

教育用端末を用いた並列計算環境の構築

田實浩一[†]，久保田真一郎[‡]，升屋正人[‡]

[†] 鹿児島大学大学院理工学研究科情報工学専攻 〒 890-0065 鹿児島市郡元 1-21-40

[‡] 鹿児島大学学術情報基盤センター 〒 890-0065 鹿児島市郡元 1-21-35

E-mail: [†]tajitsu@bio.cc.kagoshima-u.ac.jp, [‡]{kubota, masatom}@cc.kagoshima-u.ac.jp

あらまし 並列プログラミングの演習に利用することを目的として，MPICH2 を用いて教育用パソコン端末上に並列計算環境を構築した．本環境はオペレーティングシステムとして Windows XP を使用し，他の授業や自習で使用する環境をほとんど変更する必要がない．特殊なハードウェアやソフトウェアを必要としないため，一般的なパソコンが設置されている教室であればほとんど費用をかけずに導入が可能である．構築の手順及び並列計算機としての性能を検証した結果を報告する．

キーワード 並列プログラミング，情報教育，MPI，MPICH

PCEC: Parallel Computing Environment on the Classroom

Koichi TAJITSU[†]，Shinichiro KUBOTA[‡]，Masato MASUYA[‡]

[†]Department of Information and Computer Science, Graduate School of Science and Engineering, Kagoshima University, 1-21-40, Korimoto, Kagoshima, 890-0065 Japan

[‡]Computing and Communications Center, Kagoshima University, 1-21-35, Korimoto, Kagoshima, 890-0065 Japan

E-mail: [†]tajitsu@bio.cc.kagoshima-u.ac.jp, [‡]{kubota, masatom}@cc.kagoshima-u.ac.jp

Abstract We have developed the PCEC, Parallel Computing Environment on the Classroom, for educations of parallel programming. The PCEC consists of average Windows XP PCs and additional costs are not necessary. PCEC adapted MPICH2, the MPI-2 implementation from Argonne National Laboratory, in order to run MPI applications. In this report, we introduce the procedure of PCEC installation and show a benchmarking result.

Keywords Parallel programming, Information Study, MPI, MPICH

1 はじめに

クロック周波数の向上による計算機の演算性能の向上には限界があるため，近年の CPU の性能向上のかなりの部分が内部の命令レベルの並列性を高めることによって達成されている．しかし，命令レベルの並列性による性能向上は限界に近い．このため，複数の CPU による並列性の向上によって

演算性能を向上させることが一般的になりつつある．こうした複数の CPU により構成される並列計算機には，大きく分けて共有メモリ型と分散メモリ型がある．最近では，複数のパソコンをネットワークで結ぶことにより実現される分散メモリ型並列計算機 — PC クラスタ — が広く用いられるようになってきた．

プログラムを作成して PC クラスタを活用する

には、並列プログラミングの知識が不可欠である。共有メモリ型並列計算機であれば、まだ十分でないものの、自動並列化機能を持ったコンパイラが存在する。これを使えば、共有メモリ型並列計算機を用いた並列計算を比較的容易に実現できる。一方、PC クラスタのような分散メモリ型並列計算機の場合は、MPI[1] や PVM[2] などのメッセージパッシングの仕組みを用いるのが一般的である。これらの仕組みについて知らなければ分散メモリ型並列計算機を並列計算機として稼働させることはできない。

PC クラスタは科学技術計算以外にも様々な分野で活用されつつある。情報系の学校・学部・学科のみならず、コンピュータプログラミングの教育を実施しているすべての教育組織において、並列プログラミングへの取り組みが近い将来必要になるものと思われる。

ところが、これら分散メモリ型並列計算機上の並列プログラミングの教育を行うことができるのは、PC クラスタを有する一部の大学や研究室に限られる。そうした大学や研究室であっても PC クラスタは専ら高度な科学技術計算のために利用されており、教育目的、特に学部レベルの教育に利用できる場合は少ない。高性能な CPU を用いた本格的な並列計算機システムに比べて安価ではあるものの、PC クラスタを新規に導入するには経費が必要である。このため、教育目的で PC クラスタを導入する組織は少ない。

そこでわれわれは、特別な装置や高度な技術を用いず、パソコン教室などに設置してあるネットワークに接続した複数台のパソコンのみで並列計算環境 PCEC(Parallel Computing Environment on the Classroom) を構築することにした。PCEC は既に設置されているパソコンを用いて容易に構築できるため、多くの大学・高校等で応用できる。これにより、現在ほとんど行われていない PC クラスタを活用するための並列プログラミングの教育を多くの教育機関で実施できるようになるものと考えている。

端末室のパソコンを用いて並列計算環境を構築した例としては、パソコン教室を夜間に PC クラ

スタ化する庄司ら [3] の研究がある。このシステムは夜間に自動的に再起動して Linux による PC クラスタを構築し翌朝にはシャットダウンする、という仕組みを用い、科学技術計算用の PC クラスタを端末室のパソコンの遊休時間を利用して実現するものである。PCEC は、科学技術計算そのものではなく並列プログラミングの教育を目的としているほか、授業や自習で利用する Windows XP の環境のまま利用できるなど、庄司らのものとは目的も実現方法も大きく異なる。庄司らのシステムを教育に応用することも不可能ではないと思うが、自動的に再起動やシャットダウンする仕組みやネットワークブートの仕組みをパソコン教室に追加しなければならず費用が必要になるほか、システム管理者の負担が大きくなってしまう。PCEC であれば、既存のパソコン教室に無償ソフトウェアを追加するだけであるので、費用や導入後の負担は全く不要である。

2 PCEC の概要

PCEC は、ネットワークに接続された複数台のパソコンから構成される。Windows XP をオペレーティングシステムとして使用しているパソコンを用いて問題なく構築できることを確認した。検証は行っていないが、Windows XP 独自の機能は用いていないので Windows 2000 をオペレーティングシステムとした場合でも同様に構築できるものと思う。こうした環境は多くの教育機関でパソコン教室として整備されているはずである。本研究では、鹿児島大学学術情報基盤センター [4] の端末室に設置されているパソコンを用いて PCEC の構築を試み、PC クラスタとしての性能について検証を行った。

本研究で用いたパソコンの仕様を表 1 に示す。

表 1 のパソコンは表 2 のスイッチングハブに 1000BASE-T で接続している。

すでに授業や自習で利用されているこの環境に、MPICH2[5] をさらに導入することで、PCEC を構築できる。実行環境だけであれば、いくつかの dll

表 1: PCEC 構築テストに用いたパソコン

メーカー・型番	富士通・FMV-E630
OS	Microsoft Windows XP Professional SP2
CPU	Intel Pentium4 2.80AGHz
L2 キャッシュ	1MB
チップセット	Intel 865G
FSB/メモリバス	533 MHz / 333 MHz
メモリ	512 MB (うち 8 MB をビデオメモリとして利用)
ハードディスク	80 GB (Ultra ATA 100, 7200 rpm)
LAN	1000BASE-T/100BASE-TX/10BASE-T

表 2: PCEC 構築テストに用いたスイッチングハブ

メーカー・型番	Allied-Telesis・CentreCOM GS916GT
準拠規格	IEEE802.3u(100BASE-TX), IEEE802.3ab(1000BASE-T), IEEE802.3x(Flow Control)
スループット (1000Mbps)	1,453,489 pps (64 バイト)
スイッチング遅延	2.1 μ 秒 (64 バイト)
スイッチング・ファブリック	32 Gbps
パケットバッファ	256 K バイト

ファイルと smpd.exe をコピーして実行するだけでよく、短時間で環境を構築することが可能である。Windows XP 上で MPI による並列計算環境を構築する方法はいくつかあるが、無償であることと、この簡便さから PCEC では MPICH2 を用いることにした。

2.1 PCEC における MPICH2 の導入

Windows への MPICH2 の導入は、MPICH2 の Win32 パイナリ (平成 17 年 11 月 10 日現在、mpich2-1.0.2-1-win32-ia32.msi) を MPICH2 の WWW サイト [5] からダウンロードし、PCEC を構成するパソコンにインストールするだけでいい。これにより、実行環境と開発に必要なインクルードファイルやライブラリファイルがインストールされ、設定が行われる。

インクルードファイルは %Program Files%\MPICH2\include に、ライブラリファイルは %Program Files%\MPICH2\lib に、実行に必要な DLL は Windows の system32 ディレクトリにインストールされる。

実行命令を待ち受ける smpd.exe やプログラムを実行するために用いる mpiexec.exe のほか、GUI ツールの wmpiexec.exe, wmpiconfig.exe などが %Program Files%\MPICH2\bin にインストールされる。

開発環境が不要な場合や、システムディレクトリへの書き込みが制限されている場合は、以下の手順でファイルコピーにより実行環境のみを構築することもできる。

1. MPICH2 を任意のパソコンにインストールし、%Program Files%\MPICH2\bin から、mpiexec.exe と smpd.exe, system32 ディレクトリから mpich2*.dll (mpich2 でファイル名が始まるすべての DLL ファイル) と fmpich2*.dll (fmpich2 でファイル名が始まるすべての DLL ファイル) を USB メモリなどにコピーする。ファイルサーバがある場合には、ファイルサーバに置いてもいい。
2. PCEC を構成するパソコンを起動して C:\%MPICH2 ディレクトリを作り、ファイルをコピーする。ここでのディレクトリは任意である。
3. Service Pack 2 が適用された Windows XP や、各パソコンにソフトウェアファイアウォールが導入されている場合は、smpd.exe が命令を待ち受ける TCP ポート (デフォルトは TCP8675 番) を開くようファイアウォールを設定する。簡易に PCEC を構築するには、PCEC を構成するパソコンでファイアウォールを無効にする。
4. PCEC を構成するパソコンでコマンドプロンプトを起動し、カレントディレクトリを C:\%MPICH2 に移動して、smpd -install と入力する。この操作で、smpd.exe がサービスとしてインストールされ、起動する。以後は、Windows 起動時に自動的にサービスとして起動される。
5. 2. から 4. を PCEC を構成するマシンすべてについて繰り返す。

6. PCEC を構成するすべてのパソコンのホスト名を `hosts.txt` というファイルに 1 行 1 台で書き込み、`C:\MPICH2` に保存する。ホスト名は Windows が認識できるものであれば、DNS 名でも NetBIOS 名でもよい。PCEC を用いた並列計算実行時には `hosts.txt` に記述したパソコンは起動していなければならない。

2.2 PCEC におけるプログラムの実行

ファイルコピーにより 2.1 に示す手順で実行環境だけを構築した場合は、計算を指示するパソコンの `C:\MPICH2` にプログラムをコピーし、コマンドプロンプトを起動して `C:\MPICH2` をカレントディレクトリとした後、

```
mpiexec -machinefile hosts.txt -n N program
```

としてプログラムを実行できる。ここで、`N` は実行に用いるパソコンの数、`program` はプログラムの実行ファイル名である。実行すると `account(domain%user):` というプロンプトが現れるので、PCEC を構成するすべてのパソコンで有効なユーザー名を入力する。次に、`password:` というプロンプトが現れるので、ユーザー名に対するパスワードを入力する。ユーザー名とパスワードを入力するのはコマンドプロンプト内で最初に実行するときだけでよく、次からは `mpiexec` を実行すればすぐにプログラムが実行される。

プログラムを `C:\MPICH2` にコピーしない場合でも、絶対パスでプログラム名を指定したり、プログラムの存在するディレクトリをカレントディレクトリとして、`C:\MPICH2\mpiexec` とするなどして `mpiexec.exe` を絶対パスで指定すれば実行できる。

MPICH2 をインストールし開発環境・実行環境とも導入している場合でも、上述の方法でプログラムを実行できる。また、この場合は GUI ツールもインストールされるので、スタートメニューのプログラム MPICH2 に含まれる `wmpiconfig` を使って稼働ホストを検索し、`wmpiexec` を使ってもプログラムを実行できる。

コマンドプロンプトを使った操作方法は、初学者にはとりつきにくい Linux PC クラスタにおける実行方法と互換性があるため、PCEC ではこちらの方法を主として用いる。並列プログラミングの教育を Linux 上で行う場合、オペレーティングシステムとしての Linux の使い方を教える必要も生じるが、Windows を用いる PCEC ではそういったことが不要になるという利点もある。また、セキュリティ対策のためシステムディレクトリへの書き込みを制限している場合は、ファイルコピーによる方法しか用いることができない。このような場合は必然的にコマンドプロンプトを使う操作方法を用いることになる。

2.3 PCEC 用並列プログラムの開発

MPICH2 環境で実行するプログラムの開発には、Microsoft Visual Studio .NET 2003 と `cygwin`[7] を用いることができる。後者は無償で入手でき、多くの大学のパソコン教室に導入されているほか、Linux PC クラスタで用いられている開発環境とのソースレベルの互換性が高い。このため、PCEC では `cygwin` を用いることにした。`cygwin` はセットアッププログラムによりインストールできるが標準ではコンパイラ等がインストールされないため、`gcc`、`make` などプログラム開発に必要なツールを明示的に指定してインストールする必要がある。

MPICH2 を PCEC を構成するそれぞれのパソコンにインストールし開発環境・実行環境とも導入している場合は、インクルードファイルとライブラリファイルのディレクトリをコンパイル時に適切に指定すればよいことになる。具体的には、`C:` ドライブに MPICH2 をインストールしたとして、

```
-L/cygdrive/c/program%files/mpich2/lib
```

と、

```
-I/cygdrive/c/program%files/mpich2/include
```

をコンパイル時に指定する。MPICH2 のライブラリをリンクするため、

```
-lmpich2
```

も必要になる。

ファイルコピーにより構築した PCEC の実行環境に対しては、MPICH2 をインストールした任意のパソコンから、インクルードファイルとライブラリファイルをそれぞれ、`C:\MPICH2\include`、`C:\MPICH2\lib` にすべてコピーすることで MPICH2 の開発環境を追加することができる。この場合は、
`-L/cygdrive/c/MPICH2/lib`
と、
`-I/cygdrive/c/MPICH2/include`
および、
`-lmpich2`
をコンパイル時に指定する。

この方法でコンパイルしたプログラムを PCEC で実行する際には、`C:\cygwin\bin` にインストールされる `cygwin1.dll` をプログラムの実行ファイルと同じディレクトリに置く必要がある。ファイルコピーにより PCEC を構築する場合は PCEC を構成するすべてのパソコンの `C:\MPICH2` にあらかじめ `cygwin1.dll` をコピーしておくのが簡単である。

なお、コンパイル時のオプションに `-mno-cygwin` を指定すれば、`cygwin1.dll` を必要とせず単独で動作できる Windows アプリケーションを作成できるが、Linux 環境とのソースレベルの互換性が低くなるため、Linux PC クラスタ用に作成された MPI 対応並列プログラムをコンパイルできない場合が多くなる。

3 PCEC の検証

鹿児島大学学術情報基盤センター端末室内にファイルコピーにより構築した PCEC を用い、簡単なプログラムを動作させて PC クラスタとしての性能を検証した。結果を以下に示す。

3.1 検証に用いたプログラム

検証に用いたのは、MPICH2 に標準で付属しているサンプルプログラム `cpi.exe` である。`%Program Files%\MPICH2\examples` ディレクトリにソースファイル `icpi.c` とともにインストールされる `.MPICH2`

の付属文書によれば、`cpi.exe` は Microsoft Visual Studio .NET 2003 でコンパイルされており、実行に必要なのはこのファイルのみである。

サンプルプログラム `cpi.exe` は、

$$\pi = \int_0^1 \frac{4}{1+x^2} dx$$

の関係を用いて定積分により円周率を計算するプログラムである。積分区間 $[0, 1]$ を細かく分割することにより、精度を高めることができるが、それだけ計算に時間が多くかかるようになる。並列計算環境を構成するそれぞれのパソコンに区間を分割して計算させることで、計算時間の短縮が実現できる。`cpi.exe` では実行時に分割数を指定して計算を行う。計算を実行するパソコンの台数は、`mpiexec.exe` を実行する際のコマンドライン引数で指定する。

3.2 検証結果

鹿児島大学学術情報基盤センター端末室内の 16 台のパソコンを用いて構築した PCEC 実行環境において、`mpiexec.exe` に `-n` オプションで指定する実行マシン数を 1~16 まで変えて `cpi.exe` を実行し、計算に要した時間を計測した結果、及びそれに基づいて計算した高速化率と並列化効率を表 3 に示す。

`cpi.exe` に与える分割数としては、 10^{11} を用いた。16 台で計算した場合に、円周率の近似値として 3.1415926535896839 を得た。同じ桁数の正解 3.1415926535897932 と比べ、 1.093×10^{-13} の誤差となる。

高速化率 $S(p)$ は、台数を p 、時間を $T(p)$ として、以下の式で定義される。

$$S(p) = \frac{T(1)}{T(p)}$$

理想的な並列計算機であれば $S(p) = p$ となるが、実際にはそうはならない。理想との差を並列化効率 $E(p)$ といい以下の式で定義される。

$$E(p) = \frac{S(p)}{p}$$

表 3: 16 台までの PCEC 環境で `cp.exe` を実行した時の計算時間，高速化率，並列化効率．

台数	時間 (秒)	高速化率	並列化効率
1	18.07	1.00	1.000
2	9.06	1.99	0.997
3	6.02	3.00	1.001
4	4.52	4.00	1.000
5	3.63	4.98	0.995
6	3.02	5.98	0.997
7	2.59	6.98	0.996
8	2.27	7.95	0.993
9	2.02	8.94	0.993
10	1.82	9.92	0.992
11	1.66	10.88	0.989
12	1.53	11.84	0.986
13	1.43	12.66	0.974
14	1.31	13.76	0.983
15	1.25	14.47	0.964
16	1.18	15.29	0.956

`cp.exe` は，並列化効率が高くなる性質を持つアルゴリズムを用いたプログラムではあるが，表 3 に示すように，PCEC において高い並列化効率を達成できた．16 台までであれば，理想的な並列計算機に近い性能を達成している．

4 まとめ

一般的な学校のパソコン教室に導入可能な並列計算環境 — PCEC — の構築方法を示すとともに，実環境への導入と実際の並列計算を行って実用性を検証した．

鹿児島大学学術情報基盤センター端末室では，Active Directory によりユーザー認証が行われており，ユーザーには各パソコンの管理者権限はない．しかしファイルコピーによる PCEC の構築方法は Windows のシステムディレクトリへの書き込みを伴わない．このため，ユーザーとしてログインしてファイルをコピーするだけで PCEC を構築でき，問題なく並列計算機として動作した．あらかじめ

ファイルサーバにコピーするファイルを用意したため，1 台あたりの作業時間は 3 分とかかっていない．このことから，パソコン教室に事前に PCEC が導入されていない場合でも，生徒に作業させることで数分で環境を用意することができるものと思う．パソコン教室では再起動時にハードディスクの内容があらかじめ設定されたものに初期化される仕組みが導入されていることが多いが，そうした環境でも PCEC は容易に導入できるはずである．

PCEC は並列プログラミングの演習環境を提供することを目的としており，並列計算機として科学技術計算を行うことは考慮していない．しかし，並列計算を実際に行った結果，単純なプログラムではあるが高い並列化効率を達成できた．このことから，PCEC は PC クラスタとして活用できる可能性もあるものと考えている．計算の度にファイルコピーにより PCEC を導入するのは現実的ではないが，事前に PCEC を導入したパソコン教室であれば，遊休時間に HPC 環境として用いることも難しくはない．ただし，ジョブの投入，停止，待避など，HPC 環境に必要なツールが未整備であるので，長時間の計算を行う場合にはこれらを開発する必要がある．

参考文献

- [1] <http://www.mpi-forum.org/>
- [2] http://www.csm.ornl.gov/pvm/pvm_home.html
- [3] 庄司文由, 隅谷孝洋, 石井光雄, “教育端末の遊休時間を利用した HPC 環境,” 学術情報処理研究, No. 9, pp. 27–36, 2005.
- [4] <http://w3.cc.kagoshima-u.ac.jp/>
- [5] <http://www-unix.mcs.anl.gov/mpi/mpich2/>
- [6] <http://www.lam-mpi.org/>
- [7] <http://www.cygwin.com/>