

ミニ・ミニ・スーパーコンピュータ 2

— 作ってみよう・使ってみよう —

村上佳久

筑波技術大学 障害者高等教育研究支援センター 障害者基礎教育研究部（視覚障害系）

要旨：スーパーコンピュータ（以下：スパコン）の超小型タイプを製作し、視覚障害者がスパコンを利用するに当たってのプログラミングの手法などを原寸大で教育できるようなカリキュラムを開発し、検証した。実際、作製した超小型スパコンは、東京工業大学の大型スパコンである“Tsubame 2.0”とソフトウェア面で互換となるように設計されており、現行の“Tsubame 2.5”でも動作するように配慮され、将来的な発展が見込めるようなものが実現できた。

キーワード：視覚障害、Tsubame 2.5, GPGPU, Tesla, プログラミング

1. はじめに

現在、スパコンは様々な業務に利用されるようになり、気象予測シミュレーションや流体力学、化学結合や分子生物学などに活用されるようになってきた。例えば、東京工業大学の大規模スパコンである“Tsubame”は、「みんなのスパコン」として、東京工業大学の学生以外にも利用され、稼働率は2012年度の繁忙期（11～2月）で99%に達したと発表している。[1]

自前のスパコン導入が不可能な筑波技術大学では、東京工業大学の大規模スパコンのミニミニタイプを製作し、できるだけソフトウェア互換のシステムで、スパコンのソフトウェア学習を行い、プログラミングを検証して、最終的に他大学のスパコンを利用して、その成果を検証するほかはない。そこで本研究では、視覚障害者が、スパコンを利用するに当たり、スパコン使用のプログラミングについて、スパコン利用のプログラミング技術を習得するための、基礎から応用までの段階を得た教育手法を検討し、実践することを目的とし、併せて前回のミニミニスパコンを改良したので報告する。

2. 前回のミニミニスパコンの改良

前回の2台目のミニミニスパコンのスペックは次の通りであった。

M/B:ASUS P9-X79E WS
CPU:Intel Xeon E5-2670
GPU:nVIDIA GeForce 570
GPGPU:nVIDIA Tesla C2070 x2
RAM:64GB
HDD:1TB + 2TB

P/S:1250W Gold

OS:Windows 7 Enterprise 64bit

OS:SuSE Linux Enterprise Server

このシステムで動作検証を行うため、2日間の負荷テストを実施した。

負荷条件：超高負荷（95%以上負荷）

電源負荷：1200W（96%負荷）

CPU負荷：All Core, 115W（100%負荷）

GPU負荷：250W（98%負荷）

GPGPU負荷：250W x2（98%負荷）

メモリ負荷：Wright and Read（95%負荷）

HDD負荷：連続読み書き処理（95%負荷）

GPUとGPGPU、メモリの負荷は100%に出来なかったため、最大の負荷を駆けた。電源は、この状態で達した連続運用値での最大負荷である。

- 1) 電源高負荷による自動停止
- 2) HDD 高温化
- 3) CPU 高温化
- 4) GPGPU 高温化
- 5) マザーボード高温化

電源やHDDやCPU、メモリ等は、高負荷・連続負荷に耐えるように、サーバ用途の高可用性の部品を利用しており、耐久性には問題ないと思われたが、このような超高負荷を2日間（48時間）実施したところ、現実に様々な問題が発生した。

相性保証を活用して、これらの対応を行うため、一部の

部品を交換して再設計し直すこととした。

M/B:ASUS P9-X79E WS
CPU:Intel Xeon E5-2660 v2
GPU:nVIDIA GeForce GTX570
GPGPU:nVIDIA Tesla C2070 x2
RAM:64GB
SSHD:1TB
HDD:1TB
P/S:1500W Gold
OS:Windows 7 Professional 64bit

様々な負荷について再検討を行った。電源負荷はサーバ用なので、95%を超える負荷が数時間続いても問題ないはずであるが、50時間近い高負荷が続く場合を想定して、もう少し容量の大きいものを選択する必要がある。

CPUについては、1段階低電圧仕様に変更し、物理Coreを10coreのものに変更し、CPUクーラーをより大型のものに取り替えて、超高負荷に対応した。高負荷時の温度上昇が若干小さくなった。

また、HDDの負荷を下げるため、システム用をSSHD(HDDとSSDのハイブリッド)とした。入力のキャッシュ部分を内蔵のSSDが対応し、HDDのデータ転送をキャッシュするため、高負荷時における温度上昇を避けることが可能となる。また、データ用HDDをサーバ用HDDの省電力仕様に変更した。

最大の問題はGPGPUの温度上昇である。GPGPU2台とGPUが隣接しているため、中央部分のGPGPUは空冷ファンで冷却しても、冷却が十分ではなく温度は100℃以上となり、高負荷状態ではかなり危険となるため、空冷ファンの回転数を多くして風量を大きくして対応した。

さらに、電源部を1500W Gold(瞬間ピーク時1800W)に変更し、最大負荷時の電力供給に余裕を持たせた。マザーボードの高温化は、12cmの空冷ファンを増やし対応した。

これらの改良によって、5日連続程度の高負荷稼働でも安定動作を確保できるものと思われる。写真1に改良されたミニミニスパコンを示す。

3. スパコンプログラミング

スーパーコンピュータのプログラミングについて様々な手法があるが、実際にプログラミングを組むためには、その手法を理解する必要がある。そこで、スパコンのプログラミングに必要な様々な大規模計算時の手法について整理する。

3.1 OSの選択

スパコンにおけるプログラミングには、OSの選択が重要と

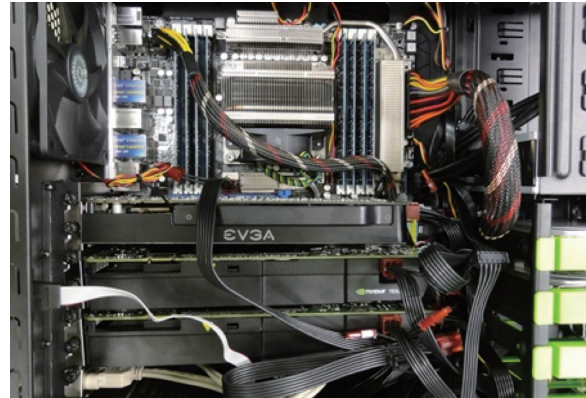


写真1 ミニミニスパコン
(GeForce GTX570, Tesla C2070 x2 搭載)

なる。東京工業大学のスパコン“Tsubame”の場合は、Linux計算ノードとWindows計算ノードを有している。

Linux計算ノードには、SUSE Linux Enterprise Server 11 SP1と合わせて、マルチサーバに対応するOpenMPIが導入されている。このMPI(Message Passing Interface)は、ネットワーク上の複数の計算機に分散して計算させる方法で、プログラミングを最適化出来る。

Windows計算ノードは、Windows HPC Server 2008 R2と合わせて、MS社のMPIが組み合わされている。

両計算ノード共に、大規模な複数台のシステムで並列分散処理が可能なOSとそれに対応した様々なコンパイラが導入されており、スパコン利用のプログラミングを容易にしている。

一方、今回のミニミニスパコンでは、1台で構成されるため、OpenMPと呼ばれる、マルチ・プロセスやマルチ・スレッドに対応するシステムをOSと共に併用する。

OSは安定度が要求されるため、最新のOSよりも枯れたシステムの方が安心である。また、ワークステーション用OSかサーバ用OSかの選択であるが、今回の様に1個のCPUで、メモリが64GB程度では、ワークステーション用のOSの方がサーバ用OSに比べて、高速化が期待出来る。また、OSに付随している大規模ビジネス用のセキュリティ対策等は不要であり、余分な負荷となりうる。そこで、Windows 7 Professionalに変更し、OpenMPと共に利用した。

3.2 Multi CPUによるプログラミング

最近のパソコンに利用されるCPUは1個のCPUに数個のCoreと呼ばれるCPU本体が内蔵され、一体化となっている。例えば今回のミニミニスパコンのCPUである、Intel Xeon E5-2660 v2は、1個のCPU内に10個のCoreを有している。

はじめに、この複数個のCoreを利用する場合のプログラ

ミングを考える。CPU に複数の処理を同時にさせる場合の分割・処理の方法として、メモリを別々に利用する「プロセス」と単一に利用する「スレッド」と呼ばれる単位で動作する。

同じような処理であれば、「スレッド」単位で動作させた方が、メモリ効率も良く、高速化が期待できる。

一方、CPU の方でも、Core 単位で動作させるか、スレッド単位で動作させるかを選択できるようになってきた。

Intel 社の Hyper thread では、1つの Core で2つのスレッドを動作させることが可能となる。多くのスレッドで動作させる場合、この機能を利用することによって、プログラム利用効率を上げることが可能となる。今回の場合は、1CPU 当たり 10Core なので 20 スレッドが動作可能である。この多数のスレッドに対応したプログラミングを行えば、並列処理の第 1 歩となる。写真2に、CPU の 20 スレッドの負荷状態を示す。上段 20 個が CPU スレッドである。

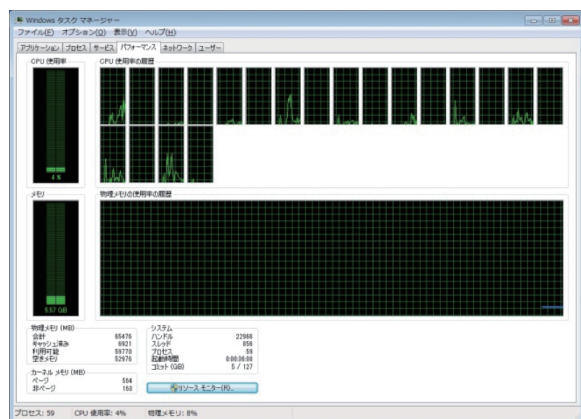


写真2 CPU スレッド使用率
(Windows 7 Professional タスクマネージャー)

3.3 GPU によるプログラミング

GPU を利用して高速化並列プログラミングに利用する試みは前にも述べたが、実際のプログラミングはどうか。

nVIDIA 社から GPU を GPGPU として動作させるためのツールとして、CUDA と呼ばれる一連のツール類が発表されている。これらのツール類を、プログラム開発キットに組み込むことにより、GPU を利用したプログラミングが利用可能となる。欠点としては、GPU を利用するため、画面出力とプログラミングが平行して GPU で処理されることになり、プログラミング効率が若干悪い。出来れば、並列化処理の部分は専用のプロセッサに担当させた方が、効率が良いので、nVIDIA 社では、GPGPU 専用の Tesla と呼ばれるプロセッサを用意している。

ハードウェアを実装してから、開発ソフトウェアを導入し、最後に CUDA 関連のライブラリーを導入することにより、開発環境は整えられるが、GPU を GPGPU として動作させる

場合には、Multi CPU の時とは異なるプログラミング手法が必要である。CUDA には、様々なサンプルプログラムがあり、それらを使いながら、一般のプログラム開発とは異なった手法を理解していくことが、プログラム習得の早道である。

基本的には、ソースコード中の CPU で動作する部分と GPU で動作する部分をそれぞれ分けて、CPU で動作させる部分のコードを C コンパイラに渡し、GPU で動作させる部分のコードを GPU 用にコンパイルするのが CUDA ライブラリーの働きである。したがって、CUDA ライブラリーは、GPU プログラミングの部分だけを GPU 用にコンパイルしているだけで、実際のプログラミングは C のコンパイラが担当するため、今回の場合、プログラム開発ツールである Microsoft 社の Visual Studio が行う。

したがって、プログラミングに関して記述の仕方が同じような表現であっても、CPU 用と GPU 用ではソースコードの書き方が異なる。

3.4 GPGPU によるプログラミング

GPGPU (General-purpose computing on graphics processing units) とは、GPU の演算資源を画像処理以外の目的に応用する技術のことで、GPU を GPGPU 化し専用の計算プロセッサとする傾向はここ数年、非常に進化している。

前項の GPU によるプログラミングよりも、GPGPU 専用によるプログラミングでは、GPGPU に搭載されたメモリを活用し、プログラミングを行う方が最も効率が良いため、GPGPU に搭載されるメモリの容量によって、計算能力が飛躍的に向上する場合もある。

GPGPU によるプログラミングでは、GPU プログラミングに加えて、GPGPU のグラフィックメモリをメモリとして活用し、GPGPU 単体での演算速度を向上させる手法を習得する必要がある。nVIDIA 社の CUDA の最新版では、この GPGPU のメモリ展開に対応しており、OS と OpenMP とは別にメモリを有する GPGPU の性能を引き出すことが可能となる。GPGPU のメモリは今回のミニミニスパコンでは、1ユニット当たり 6GB 程度であるので、そのメモリで処理出来る範囲内でプログラムを処理させる場合には、高速化が期待出来る。

3.5 Multi GPGPU によるプログラミング

次に、GPGPU を複数台活用するプログラミングである。GPGPU によるプログラムを複数の GPGPU に展開して並列処理させると、高速化が期待出来る。今回のミニミニスパコンでは2台の GPGPU を搭載している。そこで、GPGPU 並列化処理を最終的に導入し、並列処理の効率化を高めるプログラミングを習得する。

4. スパコンプログラミングの教材化

4.1 プログラミングの流れ

前章でスパコンのプログラミングを整理したが、視覚障害者がスパコンを利用してプログラミングを行う場合の教材化について検討する。

スパコンを利用する前に、C言語でのプログラミングがある程度出来ていることを学習の前提とする。プログラミングが身についていない場合には、このようなスパコンのプログラミングは行うべきではなく、最初にC言語を基本言語として習得する必要がある。

次に、ミニミニスパコンのハードウェア構成を理解することである。スパコンは、ハードウェア構成毎にプログラミングの手法が異なる場合がある。今回の場合でも、東京工業大学のスパコンである”Tsubame”とプログラム互換になる様にハードウェアを設計しているが、全く同一ではない。したがって、ハードウェアを理解した上で、スパコンプログラミングの学習を行う必要がある。

プログラミングについては、前項の通り、以下の順番で、基本的なプログラミングを習得する。

- 1) Single CPU プログラミング
- 2) Multi CPU プログラミング
- 3) GPU プログラミング
- 4) GPGPU プログラミング
- 5) Multi GPGPU プログラミング

サンプルプログラムを始めるの手がかりとして、10000までの素数を求めるプログラミングや π を10000桁求めるプログラミングなど様々な事例を数種類、1) から4) の手法に沿って習得することにより、スパコンのプログラミング手法を習得させる方法である。

実際にプログラミングを1) から5) で行ってみると、全てのプログラムで5) が最も高速なわけではない。場合によっては、1) が最も速い場合もある。

どの手法が最も高速で動作するプログラムとなるかは、実際に比較してみないとわからないことが多く、場合によってはとりあえずプログラミングを行って、デバッグ時に高速化を検討した方が良い場合もある。

したがって、まずスパコンにおける並列計算の手法を学習した上で、様々なプログラミングについてケースバイケースで複雑化を行っていく。

4.2 プログラミングと視覚障害補償

このスパコンプログラミングにおいて、視覚障害者に対して配慮することは何であろうか？

一般にプログラミングは、3つの過程から成る。

- 1) コーディング

- 2) デバッグ

- 3) 修正

基本的に1) のコーディングは、プログラム書くことであるから、Windowsに標準搭載されている、テキストエディタである「メモ帳」程度で十分である。

画面読み合成音声ソフトに対応したエディタ上から、直接プログラム開発ソフトの命令をコントロール出来るエディタがあれば良いのであるが、現状ではそのようなソフトは存在しない。しかし、メモ帳であれば、画面読み合成音声に対応するため、合成音声による出力や場合によっては点字ディスプレイの併用も可能である。

また、参照するサンプルプログラムも、テキストデータなのでメモ帳であれば、簡単に参照することが可能である。

次に、コーディングが終われば、プログラム開発ソフトでコンパイルを行う。この場合、Microsoft社のVisual Studioを利用するので、最低限ファイルを読み込んで、「ビルド」と呼ばれるコンパイルが実行できれば良い。従って、このコンパイル時点までは合成音声による対応が可能である。

その後、2) デバッグであるが、デバッグに必要な情報は、コンパイラが実行中の2つの画面の内容が出力されれば良い。エラーメッセージとビルドメッセージである。

エラーメッセージは、プログラムエラーに対するコンパイラのメッセージであり、ビルドメッセージは、コンパイルしている順序に従って提示されるコンパイルの状況である。この2つのファイルが参照できれば、リアルタイムにコンパイラの出力を得られなくとも、デバッグは可能である。つまり、リアルタイムのデバッグをあきらめて、エラーファイルを参照しながら、デバッグを進めていく手法である。スパコンのプログラミングでは、GPGPUやCPUを組み合わせて利用しているため、どの部分でのプログラムエラーなのかを把握することが最も重要である。したがって、リアルタイムである必要は無く、その部分のエラーが把握できれば良い。

二つのエラーメッセージを、標準出力でテキストファイルとして出力できれば、メモ帳で参照可能である。Visual Studioの設定でファイル出力設定を行い、この手法に従ってデバッグを行えば、3) 修正も容易となる。

4.3 教材化

教材化に当たっては、前々項のプログラミングの流れと、サンプルプログラム、前項のプログラミングと視覚障害補償方法、そして、第2章のハードウェア構成についての理解の3つを柱として、教科書として流れを整理する。

ハードウェアの理解なしにスパコンのプログラミングは不可能である。また、スパコンの1) ~ 5) のプログラミングの流れは、サンプルプログラムと共に学習しないと、なぜこのような手法でプログラムを書くのが、理解できなくなる。あくま

でも、スパコンという特殊なプログラミングを学習するための手法であることを理解させなければならない。

また、デバッグも重要で、なぜこのプログラムの書き方ではだめなのかを、エラーメッセージを熟読することにより理解させなくてはならない。

一般のプログラム開発では、少々の間違いはコンパイラが修正するが、スパコンの場合はハードウェアを指定して計算させるため、間違ったプログラムは非常に低速なプログラムになり、スパコンを使う意味が無い。

このようなことを学習させるためにも、学習の流れを明示することは重要であり、その意味でもハードウェア構成の順番に学習することは大切である。

5. おわりに

近年、スパコンの需要が大きくなり、多くの分野でスパコンが利用されるようになってきた。本来スパコンは、特定用途向けであり、一般的な計算には向かず、プログラミングも特殊である。

また、本ミニミニスパコンは、CPU など全部品を入手可能な秋葉原で購入して組み立てたが、部品も高価なため、3年半に渡って部品を購入して組み立てたものである。実際

に組み立てにかかった部品代だけは100万円を超える額であった。高価ではあっても、今回のミニミニスパコンも、特定用途向けのワークステーションではないため、純粋にスパコン学習用の専用ミニミニスパコンである。

「大型スパコンを操れるようになろう」と言う高い目標を掲げ、このようなミニミニスパコンで学習することは、今後のスパコンの発展を考えると、意味のあることと考える。

視覚に障害があっても、この手法によって実際にスパコンを活用できるようになり、東京工業大学のような大型のスパコンで、実際に自分の組んだプログラムがどの程度の早さで処理できるかを試してみたいものである。

そうすれば、視覚障害者のプログラミングの新しい可能性が開けるものと信じてやまない。

参考文献

- [1] 東京工業大学 学術国際情報センタープレスリリース, www.titech.ac.jp/news/file/pr20130716_tsubame.pdf
- [2] 村上佳久. ミニ・ミニ・スーパーコンピュータ -作ってみよう-. 筑波技術大学テクノレポート. 2013;21(1):p58-63.

A Micro Supercomputer (2)

MURAKAMI Yoshihisa

Research and Support Center on Higher Education for the Hearing and Visually Impaired,
Tsukuba University of Technology

Abstract: A micro supercomputer was created as an educational tool for visually impaired students. Curriculum development of the programming learning for visual impaired students to use a super computer was tried. This micro supercomputer was designed so that "Tsubame 2.5" that was the newest supercomputer of Tokyo Institute of Technology and the program may become interchangeable.

Keywords: Tsubame 2.5, General-purpose computing on graphics processing unit, Tesla, Programing