

実アプリケーションによるチップマルチベクトルプロセッサの性能評価

佐藤義永¹, 撫佐昭裕², 江川隆輔¹, 滝沢寛之¹, 岡部公起¹, 小林広明¹

1 東北大学, 2 日本電気株式会社

yoshiei@sc.isc.tohoku.ac.jp, a-musa@bq.jp.nec.com, {egawa, tacky, okabe, koba}@isc.tohoku.ac.jp

概要：本研究では、ベクトル型スーパーコンピュータで利用されるアプリケーションに内在する高い並列性とスレッド内・スレッド間のデータ参照の局所性に注目し、複数のベクトルコアによってスレッドの並列処理を可能とする、チップマルチベクトルプロセッサ(CMVP)を検討している。本報告では、実アプリケーションを用いて CMVP の性能評価を行い、複数コアが実現する高い演算性能とベクトルキャッシュが実現する高い実効メモリバンド幅の双方の効果により、これまでよりも非常に高い実効性能を達成可能であることを明らかにする。

1 はじめに

半導体加工プロセスの微細化により、プロセッサに搭載可能な演算ユニットは年々増加している。しかし、プログラムに含まれる命令レベル並列性の限界から、単一コアの演算ユニットを増加するだけでは著しい性能向上は得られなくなってきている。そこで、スレッドレベル並列性に基づき、複数スレッドを同時処理可能なチップマルチプロセッサ(CMP)の研究が盛んに行われている。しかし、スカラプロセッサでは既にその効果が実証されて一方で、ベクトルプロセッサの CMP はまだ研究段階である。

本報告では、複数のコアを搭載するベクトルプロセッサ(チップマルチベクトルプロセッサ, CMVP)について検討する。CMVP は複数のコアを搭載することでベクトル演算性能の飛躍的な向上が期待できる。一方、メモリバンド幅は、物理的限界のためこれ以上の向上が困難である。したがって、マルチコア化により演算性能に対する相対的なメモリバンド幅である B/F が大幅に減少する。 B/F の減少によってアプリケーションの実行効率は低下することが報告されており[1], コア数を増加し、演算能力のみを向上するだけではメモリバンド幅がボトルネックとなり、マルチコア化によるスケラビリティが得られない恐れがある。これまでの研究からベクトルプロセッサにキャッシュメモリ(ベクトルキャッシュ)を搭載することで B/F 低下の影響が緩和可能であることを実証しており、同機構を CMVP に用いることでスケラビリティの向上が期待できる。本報告では、実アプリケーションによる CMVP の性能評価を行い、ベクトルプロセッサのマルチコア化による性能向上の可能性を示す。

2 チップマルチベクトルプロセッサ

本報告で検討する CMVP の構成を図 1 に示す。プロセッサは、複数のベクトルコアと、コア間でのデー

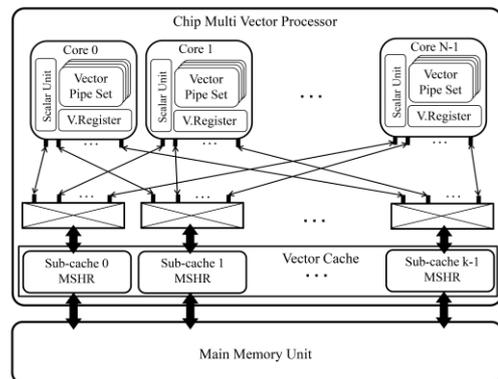


図 1. CMVP の構成

タ共有を可能とする共有ベクトルキャッシュで構成される。プロセッサの最大演算性能はコア数に比例して増加させることができる。共有ベクトルキャッシュは k 個のサブキャッシュによって構成されており、各サブキャッシュが 1 つのメモリポートに直結されている。この構成により、データをメインメモリから共有ベクトルキャッシュにインターリーブ転送することが可能となり、メモリアクセス時間を隠蔽することができる。メインメモリから送られるデータを共有ベクトルキャッシュに保存し、次回アクセス時には共有ベクトルキャッシュから各コアへ直接データを転送する。さらに、キャッシュミス時のアドレス情報管理を行う Miss Status Handling Register (MSHR) を各サブキャッシュに搭載する。ロード命令はメインメモリへアクセスする前に MSHR を確認し、既に先行するロード命令が同一アドレスへアクセスを行っている場合にはデータの到着を待つ。結果として冗長なメモリアクセスを削減でき、実効メモリバンド幅が向上する。各コアと共有ベクトルキャッシュ間の通信は、 k 個のクロスバースイッチを用い、高いメモリバンド幅を実現する。

3 性能評価

本評価では、コア数が 1 コアから 16 コアまで増加

した場合の、アプリケーション実効性能のスケラビリティとベクトルキャッシュの効果シミュレーションにより評価する。メインメモリとプロセッサ間のメモリバンド幅はコア数によらず一定の値で固定とする。従って、コア数増加に伴い、1 コアあたりのメモリバンド幅は減少し、1 コアでは4B/F、2 コアでは2B/F、16 コアでは0.25B/Fまで減少する。評価にはベクトルプロセッサのタイミングシミュレータを用い、キャッシュ容量1MB、サブキャッシュ数32、MSHR エントリ数8,192とする。評価用アプリケーションにはアンテナ解析プログラム[2]と姫野ベンチマーク[3]を用いる。アンテナ解析プログラムはFDTD法を用いたシミュレーションプログラムである。姫野ベンチマークはポアソン方程式をヤコビの反復法で解く際の主要ループで構成されたベンチマークプログラムである。

アンテナ解析プログラムの実行結果を図2に示す。このプログラムは、メモリアクセス数に比べて演算数が多く、メモリバンド幅がボトルネックになりにくいという特徴がある。そのため、キャッシュを用いない場合でもコア数増加に伴い実効性能は向上する。しかし、8 コア(B/F=0.5)以上では性能が低下し始める。性能低下の原因として、コア数増加により各コアからのメモリアクセス要求が同時に同一のメモリポートに集中し、メモリポートで競合が頻繁に生じる結果、実効メモリバンド幅が低下することが考えられる。メモリバンド幅が十分に高い場合には隠蔽可能であったポートコンフリクトによるロード遅延時間の増大が、コアあたりのメモリバンド幅が低下したことにより、隠蔽不可能となったために、実効性能が低下した。一方、キャッシュを用いる場合は、16 コア(B/F=0.25)においても高いスケラビリティが得られる。コア毎の実行効率では1 コア時83.7%に対して、16 コア時でも78.9%と高く保つことができる。これは、各コアで利用されるデータを再利用することで、実効メモリバンド幅が向上し、コアあたりのB/F低下の影響が緩和される結果、高い実効性能が実現できたためである。

次に、姫野ベンチマークの結果を図3に示す。このプログラムはメモリアクセスが多く、メモリバンド幅がボトルネックになりやすいという特徴がある。そのため、キャッシュを用いない場合では、コア数増加に伴うB/F低下の影響を受けるため、コア数を増加させても性能は向上しない。一方、キャッシュを用いる場合は、4 コア(B/F=1.0)までは性能向上が得られるが、8 コア以上(B/F=0.5以下)では、キャッシュを用いているにも関わらず性能が低下する。コア毎の実行効率は、1 コア時49.9%に対して、4 コア時では49.1%、

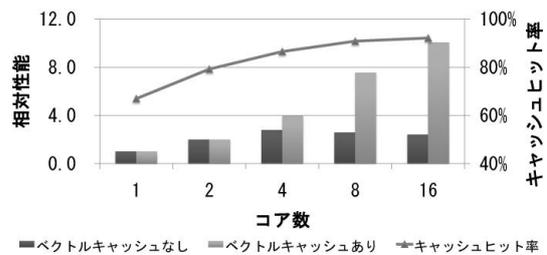


図2. アンテナ解析プログラムによる性能評価

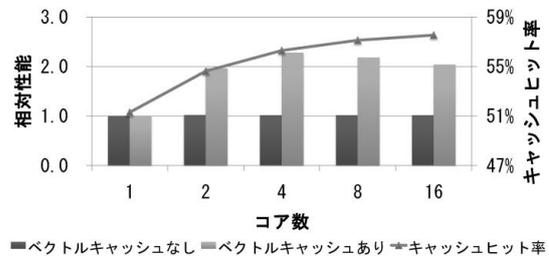


図3. 姫野ベンチマークによる性能評価

8 コア時では28.5%、16 コア時では6.4%まで低下する。これは、姫野ベンチマークのメモリアクセスが多いという特徴から、キャッシュによるデータ再利用の効果のみでは、コア数増加によるB/F低下の影響を補うことができず、ポートコンフリクトによる遅延時間を隠蔽できないためである。

4 まとめ

本報告では、CMVPの有効性を実アプリケーションにより評価した。CMVPは、複数コアを搭載することで、ベクトル演算性能を飛躍的に向上させ、さらにベクトルキャッシュを用いることで実効メモリバンド幅を高く保つことが可能であることが示された。しかし、8 コア以上では、コア数増加に伴うコアあたりのB/F低下により、キャッシュを用いているにも関わらず実行効率が低下する場合もあることが分かった。今後、キャッシュ構成やプログラム最適化などによる効率的なデータ転送について検討し、より高いスケラビリティを有するCMVPを実現する。

参考文献

- [1] H. Kobayashi, et al., "Implication of Memory Performance in Vector-Parallel and Scalar Parallel HEC," High Performance Computing on Vector Systems 2006, Springer-Verlag, pages 21-50, 2006.
- [2] Y. Takagi, et al., "Study of High Gain and Broadband Antipodal Fermi Antenna with Corrugation," In: Proceedings of 2004 International Symposium on Antennas and Propagation, vol. 1, pages 69-72, 2004.
- [3] Himeno benchmark.
<http://w3cic.riken.go.jp/HPC/himenoBMT/>