

第一原理分子動力学を用いたスーパーテクニカルサーバ SR11000 モデル K1 のバリア同期機構の性能評価

高山恒一[†] 青木秀貴[†] 富田亜紀[†]

(株)日立製作所 中央研究所[†]

1 はじめに

高性能並列計算機を利用する主要アプリケーションの一つに第一原理分子動力学[1]がある。第一原理分子動力学は、分子間に働く電磁相互作用を求める計算手法であり、蛋白質や金属の物性など幅広い分野のシミュレーションで利用されている。シミュレーションで扱える原子の数は計算機の演算性能によって制限される。原子数の多い高精度なシミュレーションを実行するには、演算性能の高い並列計算機が必要である。

日立スーパーテクニカルサーバ SR11000 は、ベクトル・スカラー融合を特長とする SR8000 の後継機である。2005 年 10 月に製品化した SR11000 モデル K1 は、2.1GHz で動作する POWER5+ 16CPU がメモリを共有するノードを基本単位とし、多段クロスネットワークで最大 512 ノードを繋げることで理論性能 68.8TFLOPS を実現する。SR11000 は SR8000 の特長技術である、協調型マイクロプロセッサ機構(COMPAS)を継承している[2]。COMPAS を実現する主な技術要素は、ユーザプログラムから逐次実行部と並列実行部からなる SIMD 並列コードを生成する自動並列化コンパイラと、SIMD 並列コードをノード内で高速に実行する上で必要となる CPU 間的高速バリア同期機構である。バリア同期とは、全 CPU で一斉に待ち合わせを行う同期であり、逐次実行部と並列実行部の間や、依存性のある並列実行部同士の間に行われる。

第一原理分子動力学計算は、複数ノードを使用する大規模な計算であり、各ノードの演算性能がシステム全体の性能を決定する。各ノードの演算性能を高めるには、ノード内の 16CPU を並列に実行して効率的に演算をすることが重要である。並列演算では、1CPU の逐次演算には無いバリア同期を加えることによって、複数の CPU を制御する。バリア同期を高速に実行することが並列演算の実行効率を高める上で必須である。本研究では、SR11000 モデル K1 の 16CPU 間のバリア同期

の高速化によって得られる、第一原理分子動力学計算の演算性能向上の定量的評価を目的とする。

2 バリア同期の基礎評価

2.1 複数 CPU を用いた演算とバリア同期

SR11000 ではメモリを共有する 16CPU を協調動作させることにより演算性能を高めている。科学技術計算の中で異なるデータに対して同じ種類の演算を行うときの各 CPU の演算分割の例を図 1 に示す。

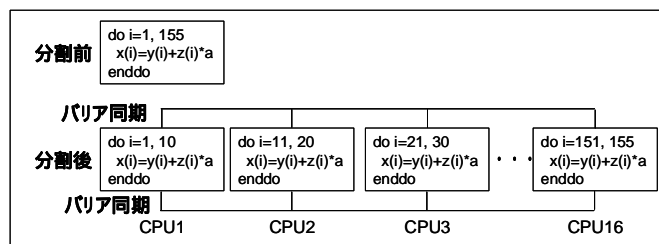


図 1 : 演算ループの分割実行の例

図 1 の分割前ではループを 1 から 155 まで実行して配列 X を求めている。この演算ループを 16CPU で分割して同時に実行することにより演算時間の短縮を図る。分割後の演算では、演算ループの前後にバリア同期を入れることで、各 CPU の演算開始と終了を制御する。このバリア同期の 1 回あたりの実行時間は CPU 数と共に増加するため、CPU 数を増やして演算時間の短縮を図ると逆に演算時間に占めるバリア同期のオーバーヘッドの割合が大きくなる。バリア同期のオーバーヘッドを軽減するには、1) プログラムの最適化によりバリア同期の回数を削減、2) バリア同期自体を高速化、の 2 つの手法がある。本研究では SR11000 モデル K1 の 16CPU 間のバリア同期性能の評価を目的とすることから、2) に関して定量的に分析する。

2.2 基礎演算ループによるバリア同期の評価

SR11000 モデル K1 には従来のモデルから CPU 間的高速バリア同期機構として採用している FBS (Fast Barrier Synchronization) 方式の他に、専用

Performance Analysis of Barrier Synchronization of SR11000 model K1 for First Principle Molecular Dynamics
[†] Hitachi, Ltd., Central Research Laboratory

ハードウェアの実装により更に高速化を図った BSR-FBS(Barrier Synchronization Register - Fast Barrier Synchronization)方式を実装している。それぞれのバリア同期の実行方法を表 1 に示す。

表 1 : SR11000 モデル K1 のバリア同期機構

項番	種類	実行方法
1	FBS方式	各CPUが処理を完了したバリアポイント番号をキャッシュメモリの同期フラグに保持。同期を取る相手の全CPUのバリアポイントを確認して、同期処理を終了。
2	BSR-FBS方式	上記FBS方式における同期フラグをメモリにマップされたハードウェア資源(BSR)に格納する方法。BSRはストアされた値をノード内の全CPUに高速に伝達する機能を持ち、FBS方式よりも高速化が可能。

FBS 方式に対する BSR-FBS 方式の高速化の効果に関して、科学技術計算で一般的な DAXPY と DDOT を基に評価する。DAXPY と DDOT の演算内容を以下に示す。

$$\text{DAXPY: } A(i)=A(i)+S*B(i)$$

$$\text{DDOT: } S=S+A(i)*B(i)$$

SR11000 モデル K1 の 16CPU を用いて DAXPY の性能を測定した結果を図 2 に示す。

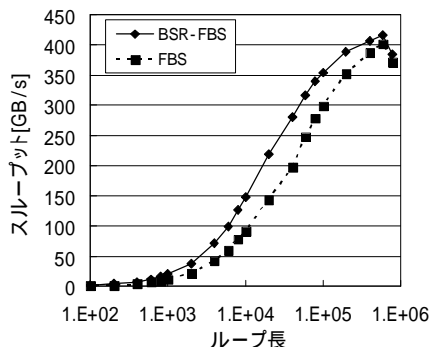


図 2 : DAXPY の性能

図 2 では横軸がループ長、縦軸がデータアクセスのスループット性能を示す。図の測定では、参照データが L2 キャッシュに入るループ長に設定した。ループが長くなると、L3 キャッシュや主記憶からデータを読み込むオーバーヘッドが支配的になることから、CPU 間のバリア同期方式の違いによる性能差は殆ど発生しない。図 2 に示す通り、FBS 方式に比べて BSR-FBS 方式の方が、CPU 間のバリア同期の起動終結時間が短いため、スループット性能が高くなっている。DDOT について測定した場合も同様の結果になる。測定した DAXPY および DDOT の性能からバリア同期の起動終結時間を求めた結果を表 2 に示す。

表 2 : DAXPY, DDOT から求めたバリア同期時間

基本演算	DAXPY		DDOT	
	FBS方式	BSR-FBS方式	FBS方式	BSR-FBS方式
起動終結時間[μs]	2.2	1.1	2.7	1.6

表 2 からバリア同期の起動終結時間は FBS 方式に対し BSR-FBS 方式の方が 1.1[μs]短く、DAXPY では BSR-FBS 方式のバリア同期の起動終結時間は FBS 方式の時間に比べて 50%、DDOT では 60%に削減できることが確認できた。次章では 2 つのバリア同期方式を第一原理分子動力学に適用し、BSR-FBS 方式の効果を定量的に評価する。

3 BSR-FBS 方式を適用した第一原理分子動力学計算の実効性能評価

第一原理分子動力学では、対象とする分子に含まれる原子が移動する度に、原子の周囲に分布する電子のエネルギーバンドの計算を実行し、原子の内部エネルギーと原子に働くクーロン力を計算する。電子のエネルギーを求める固有値計算やクーロンポテンシャルを求める際の FFT 計算等、複数の計算法を組み合わせとなるため、計算法の間でバリア同期を数多く実行する。第一原理分子動力学計算を SR11000 モデル K1 の 16CPU で実行した結果について、FBS 方式を用いた時の実行時間を 100 とした相対実行時間を表 3 に示す。

表 3 : バリア同期法による相対実行時間の違い

	相対実行時間
FBS方式	100.0
BSR-FBS方式	95.1

表 3 の通り、BSR-FBS 方式を使うことにより FBS 方式に比べて約 5%の実行時間短縮を確認できた。バリア同期を 1.1[μs]短縮することが約 5%の実行時間短縮になっていることを表 2 にあてはめて比例計算すると、FBS 方式では全実行時間の 10~12%、BSR-FBS 方式では全実行時間の 5~7%が 16CPU 間のバリア同期時間であると推定できる。

4 まとめ

SR11000 モデル K1 で実装した高速バリア同期機構の基礎性能を評価した。専用ハードウェアを実装することで、バリア同期時間を半分に短縮して 1.1[μs]を達成できることを確認した。主要なユーザアプリケーションである第一原理分子動力学計算に適用して定量的評価を行ったところ、実行時間を約 5%削減できた。今後、本研究の手法を他のアプリケーションに適用し、演算性能とバリア同期時間の関係を更に分析する。

参考文献

- [1]上田顕：“分子シミュレーション”，裳華房 (2003)
- [2]青木秀貴 他：“スーパーテクニカルサーバ SR11000 モデル J1 のノードアーキテクチャと性能評価”，SACSIC2005, 35, 2005