

# 今後のHPC環境における統計計算アルゴリズム

予測発見戦略研究センター データ同化グループ

教授 樋口 知之

## 1 計算機環境の今

### 1.1 “CPU” 成長神話の終焉

統計科学の研究において計算機を用いた数値計算実験が重要な役割を果たしていることは言うまでも無い。そこでは、陰に陽に仮定された確率の構造に由来する類似の計算操作（手順）が多数回繰り返され、その操作結果の集合、あるいは落ち着き先に興味があることが普通である。従って、統計科学の研究における計算機利用の大きな特長として、同質（類似）の計算操作数を増大させることにきりが無いという点がある。そうすると、計算機の計算スピードの著しい向上にユーザから永続的な期待感が寄せられるわけである。

計算機の計算スピード（正確には集積回路上のトランジスタ数）に関しては「18ヶ月ごとに倍になる」というムーアの法則が有名である。この予言をコンピュータの一般ユーザは単純にクロック数の向上に置き換えて理解し、事実、MHz時代からGHz時代への発展を同時体験したユーザは特に陥りやすい思考と言える。ところがこの数年、クロック数はほとんど向上していない（2001年からこの10年間にクロック数の増加は5倍にも満たない！特に2004年からはほぼ一定の淋しい状況）。むしろ、ベンダーはクロック数を現状維持、あるいはさらなる低減化を志向している。その理由は、クロック数の増大は指数関数的に熱量を増加させてしまうことにある。現在のCPU開発においては消費電力と熱（と音）への対策が最優先課題となってしまうている。

### 1.2 ビジネスモデルとリンクしたHPC (High Performance Computing)

このような環境の中で、計算機ユーザの計算処理速度に対する要求は日々高まる一方である。あまり明確には指摘されていないが、計算機ユーザはムーアの法則に従った計算機能力の増進を無意識に想定した上で各々の研究展開を行っているため、ユーザの要求とCPU能力との乖離は大きくなるばかりである。そこで登場したのが、CPUを複数、場合によっては多数つなぎ合わせるシステム、つまりマルチコア、メニーコアと呼ばれるCPUや計算機システムである（後述のGPGPUなどのマルチスレッド化も同様の方向性に沿った開発思考と言えよう）。スーパーコンピュータであれ同様の大きな流れに沿った設計で開発され、インテル系のCPUを複数ないしは膨大につながるシステム構築がベンダーの主たるビジネスモデルとなっている。独自にCPUを設計するのは膨大な時間とコストがかかり、そのコストを回収するためには商業的にも成功せねばならず、別次元の評価の視点が無い限り、ビジネス的には成立が相当厳しい。このあたりが、昨今話題となった次世代スーパーコンピュータの開発にからむ議論の底流となっている。日本のお家芸であったベクトル計算機は“戦艦大和”化しつつある（世界Top500の中でベクトル計算機数は1桁）のも、また共有型メモリ型の計算システムの計算速度向上があまり期待できないのも、同じ理由からである。

## 2 計算集約型統計計算手法の今後のアルゴリズム開発の姿

汎用CPUを超並列につないだ分散メモリ型にHPCのプラットフォームは確実に移っていく。このHPC環境の中で、計算機の集約的利用に基づく統計計算、つまり計算集約型統計計算の計算効率を高めるためには、HPCのハード特性を最初から意識した手法の開発が肝要となるであろう。考慮すべきハード特性としては、**階層性とネットワーク構造**の二つが特に重要である。

**階層性**：分散メモリ型のスーパーコンピュータは、複数の演算装置（コアと呼ばれる）を格納した物理的一要素単位—ノードと呼ばれる—を、高速の通信ネットワーク（インターコネクトと呼ばれる）で結んだ総体である。このノード内にはたいてい16GBから48GBのメモリが付随している。ノード間通信は複数の形式がありえるが（予算しだい）、その速度はコアとメモリ間の通信速度と比較すると圧倒的に遅い。よって、ノード内で計算が済む部分と、ノードを超えた計算が必要な部分に明確に分離できるアルゴリズムが有利である。つまり、ノード内のメモリサイズの制限内で、可能な限りノード内計算にすむように工夫し、精度等の計算パフォーマンスの観点から明らかに望ましいノードを超えた計算操作をたまに実行するように、最初からアルゴリズムを設計すべきである。さらに言えば、ノード内の並列化はフラットMPI等でなくOpenMPを利用した並列化を、またノードを超えた並列化はフラットMPIで対処するといった、ハイブリッドタイプの並列化が好ましい場合も多い。我々データ同化グループがアプリケーションソフトの開発面でかかわっている次世代スーパーコンピュータでもハイブリッド並列化が推奨されている。

**ネットワーク構造**：各ノードを結ぶ形式、つまり通信ネットワーク構造にも十分注意を払わなければならない。次世代スーパーコンピュータは、3次元トーラスの直接結合網（ただし6方向で結合）である。Cray社のスーパーコンピュータも3次元トーラスのネットワーク構造を持つ。直接結合の場合は、なるべく近いノード間で通信（データ転送）を行うほうが明らかに理にかなっており、離散化された偏微分方程式の時間発展を求めるシミュレーションコードを計算機に実装する際には、空間的に近い離散量を近いノードに割り当てるなどの工夫が実効的である。一方、統計計算においてはシミュレーション時の空間のような自明な距離がないため、ノード間通信の負荷を念頭においた、各ノード上のメモリの利用法を考えねばならない。すなわち、ソースプログラム内の各変数をノードにどのように割り振るかに細心の注意が必要である。

### 3 GPGPUのインパクト

GPGPUとはGeneral-purpose computing on graphics processing unitsの略で、画像処理を専門とする補助演算装置であるGPUを画像処理以外の一般的な計算に用いる技術の総称である。GPUのクロック数自体はCPUと比較すると見劣りする（1.3GHz程度。CPUの約1/2~1/3）が、一つのインストラクションを多数の異なるデータ（統計でのデータの意味でなく、計算機科学でのデータの意味）に同時に適用する操作、つまりSIMD型タイプとしては著しく性能が良い。その理由は、一つのGPU内にスレッドプロセッサと呼ばれる多数（128~240個）の演算装置があり、それらが同時並列にデータ処理を行う点にある。ただしMPIのような各コアに違う操作を分担してもらうことはできない。このように多数のデータに対して同じ操作を加えることに特化しているため、著しく高速化できるアルゴリズムは限られている。また、現時点においては、単精度計算のみが計算スピードが速く、倍精度製品はCPUと比較するとさほど魅力的でない。さらにはメモリサイズもCPUと比べると相当小さい。

このように語るとGPGPUはゲームマシンへの用途からさほど広がっていないように思われるかもしれないが、統計科学には『違うデータに同じ処理を適用する操作を膨大に繰り返す』計算が数多くある。まずブートストラップがすぐに思いつく。実際すでに我々はGPGPUを用いて20数倍もの計算高速化に成功している。粒子フィルタにも適用済みであり（林 他(2009)）、これまで非効率であると思われ、捨て去られていたアルゴリズムに再度、陽が当たる可能性がでてきた。

### 参考文献

林圭佐、齋藤正也、吉田亮、樋口知之. (2009). GPGPU を活用したバイオロジカルパスウェイモデルの高速なパラメータ推定、情報処理学会研究報告, 2009-BIO-19, 19, 1—3.