, H. Fujiwara, K. Ishida, K. Ishimi, S. Kaneko, T. Itoh, M. Sato, O. Yamamoto and K. Arimoto, "Design and Implementation of a Configurable Heterogeneous Multicore SoC With Nine CPUs and Two Matrix Processors", IEEE Journal of Solid-State Circuits, Vol.43, No.4, pp.892-901, 2008

[2] H. Shikano, M. Ito, M. Onouchi, T. Todaka, T. Tsunoda, T. Kodama, K. Uchiyama, T. Odaka, T. Kamei, E. Nagahama, M. Kusaoke, Y. Nitta, Y. Wada, K. Kimura, H. Kasahara", "Heterogeneous Multi-Core Architecture That Enables 54x AAC-LC Stereo Encoding", IEEE Journal of Solid-State Circuits, Vol.43, No.4, pp.902-910, 2008