

スカラ型並列計算機に向けた気象シミュレーションコードの作成法

高山恒一†
Koichi Takayama

合田徳夫‡
Norio Gouda

1 はじめに

高性能な並列計算機を必要とする主要な分野として、気象シミュレーションがある。気象庁及び気象研究機関では気象シミュレーションを実行して、日々の天気予報から100年後の地球温暖化の予測まで行っている[1]。精度の高いシミュレーションを短時間に実行するため、高性能な並列計算機上で演算性能を引き出すシミュレーションコードを実行することが必要とされている。気象シミュレーションでは、大気中の風速、気温や水蒸気量といった多量の物理変数を処理するため、扱うデータサイズが大きく、これまでベクトル型計算機が有効とされてきた[2]。しかし、近年開発される高性能計算機は、大容量キャッシュを持つスカラプロセッサを複数繋ぐスカラ型並列計算機が主流である[3]。そのため、スカラ型並列計算機の高い演算性能を引き出す気象シミュレーションコードの作成法が必要とされている。

従来よりスカラ型並列計算機に適した高速化法として、キャッシュブロッキングによりデータの再利用性を高め、メモリへのアクセス時間を短縮する方法が知られている[4]。気象シミュレーションコードでは複雑な物理現象を解くため、一般にサブルーチン数が100以上、演算ループ数が1000以上と規模が大きく、個々のループに対してキャッシュブロッキングを適用しても、全体に対する高速化の効果が少ない問題点がある。また、高速化法はプロセッサが持つキャッシュ容量などの計算機アーキテクチャーに依存するため、計算機毎に違ったコードの開発が必要となる。本研究では、アルゴリズムレベルからの解析によってコードの特徴を捉えた上で、気象分野に適用できる一般的な高速化法を求めることを目的とする。本手法により、キャッシュブロック化がコード全体に適用できると共に、計算機アーキテクチャーに依存しない高速化を実現する。ここでは、1ノードのメモリを共有するスカラ型並列計算機に絞って高速化法を検討する。

2 気象シミュレーションの概要

気象シミュレーションでは、地球の周囲の大気を数[Km]から数十[Km]の間隔で区切った3次元格子を作り、各格子点の風速、気温といった物理量の時間変化を求める[1]。シミュレーションには、格子点上の物理量を使い大気の流れを表すナビエ・ストークス方程式、水蒸気や空気の質量保存の式、気体の状態方程式、エネルギー保存の式を連立して解く力学過程と、水蒸気の凝結による雲の生成といった格子間隔よりも微小な現象をモデル化して解く物理過程の2つの部分がある。気象シミュレーションのメインループでは、力学過程と物理過程を交互に解き、数十秒後の物理

量を予測する。メインループを繰り返すことで、数時間後から数日後の気象予報を求める。

力学過程の解き方には、差分法とスペクトル法があり、並列計算機のプロセッサ数によって、最適な並列化手法と解き方が違う。一方、物理過程の解き方は計算機に依らず同じ手法を用いる。本研究では、第一段階としてアプリケーションの共通部分を抽出して一般的高速化法を求め易い物理過程に絞って検討する。力学過程の高速化法に関しては、今後次の段階として検討する。

3 高速化法

3.1 気象シミュレーションの特徴

気象シミュレーションでは格子間隔を狭めると予報の精度を高めることができる。各国気象庁が現在使用している気象シミュレーションの規模と倍精度実数の1物理変数当たりのデータ量を表1に示す[1]。

表1：シミュレーションの格子間隔とデータ量

シミュレーション	国名	格子間隔[Km]	データ量[MB]
地域予報 (翌日の予報)	日本	5.0	158.7
	ドイツ	3.5	127.8
	カナダ	2.5	76.4
大気大循環 (週間予報)	ヨーロッパ	25.0	888.7
	日本	20.0	843.8
	アメリカ	40.0	240.0

表1から1変数当たり100から900[MB]であることが分かる。気象シミュレーションではこのような変数を60から100種類使う。スカラプロセッサのキャッシュ容量は高々プロセッサ当たり数10[MB]以下であり[5]、扱うデータ量がキャッシュ容量より著しく大きいことが分かる。

図1に物理過程で典型的な演算ループを示す。

```

do k = 1, nz ← 高度格子数のループ (ループ長: 30~90)
do j = 1, ny ← 緯度格子数のループ (ループ長: 300~1000)
do i = 1, nx ← 経度格子数のループ (ループ長: 300~2000)
  a(i, j, k) = a(i, j, k) + r * a(i, j, k-1) + s * c(i, j, k) + d(i, j, k) + e(i, j, k)
enddo
enddo
enddo

```

図1：シミュレーションの演算ループの例

演算ループは内側から経度格子、緯度格子、高度格子の順に3重ループとなっている。使用する配列a, c, d, eは経度、緯度、高度の順に連続アドレスに配置されている。また、演算ループでは更新配列a(i, j, k)はa(i, j, k-1)から作成されることからデータ依存があり、kループを分割して複数のCPUで同時に実行することはできない。表2に気象シミュレーションコードの特徴を抽出した結果を示す。表2に示す物理過程の演算の特徴を基に、次節ではスカラ型並列計算機に向けた高速化法を提案する。

† (株)日立製作所 中央研究所

‡ (株)日立製作所 エンタプライズサーバ事業部

表2: 気象シミュレーションの特徴

項番	プログラムの特徴	内容	性能に対する特徴
1	1変数のデータ量が数百[MB]	表1参照	
2	変数の種類が60以上	風速(緯度・経度・高度方向の成分) 気温、気圧、乱流エネルギー 雨・雪・水蒸気のそれぞれの混合比他	・数百[MB]の変数が80以上のため使用メモリは10[GB]以上 ・キャッシュ溢れが発生して性能低下の要因
3	ループの引用配列が多い	引用配列及び中間作業配列数が多い	
4	3重ループ単位の演算が多い	図1参照	
5	変数のメモリ上の配置は経度、緯度、高度の順	最内側ループが長い	・連続アドレス参照によりプリフェッチ機構が有効
6	隣接データを引用する演算	水平方向のデータ間に依存が無い 保存量の算出の際だけ、3次元の6方向のデータ参照と更新がある	・分割法によりCPU間で並列実行できず逐次処理を実施 ・バリア同期回数の増加
7	3重ループの数が多	太陽放射、地表面からの水蒸気放出、 雲の形成など複数の物理現象を解くためループが多い	・ループ単位の高速化では効果少 ・多層演算のため大規模行列計算に帰着できない
8	除算が多い	差分化の際の格子間隔による除算 気圧や時間変化する物理量による除算	・リスプロセッサでは除算の実行時間が長い
9	べき乗計算が多い	気温を基準圧力(1000hPa)下の温度に変換して評価するため	

3.2 スカラ型並列計算機に向けた高速化

本研究で提案するスカラ型並列計算機に向けた高速化方針を表3に示す。

表3: 気象シミュレーションに向けた高速化手法

項番	方法	内容	表2の対応
1	ループ融合ブロック化	・サブルーチンを跨ぐ大規模ループ融合と作業配列の削除をした後でブロック化を行い、キャッシュの再利用性向上	1, 2, 3, 4, 7
2	並列化位置の調節	・最内側経度格子を残してプリフェッチの有効利用 ・依存の少ない緯度格子分割によりCPU負荷の均等化とCPU間同期回数の削減	5, 6
3	除算の乗算化	・逆数表を作成して除数が共通の除算を乗算化	8

気象シミュレーションでは多数のループがあるため、1つ1つのループに対してキャッシュブロック化をしても効果が少ない。気象の物理過程では一般的に高度方向の物質対流を扱うため、緯度格子と経度格子の演算順序変更による演算結果の影響はない。そこで、演算順序の変更となるループの交換と融合を利用して高速化を行う。ループの構造は図1の3重ループであるため、高度格子と緯度格子ループを交換した後、緯度格子ループをサブルーチンの外側で融合する。サブルーチンを跨ぐループ融合により、ループ中の演算に利用する配列の再利用性を高めた上でブロック化を行うため、演算全体に渡ってキャッシュの効果を引き出すことができる。本研究の高速化法ではサブルーチン内の数式を変更しないため、サブルーチンの持つ物理的解釈を損なわず、コード保守性が高い。

4 性能評価

3章で提案した高速化法を気象シミュレーションコードの物理過程演算部に適用し、テクニカルサーバ SR11000 モデル H1 の1ノードで測定した際の結果を表4に示す。

表4: 物理過程の主要演算部の高速化

項番	機能	内容	高速化前の 相対実行時間	高速化前の 相対実行時間	高速化後の 相対実行時間 (加速率)
1	雲中の熱対流と水蒸気の 輸送の計算	制御部	0.62	53.46	2.37 (22.58)
		雨粒の落下	49.97		
		水分の移動と温度計算	0.41		
		近似計算処理部	0.05		
		気温計算部	0.06		
			0.83		
			0.47		
2	雲中の微粒子の形成と降 水の発達計算		1.24	28.55	24.58 (1.16)
		制御部	14.41		
		雲中の両成分の計算	1.37		
		降水演算部	1.21		
		雪と雹の計算 (形成から融解)	3.03		
			0.97		
			1.83		
3	雲中の水蒸気量の計算		2.28	18.00	11.17 (1.61)
			1.13		
		氷の生成	2.33		
全体			12.07	100.00	38.12 (2.62)
			5.93		

表4ではサブルーチンを機能毎に分類し、各サブルーチンの演算内容を表す。また表中では、高速化前の全体実行時間を100.00とする相対実行時間で示し、高速化後の性能向上を加速率として括弧内に示す。高速化前は、項番1が53.46であり、特に制御部が49.97と全実行時間の半分を占めている。提案した高速化法ではサブルーチンを跨いでループ融合しているため、高速化後の実行時間比率は機能単位で示す。高速化法を適用することにより全体の相対実行時間は100から38.12に短縮して、2.62倍の実効性能向上となった。項番1の演算性能が22.58倍と大幅に改善されているが、これはSR11000モデルH1の16CPUの演算負荷均等化、キャッシュヒット率向上、CPU間同期回数の削減の効果である。

5. おわりに

本研究では、気象シミュレーションコードのアルゴリズムレベルから解析をして同じアプリケーション分野に適用できる一般的な高速化法を求めた。提案した手法を適用したコードをSR11000モデルH1上で評価して高速化の効果を確認した。今後、本研究の解析法を力学過程にも適用する。また、表3の手法を他のシミュレーションに適用することにより、分野毎に共通な高速化コードの開発を行うと共に、並列計算機アーキテクチャーに向けたアルゴリズム開発に役立てたい。

参考文献

- [1] 気象庁予報部: “気象庁非静力学モデル”, (財)気象業務支援センター(2003)
気象庁予報部: “全球モデル開発プロジェクト(I)”, (財)気象業務支援センター(2005)
- [2] <http://www.spsscicomp.org/ScicomP5/Presentations/Hamrud/mats.ppt#1>
- [3] <http://www.top500.org/lists/plists.php?Y=2005&M=06>
- [4] 寒川光: “RISC 超高速化プログラミング技法”, 共立出版(1995)
- [5] <http://www.top500.org/ORSC/2004/power5.html>
<http://www.top500.org/ORSC/2004/itanium.html#itanium>