

地球シミュレータの 次世代システム

運用状況

本誌2004年9月～11月号に、「国際間競争になっているハイパフォーマンスシミュレーションの最前線」と題して、地球シミュレータの開発から導入、運用状況として、2004年度までの統計につい

て述べた。

ここでは、2002年度から2007年度の6年間に亘る利用状況と成果などについて報告したい。なお、これらの統計については、参考文献（注1）に示される海洋研究開発機構計算システム計画・運用部のホームページで入手可能である。地球シミュレータの利用は、大

きく公募による共同研究、文部省などの委託事業による特定プロジェクト、国際共同研究等、有償利用であるが、この2007年度におけるリソースの配分比率は図1のようになっている。

それぞれ、どんな課題で利用されているかリストアップしたものを「表 研究プロジェクト一覧」に示す。

共同プロジェクトの項目番号1～42は、地球シミュレータセンターからの公募によるものである。

JAMSTECとの国際共同研究等は、13件ある。文科省等からの委託事業は、特定プロジェクトとして合計22件ある。この他に、成果を公開しない民間企業の有償利用（成果専有型有償利用制度に基づくもの）が10件程度ある。

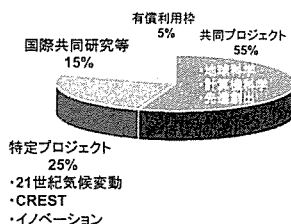
有償以外のそれぞれの課題の詳細については、注2を参照願いたい。また、これらの課題でどんな成果が出ているかは、地球シミュ

分野および機関別利用状況

地球シミュレータの利用機関数と研究者数の年次推移

	利用機関数	利用研究者数
'02年度	93	480
'03年度	171	699
'04年度	193	823
'05年度	199	833
'06年度	193	846
'07年度	174	742

'07年度リソース配分枠



'07年度の内訳

大学: 74	公的機関: 17	民間会社: 42	海外機関: 39
--------	----------	----------	----------

図1

成果と

に向けた課題 その2

独立行政法人海洋研究開発機構

計算システム計画・運用部長 平野 哲

レータセンター利用報告会(注3)や地球シミュレータ成果データベース(注4)を検索すれば得ることができる。

図1において、民間会社や海外機関が多いように見えるが、大学や公的機関の代表者のもとで共同研究者として参加しているケースがほとんどである。

2007年度において利用機関数や研究者数が減ってきているのは、継続課題で名を連ねる研究機関や利用者が過去の実績で使われていない場合、登録しないようお願いした影響である。

次に処理状況の推移について述べたい。図2は大規模ジョブを処理するL系の処理件数推移である。ざっと、月間1万件の処理を行っているがシステム全体では、1ノード内で行う小規模ジョブやデバッグなどを処理するS系(16ノード)も月間で1万件程度処理しているので年間24万件というところ

である。

図3は、2007年度における、これらの処理の規模を示している。テラFlops級のジョブが全体のリソースの75%を使っていることがわかる。

特筆すべきは16テラFlops以上のジョブがリソースの13%を使っているということである。このク

ラスのジョブは、他の機関においても容易に処理はできないのではないかと考えられる。

図4は、64ノード以上を使用するジョブについて、どのくらいの効率で動いているかを示したもので、当然、グラフの縦軸が実効性能を示すので高い方が良い。

横軸はノード数なので8倍する

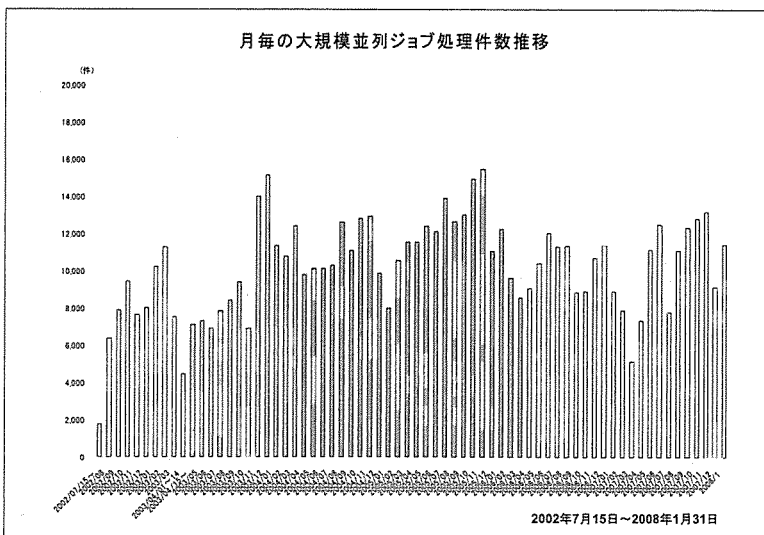


図2

地球科学分科(17件)

プロジェクト名	責任者名	所属
1 化学輸送モデルによる大気組成変動と気候影響の研究	秋元 肇	海洋研究開発機構 地球環境フロンティア研究センター
2 大気・海洋観測現象の理解と予測	大淵 清	海洋研究開発機構 地球シミュレーションセンター
3 地球シミュレータ用・非静力・大気海洋結合モデルの開発	高橋 桂子	海洋研究開発機構 地球シミュレーションセンター
4 気候・海洋変動のメカニズムの解明およびその予測可能性の研究	山形 俊男	海洋研究開発機構 地球環境フロンティア研究センター
5 気候システム研究のための高精度大気・海洋・陸面結合大循環モデルの開発	時田 達志	海洋研究開発機構 地球環境フロンティア研究センター
6 AFESを用いた地球型惑星の大気大循環シミュレーション	林 祥介	神戸大学大学院 理学研究科
7 地球環境変化に伴う生態系変動の診断と予測に関する研究	和田 英太朗	海洋研究開発機構 地球環境フロンティア研究センター
8 高解像度データ同化による海洋状態推定	杉浦 望実	海洋研究開発機構 地球環境フロンティア研究センター
9 ヒートアイランドの数値モデルの開発	足永 靖徳	建築研究所
10 全地球弾性応答シミュレーション	坪井 誠司	海洋研究開発機構 地球内部変動研究センター
11 実地球環境での地球磁場・変動シミュレーション	浜野 洋三	海洋研究開発機構 地球内部変動研究センター
12 マントル対流の数値シミュレーション	深尾 良夫	海洋研究開発機構 地球内部変動研究センター
13 日本列島の地震活動予測シミュレーション	松浦 孝忠	東京大学大学院 理学系研究科
14 3次元不均質場での波動伝播と強震動のシミュレーション	古村 孝志	東京大学 地震研究所
15 複雑断層系の地震発生過程シミュレーション	平原 和朗	京都大学大学院 理学研究科
16 先端の固体地球シミュレーションコードの開発	陸山 聡	海洋研究開発機構 地球シミュレーションセンター
17 火山ダイナミクスの数値シミュレーション	小笠原 剛博	東京大学 地震研究所

計算機科学分科(2件)

プロジェクト名	責任者名	所属
18 連結階層シミュレーションアルゴリズムの開発	草野 完也	海洋研究開発機構 地球シミュレーションセンター
19 大規模科学計算向け汎用数値ソフトウェア基盤の開発	西田 晃	中央大学 21世紀COEプログラム

先進・4次元分科(23件)

プロジェクト名	責任者名	所属
20 ロケットエンジン内部流れのシミュレーション	坪井 伸幸	宇宙航空研究開発機構 宇宙科学研究本部
21 カーボンナノチューブの特性に関する大規模シミュレーション	南 一生	高度情報科学技術研究機構 計算科学技術第1部
22 パーシャル実証試験のための次世代計算固体力学シミュレータの開発	塩谷 隆二	九州大学大学院 工学研究科
23 テラヘルツ帯伝導素子に関する大規模シミュレーション	立木 昌	高度情報科学技術研究機構 計算科学技術第2部
24 DEMによる内部構造を持つ複雑多相系の粒子モデル	坂口 秀	海洋研究開発機構 地球内部変動研究センター
25 宇宙の構造形成とダイナミクス	松元 亮治	千葉大学 理学部
26 乱流の世界最大規模直接計算とモデリングによる応用計算	金田 行雄	名古屋大学大学院 工学研究科
27 ドラッグデリバリーシステム(DDS)の大規模・ソフトマテリアル・シミュレーションに関する研究	宮内 教	高度情報科学技術研究機構 計算科学技術第1部
28 燃料電池の電極反応ナノシミュレーション	池庄司 民夫	産業技術総合研究所 計算科学研究所
29 戦略的革新シミュレーションソフトウェアの研究開発	渡邊 智彦	自然科学研究機構 核融合科学研究所
30 1700°C 級ガスタービン燃焼器開発のための乱流燃焼の予測シミュレーション	加藤 千幸	東京大学大学院 工学研究科
31 大規模シミュレーションによる原子炉内複雑熱流動挙動予測に関する研究	大島 伸行	北海道大学大学院 工学研究科
32 超伝導ナノアブレーションによる新奇物性と中性子核出射デバイス開発のための超伝導ダイナミクス研究	高瀬 和之	日本原子力研究開発機構 原子力基礎工学研究センター
33 耐放射線性SiCデバイス用酸化膜の第一原理分子動力学シミュレーション	町田 昌彦	東京大学大学院 理学系研究科
34 放射線照射に伴う材料の物性変化と破壊の微視的シミュレーション	宮下 敦巳	日本原子力研究開発機構 原子力基礎工学研究センター
35 高速増殖炉燃料集合体サブチャンネル内ナトリウム冷却材の直接シミュレーション	藤本 英雄	日本原子力研究開発機構 システム計算科学センター
36 全ゲノム・全タンパク質配列の自己組織化マップを用いた大規模ホストゲノム解析	二乃方 勇	東京大学大学院 理学系研究科
37 分子動力学シミュレーションを用いた大規模生体超分子系の模擬解析	池村 淑道	長浜バイオ大学 バイオサイエンス学部
38 巨大地震による広域沿岸被害シミュレーション	松本 淳	日本原子力研究開発機構 システム計算科学センター
39 高頻度経済データの経済物理学解析	高橋 重雄	香港空間技術研究所 津波防災研究センター
40 エネルギー注入による雲・降雨制御シミュレーション	高安 英佐子	東京工業大学大学院 総合理工学研究科
41 巨大蛋白質の高次構造変化のリアルタイムシミュレーション	森 雅裕	宇宙航空研究開発機構 総合技術研究本部
42 巨大蛋白質の高次構造変化のリアルタイムシミュレーション	齋藤 聡	弘前大学 理学部

国際共同研究等

共同研究(13件)

プロジェクト名	責任者名	所属
1 海洋全層同化モデルの開発とそれを用いた海洋大循環変動の研究	河野 健	海洋研究開発機構 地球環境フロンティア研究センター
2 Canada	Michel Desgagne	カナダ気象庁数値予報研究部 (RPN/MSC)
3 NERSO	Horst D Simon	アメリカ国立エネルギー研究センター
4 Scripps	Masao Kanamitsu	スクリpps海洋研究所 (SIO)
5 NCAS	Julia Slingo	The National Centre for Atmospheric Science
6 Hadley	Malcolm Roberts	Hadley-気候研究センター
6 LODYC-IPSL	Gurvan Madec	フランス国立科学研究所
7 CNRS(IFREMER)	Patrice Klein	フランス国立海洋開発研究所 (IFREMER)
8 CIRA	Pasquale Schiano	イタリア航空宇宙研究センター (CIRA)
9 Singapore	Lu Chun	HPCC-A*Star, Singapore
10 Taiwan	Guey-Shin Chang	台湾財団法人国家実験研究センター
11 超高度建築物の耐震性評価のための地盤連成を考慮した3次元シミュレーションに関する大阪大学とJAMSTECによる共同	豊田 政男	国立大学法人大阪大学 工学研究科
12 数値シミュレーション解析によるサイト特性を考慮した風力発電機性能評価技術の開発	高橋 桂子	海洋研究開発機構 地球シミュレーションセンター
13 粘性コントラストが著しく高い物質の不安定流動現象のシミュレーション技術開発に関する基礎研究	阪口 秀	海洋研究開発機構 地球内部変動研究センター

特定プロジェクト

21世紀気候変動予測革新プログラム(5件)

プロジェクト名	責任者名	所属
1 地球システム統合モデルによる長期気候変動予測	時田 達志	海洋研究開発機構 地球環境フロンティア研究センター
2 高解像度気候モデルによる近未来気候変動予測に関する研究	木本 昌秀	東京大学 気候システム研究センター
3 超高解像度大気モデルによる将来の極端現象の変化予測に関する研究	奥野 昭隆	気象庁 気象研究所
4 雲解像モデルの高度化とその全球モデル高精度化への利用	坪井 和久	名古屋大学 地球水循環研究センター
5 LESによる海洋微物理過程の高精度パラメタリゼーション	日比谷 紀之	東京大学 大学院理学系研究科

CRESTマルチスケール・マルチフィジックス現象の統合シミュレーション(6件)

プロジェクト名	責任者名	所属
1 全球電解像大気モデルの熱帯気候予測への実用化に関する研究	佐藤 正樹	東京大学大学院 理学系研究科
2 観測・計算を融合した階層連結地震・津波災害予測システム	松浦 孝忠	海洋研究開発機構 地球環境フロンティア研究センター
3 災害予測シミュレーションの高度化	高橋 桂子	海洋研究開発機構 地球シミュレーションセンター
4 超伝導新応用のためのマルチスケール・マルチフィジックスシミュレーションの基盤構築	町田 昌彦	日本原子力研究開発機構
5 海洋生態系モデル予測のための海洋環境シミュレーション研究	山中 康裕	北海道大学大学院 地球環境科学研究所

地球シミュレータ産業戦略利用プログラム(12件)

プロジェクト名	責任者名	所属
1 新幹線車両の空力騒音シミュレーション	栗田 健	東日本旅客鉄道㈱
2 CO2排出ミニムを目指した実高炉内の多相(固気液)流れの大規模シミュレーション	松崎 眞六	新日本製鐵㈱
3 ゴム中のナノ粒子ネットワーク構造のモデル構築による高性能タイヤの開発	皆川 康史	SRI研究開発㈱
4 機能性ナノ粒子設計シミュレーション	吉田 孝史	朝来化学
5 高効率有機発光材料の開発	菅前 康成	住友化学㈱
6 酸化炭素地下貯留に関する大規模シミュレーション技術の開発	山本 肇	大成建設㈱
7 非常常構造の特性解明およびそれに基づく抜本的空気抵抗低減技術の開発	炭谷 重二	トヨタ自動車㈱
8 電子機器ファンダクト系の空力騒音の数値解析	立原 昌義	キャノン㈱
9 遠心圧縮機の空力騒音低減の研究	坂田 信二	朝日プラントテクノロジー
10 超分子化合物半導体混晶の欠陥準位に関する第一原理電子状態計算	澤村 朝賢	住友電気工業㈱
11 数値シミュレーションによる回転機の高周波数値解析技術の開発	中村 雅志	東洋電機製造(株)
12 化学気相反応における触媒に関する大規模シミュレーション研究	藤澤 義和	脚本大技術研究所

とCPU数, さらに8ギガFlopsをかけると1秒間に計算できる演算数(ピーク性能値)となる。

縦軸の値は当該プログラムで1秒間に何回の浮動小数点演算が行われたかを示す実行レポートにあるハードウェアのカウント値から作成している。平均30%以上という効率は、スカラ一機に比べるとかなり高い数値である。

因みに2月号で報告したSC'07ゴードンベル賞ファイナルリストにノミネートされたNCAR「WRF Nature Run」のピーク性能比は、13.7%であった。

地球シミュレータの利用にあたっては、ノードの有効利用を推進するため、最初からたくさんのノードを使えるようにはしていない。これができないのであれば、50%16ノードまでは、自由に使えるが、

最大利用時間は2時間までである。

これ以上のリソースを使おうとすると自分のプログラムがその要求するノード数において50%以上の並列化効率を出すことを証明しなければならない。

すなわち、もし、32ノードを必要とするならば、1ノードで実行した場合の16倍の性能を最低でも出さなければならないのである。これができないのであれば、50%の効率を出すことができるノード

数までとなる。

また、地球シミュレータはベクトル機であるので全体の演算に占めるベクトル演算の割合が95%以上も条件となっている。要は、ノードを出来るだけ効率的に使うよう利用者に求めているのである。

何度もノード拡大申請を行って、結果として図4のような素晴らしい性能をもつプログラムが出現するのである。利用者にとっては、煩わしいかもしれないが平均性能が半分になれば、2倍の物量があることになるので疎かにできることではない。

このあたり、最近のパソコンの話で、メモリ不足すれば積めばいい、パワー不足すれば増設すればいいという風潮とは逆で、使い方を磨きなさいということである。

並列処理について

デュアルコア、クアッドコアのCPUチップを積んだパソコンを何万台もクラスタとして集めれば安い費用で巨大なスーパーコンピュータができるという議論がある。

事実、2月号で掲載した、TOP500で日本のトップは、東工大のTSUBAMEの16位であるが、このシステムは、NEC/SUNがOpteronを11,664個搭載し、ネットワークとしてinfinibandでインテグレーションを行ったシステムである。

地球シミュレータは、CPUとしてベクトルアーキテクチャの共有メモリ構成であるがシステム全体は、INと呼ばれる超高速ネットワークで接続されている。

TSUBAMEも地球シミュレータもどちらも大規模化にあたっては、並列にしていることに違いはない。

そこで大規模な並列化について何が困難となるのか以下に述べる。

本誌の読者は、情報処理の黎明期から関わっておられる方も多いと思うので直感的に(わかりやすい方法で)並列化の問題点について説明してみよう。図5参照。

1台のコンピュータを使ってある処理を行うのに、経過時間(ELAPSE)が10時間かかったとする。

この内訳を調べてみると、計算時間(CPU)が5時間、入出力時間(IO)が5時間であったとしよう。計算時間は、CPUの高速化、アルゴリズムの見直しで、速度が10倍になれば、計算時間が0.5Hになる。しかしながらELAPSEでは、5.5Hであるから、45%の改善にしかない。

もし加うるに、デバイスの高速化、プログラムの工夫(結果としてIO時間を隠蔽する)などで、IOが10倍速く改善できれば、ELAPSEでは改善は90%になる。これは、計算機の歴史そのものであり、当たり前の話である。

しかしながら一方でCPUが無限に早くなってもIO

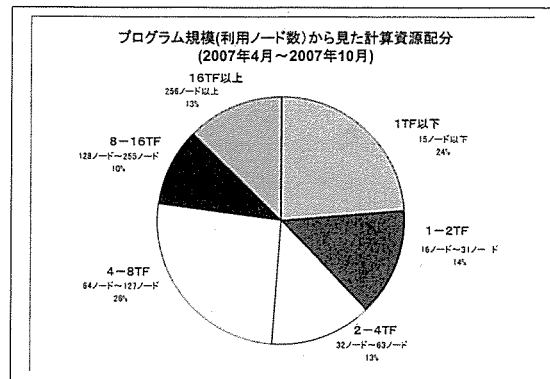


図3

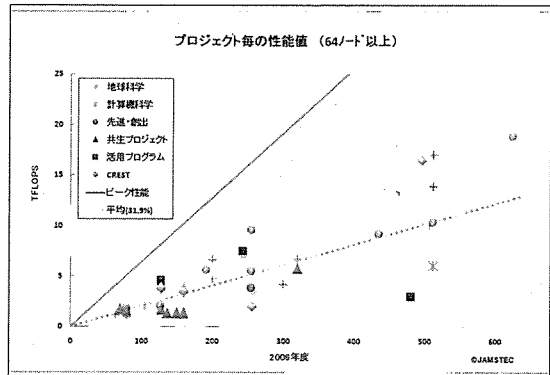


図4

が残る限り0.5Hより短くならないのも明らかであろう。スーパーコンピュータのプログラムの性能、効率の議論や、改善しようとする時に、このように壁となる存在が随所にてできて、如何にクリアしていくか腕の見せ所となる。

さて、チップの微細化が進み、今後クロックの劇的な向上が見込めなくなり、一つのCPUチップに複数のCPU機能を詰め込んだマル

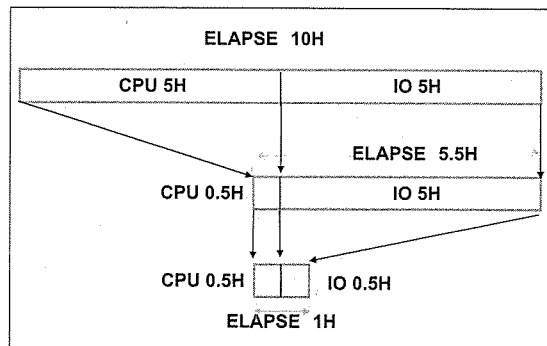


図5

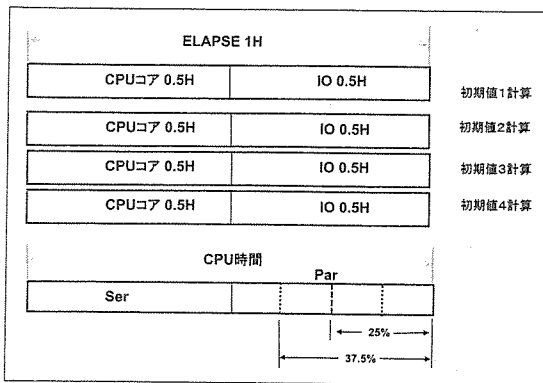


図 6

チコアで性能をあげていくのが最近の傾向である。

クアッドコア(CPU機能4個分)を考えてみよう。考えやすいのは、前記1HのELAPSEを要する処理について同時に4個の処理を走らせることである。ただし、4個の処理は、計算においても入出力においても干渉が無いものとする。

パソコン上で、インターネットから画像のダウンロードを行いつつ、MP3で音楽聞きながら、Wordで本誌の原稿を書いている、趣味の暗号計算もやっているようなものである(厳密には、メモリやディスク入出力の競合が発生するが)。もちろん、計算の初期値を変えて4通り同時に流す計算の処理でもよい。この方法では同じ1時間の中で4倍の仕事ができたことになる。図6参照。

ここで注意すべきは、ELAPSEの1Hは縮まらないということである。言うまでもないが、それぞれのCPUコアの(利用)効率は、50%である。さらに言うと、この図の計算では、実は2個のCPUコアで実行すると100%の効率を発揮できることがわかる。

さて、ELAPSEを縮めたいという要求も世の中に多々ある。家を建てる場合に設計は一人が担当し、

たら1年経っても完成しないが集中的に組上げることで工期短縮を図るのである。競争の激しい製造業の世界では、CADを駆使し、日単位で提案をしないと他社に負けるという厳しい環境がある。

この場合10日間で10倍のケースを計算できても、競争他社が2日間で提案してくれば、その優劣は明らかであろう。2日で計算した結果を相手に示して反応を見てからもう一回計算し提案しても、お釣りがくる勘定である。

家を建てる場合と同じく、計算の処理の中にも、順ぐりに計算しなければならぬ部分と同時並列的に流せる部分がある。

前者を逐次処理部分(Ser)、後者を並列処理部分(Par)と呼ぶことにする。入出力については、後段で説明する。図6下部参照。

Par部分は、いくつかの部分(n個)に分割できるとする。簡単のためにパラメータの異なる同じ処理と考える。n=2とすると図のParの時間は、並列に実行することができるのでCPU時間は、 $Ser + 1/2 Par$ となる。ほぼ25%短縮できることがわかる。では、n=4ではどうなるかということ、 $Ser + 1/4 Par$ となる。

これは、37.5%の短縮である。

材料のカットは工場に委託し、大工さんを一時的に集めて一気に組上げ、屋根、側壁ができた1、2名の大工さんでトンカチャというのが典型である。

最初から最後まで一人でやってい

n=100では、 $Ser + 1/100 Par$ で49.5%の短縮である。最初の数個のCPU増設による改善は、まあまあとしても100個のCPUにしても50%程度しか改善しないのでは、まことに勿体ないと言わざるを得ない。

これは、Ser部分がネックになり、いくら物量をつぎ込んでも意味がない(つぎ込んだ部分の効率は悪くなる一方だ)からである。

プログラムの中にどうしても並列化できない部分や入出力のように演算に比べると桁違いに遅いものが入ってくることは仕方がないところであるが、これらが邪魔になって性能が上がらず、効率も悪くなるのである。

並列化する場合の並列度と、プログラムの中で並列化できる部分の割合(並列化率)と、1個の時と比べてつぎ込んだCPUがどのくらいの効率で動くのか(並列化効率)の関係は、定式化したアムダールの名前を冠して「アムダールの法則」として知られている。

図6の下部をみれば、明らかであるが、オールマイティなのは、最初に述べたようにクロックを少しでも速くする(CPUの基本性能が高くなると全体を小さくすることができる)のが効果的である。式で表すのであれば、nを並列度、Parを並列化率とすると

$$\text{並列加速率} = 100 / \{Ser(\%) + Par(\%) / n\}$$

$$\text{並列化効率} = 100 \times \text{並列加速率} / n$$

逐次処理部分が5%とすると、1万個以上のCPUを使っても、並列加速率は20倍程度にしかならない。

当然、大部分のCPUはほとんど

遊んでいるのに等しい。注4参照。運用上、利用者がノード数を大規模化するのに、拡大申請を要求する由縁である。注5参照。

並列化に伴う問題は、この他にもいろいろある。たとえば、今述べた例では、Parの分割された部分は同一処理としたが、処理時間は、ばらつくのが普通である。

家を建てる場合の例であれば、何人かいる大工さんのそれぞれの進捗度の違いによっては、すべての大工さんの仕事が終わらないと次の工程に入れないような事態が生じる。

すなわち、一番遅い部分に引っ張られてしまうのである。これは、ロードインバランスと呼ばれている。また、数万個の並列処理においては、足並みを揃えるために必要なタイミングをどのようにとるのか、(同期)、また、分割したことにより、オーバーヘッド(主としてネットワークから発生)も無視できない。

オーバーヘッドにより元の場合より悪くなることすらある。

CPU性能に比べると入出力はあまり重要視されないが入出力も経過時間の一部をなす時間であるから疎かにはできない。CPUがうまく並列化できたとして、計算結果をどのように出力していくかということである。

CPUの並列度が仮に1,000とし(地球シミュレータでは、128ノード相当なので普通のサイズ)出力が各10ギガBとしてみよう。

この設定は3次元流体の時間発展を解く場合に普通に出てくる規模である。これをどこかで集中的にディスクに吐き出そうとすると、とんでもないことになる。

テラBオーダーのデータが集中してしまうことになるのである。

もちろん、たっぷりのメモリでバッファリングしていたとしても、少数のディスクへのアクセスの集中により、処理の滞りが

フィードバックされて、結果的に並列処理のCPUの処理を止めてしまうことになりかねないのである。

地球シミュレータでは、ノードごとに独立なRAIDを用意して並列処理への影響を最小限に留めるようにしている。これを実現するためには、処理に先立って、予めノードにデータを分散して、転送しておく必要があるし(ステージイン)、処理が終わった後は、ノードから逆に転送をする(ステージアウト)機能が必要となる。

ベクトル機とスカラー機

ベクトル機の特徴としては、メモリと演算器との間にキャッシュメモリを持たずにメモリから演算器のベクトルレジスタとの間を高いデータ転送能力を実現していることである。演算器に比べて相対的に遅いメモリアクセスに対しインターリーブを行うことにより、トータルなデータ転送量を確保している。

演算器は、内部の乗算器や加算器などに同時に切れ目なくデータを供給された場合に初めて所定の

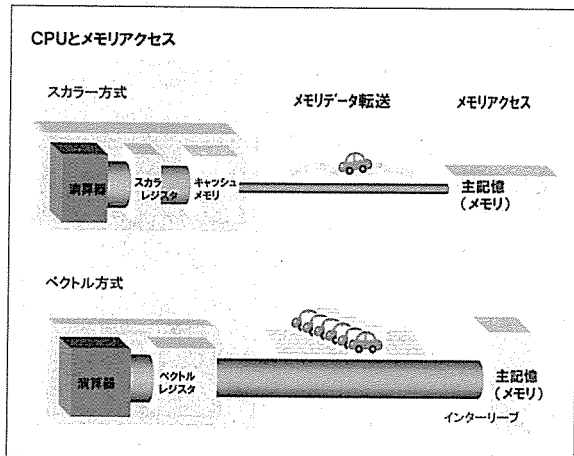


図7

性能を発揮することができる。地球シミュレータの場合、Doループの繰返し回数(ベクトル長)が256以上ないと最高性能の発揮は難しい。メモリからのデータ転送する場合に、できるだけアクセス要求がぶつからないようにしないと待ちが生じる。

これは、バンク・コンフリクトと呼ばれ、プログラマは配列要素の配置を意識しておく必要がある。単体性能の高いCPUを用いれば、並列度を高めなくても並列化の効果が出るのでプログラマにとっては負担が軽くなる。しかしながらこのような仕掛けを必要とするのでハードウェアのコストは大きくなるのが欠点である。

これに対しスカラー機は、メモリと演算器との間にキャッシュメモリを持っている。

このキャッシュメモリは高速アクセスが可能であり、データがキャッシュ上にあれば演算器は所定の性能を発揮することができる。

しかし、キャッシュメモリは、容量が小さいため、すべての処理において、データをキャッシュ上に置くようなことはできない。データがキャッシュになれば、遅

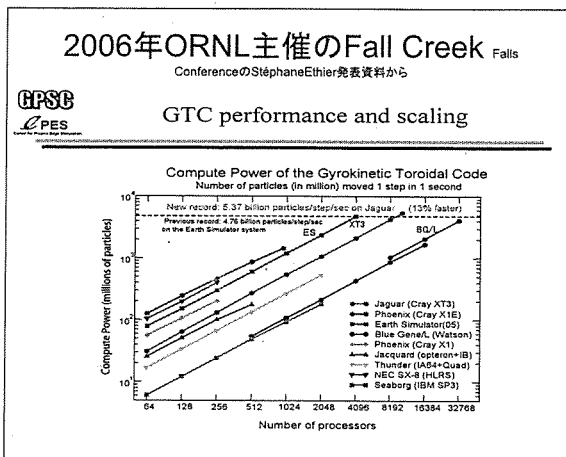


図 8

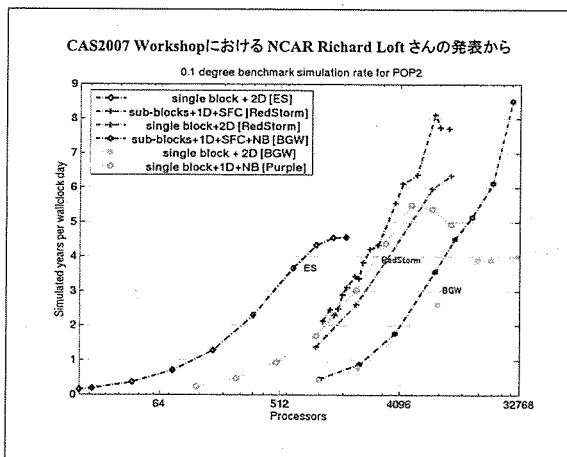


図 9

いメモリまで取りにいかなければならない。

またプログラマは大規模な処理においては、全体の性能向上を意図してデータをキャッシュ上に置くようなことは通常困難である。しかし、メモリ周りの作りは通常のパソコン同様で済むためハードウェアのコストは小さくなる。

ベクトル機にしてもスカラー機にしても、大規模なシステムを組むためにはネットワークが不可欠である。ネットワークの性能は、レーテンシ（遅れ時間）とスループット（データ転送量）が二大要素であるが、レーテンシは小さく、スループットは大きいに越したこ

いてる。図 8 は、GTCと呼ばれる地球シミュレータが比較的不得意とする粒子コードでのスカラー機との比較である。地球シミュレータへの最適化は、米国エネルギー省の研究機関であるNERSCが行った。注 6 参照。

CRAY XT3 の性能は図中の注釈にあるように、同一アルゴリズムで実行したものではない。32,768個のCPUを使用したBG/Lでも、4,096個CPUのESの処理に追いついていない。

図 9 は、POP (Parallel Ocean Program) と海洋のシミュレーションコードである。もともとスカラ

とはない。

ネットワークを流れるデータは、並列処理を行うプログラムの作り方、データの配置で大きく異なる。LINPACKベンチマークは、あまりデータの転送を伴わずハードウェアの性能を引き出しやすいベンチマークテストであるといえよう。

ベクトル機とスカラー機の比較

地球シミュレータも 6 年近くたち、スカラー機との比較データも出始めて

一パラレル機のために米国で開発されたコードである。地球シミュレータへは、IPCC第4次報告書へのNCARと共同研究を行った(財)電力中央研究所のチームが最適化を行った。

同じアルゴリズム “single block + 2D” のパフォーマンスを比べると RedStorm では 8 倍、BGW (BlueGene) では、30 倍の CPU 数が必要となっている。

並列化は、そのシステムに合ったアルゴリズムやデータの分割方法などが整合したときに初めて性能を引き出せることがわかる資料である。

参考文献

注 1

<http://www.jamstec.go.jp/es/jp/status/index.html>

注 2

<http://www.jamstec.go.jp/es/jp/project/general.html>

<http://www.jamstec.go.jp/es/jp/project/mext.html>

注 3

<http://www.es.jamstec.go.jp/projects/fy2007/index.html>

<http://www.jamstec.go.jp/seika/esdb/index.do?locale=ja>

注 4

<http://mikilab.doshisha.ac.jp/dia/smp/cluster2000/PDF/chapter06.pdf>

注 5

http://www.es.jamstec.go.jp/projects/apply/H20-1_sikaku.pdf

注 6

http://www.es.jamstec.go.jp/projects/international/2005_nersc.pdf