

テネシー大学における High Performance Computing

長谷川 秀彦

図書館情報大学

電子メール : hasegawa@ulis.ac.jp

テネシー大学の Dongarra 研究室で行われている High Performance Computing を紹介する。ここでは、(1) ソフトウェアや情報を流通させるために email, anonymous ftp, WWW などを効果的に使ったツールの作成と情報サービスの運営、(2) 多種多様なコンピュータを一様なインターフェースで使うための作業環境の開発、(3) 高性能な線形計算ソフトウェアの開発が行われている。

最後に、日本における HPC 分野の情報流通を改善するため、WWW を利用したボランティアベースの情報交換サービス PHASE (Parallel and High-performance Application Software Exchange) の構想について述べる。

High Performance Computing at the University of Tennessee

Hidehiko Hasegawa

University of Library and Information Science

Email : hasegawa@ulis.ac.jp

The author introduces what kinds of research for High Performance Computing are carried out at the Dongarra laboratory, University of Tennessee. There are three main fields of research:

- (1) development of information exchange services and managing it,
- (2) development of software environments having portability,
- (3) development of Linear Algebra Softwares.

At last, the author shows a plan of PHASE (Parallel and High-performance Application Software Exchange) served for accelerating the speed and enhancing the quality of information exchange in Japanese HPC community. This service will be managed by volunteers. The usability of it would depend on the cooperation of its users.

1. はじめに

文部省在外研究員として約1年間にわたってテネシー大学 UTK (University of Tennessee, Knoxville) の Dongarra 研究室 (ドンギャラと発音) に滞在して見聞きしたアメリカの High Performance Computing について述べる。Dongarra 研究室で行なわれていることをひとりでいえば High Performance Computing だが、大別すると、(1) ソフトウェアや情報を流通させるためのツールの作成・運営、(2) 最新のコンピュータを使うための共通の作業環境の開発、(3) 高性能な線形計算ソフトウェアの作成ということになるだろう。個別の研究なら日本でも行なわれているが、環境・作成・流通までを含んだこのように広範なプロジェクトは日本では行なわれていないだろう。

Jack J. Dongarra 氏は UTK の Distinguished Professor かつ ORNL (Oak Ridge National Laboratory) の Distinguished Scientist であり、研究室には大学院生、私のような訪問者など、常時 30 人くらいが属している。Distinguished Professor / Scientist というのは UTK 独自の制度で、科学技術の研究・教育を活発にするために外部の研究所の研究者に主として研究指導をしてもらうものである。原則として月学金は大学、火金は研究所というのが Dongarra 氏のスケジュールだが、実際は世界中を飛び回っている。

UTK は 1994 年 9 月に 200 周年を迎えた古い大学で、メインキャンパスは東部テネシーの人口およそ 15 万ほどの町ノックスビル (Knoxville) にあり、約 30 Km ほど北北西の Oak Ridge にある ORNL までは車で 40 分程である。Knoxville は Atlanta から飛行機で 45 分、Chicago から飛行機で 1 時間半のところであり、TVA (Tennessee Valley Authority) の本社がある町でもある。計算機科学科は教官数 25 人ほどで、電話帳の表紙や絵葉書になる丘のうえの歴史的な建物 Ayres Hall がある。建物は古いが、コンピュータに必要なネットワークなどはみてくれを損なわないように配置されている。

2. 計算環境

UTK はごく普通の州立大学なので、計算機科学科とはいってもそれほど特別な設備はない。学科で所有している計算機は

- Thinking Machine's CM-5 32 Processors
- MasPar MP-2 16,000 processors
- Intel iPSC i860 128 processors
- Sequent Symmetry 16 processors

と Sun, SGI, IBM, HP, DEC など約 100 台のワークステーションである。ユーザからみた標準的な計算環境は、インターネットに接続された大きめのディスクスペースをもつワークステーションで、これから世界中のコンピュータを使う。ORNL からは研究者がやってきているためか、ORNL のコンピュータを使うことは多い。このほか、学内のコンピュータセンターにはメインフレームがある。実際、目につくハードウェアはワークステーションとポストスクリプト・プリンターばかりで、自分が使っているハードウェアがどこにあるかは意識しないでも使えるようになっている。

このような環境におけるソフトウェア開発のため、個々の環境の差を吸収するソフトウェアを用意している。この方法はオーバーヘッドとなってコンピュータの性能を落とすが、機種による差を意識せずに汎用性をもったソフトウェアが開発できるという点で、人間を中心に考えると能率的なソフトウェア開発となる。

3. ソフトウェアや情報の配布

最近、学会などで会った研究者に何かを尋ねると

- send email
- get through anonymous ftp
- use WWW

という答えが返ってくる。これからもわかるように、email, anonymous ftp, WWW (ビューワとして Mosaic) という道具は広く普及している。

このような道具を使って数学ソフトウェアを配布する仕組みが netlib である [10]. netlib は 1985 年に電子メールを用いて数学ソフトウェアを自動配布する numerical Software Library として始められ、現在では多くの著名なソフトウェアが蓄積されており、年中無休、しかも無料でサービスが受けられる。ユーザはいつでも必要になったとき、ここにアクセスしさえすればよい。最初は電子メールで要求されたソフトウェアを自動応答システムを経由して電子メールで配送していたが、現在では

- email
- anonymous ftp
- Xnetlib
- Gopher
- WWW (World Wide Web)

の 5 通りの方法が利用できる。Xnetlib は Xwindow システムを用いた netlib 独自のユーザインターフェイスである [11] [12]. この他、Bell Labs, Norway, England, Germany, Taiwan, Australia などにもサイトがあるが、テネシー・サイトだけで 1995 年 1 月 1 日から 5 月 14 日までに 160 万件、開設以来約 500 万件のアクセスがある。

ここに収められているのは

- ソフトウェア (ソースプログラム)
- マニュアル
- 論文、テクニカルペーパー
- 本 (ポストスクリプト・ファイルを自分でプリントする)
- 本 (Mosaic などを用いて読む)
- シンポジウムなどのプログラム
- 発表のスライド
- ベンチマーク・データベース
- NA-net

などで、netlib についての情報、アクセス件数とともに <http://www.netlib.org/> から入手できる。UTK や ORNL の紹介もここから入手できる。

netlib の使い方は「send index」という 1 行だけの電子メールを netlib@ornl.gov に送れば入手できる。すなわち

```
% echo "send index" | mail netlib@ornl.gov
とすればよい。
```

NA-net と呼ばれるサービスでは、電子メールの転送、数値解析関係者の Directory、電子メールを利用した週 1 回のニュース NA Digest の配布などが行なわれている。Directory を検索すれば関連分野の人をリストアップできるし、ニュースを利用すれば知りたいことに対する答えが得られることも多い。ニュースは、各メンバーから送られた電子メールを編集者 (主として Cleve Moler, The MathWorks, Inc.) がまとめて全員に送るだけの簡単なものだが、雑誌の目次、研究会の案内など、たくさんの有用な情報が含まれている。WWW を用いて過去のニュースの検索もできる。

NA-net の使い方は na.help@na-net.ornl.gov に電子メールを送れば入手できる。すなわち

```
% echo "hello" | mail na.help@na-net.ornl.gov
とすればよい。
```

ベンチマーク・データベースではすべての機種の最新情報が見られる。ベンダーが公開しているコンピュータの仕様と合わせて見ることで、コンピュータ上で大抵のことが済んでしまうため、研究者は苦勞してカタログやデータを最新となるように維持する必要がなくなる。

このような情報提供のメリットは

- とにかく便利である
- 必要な時に最新の情報がすぐ入手できる
- メンテナンスがラクである

ということであろう。情報を提供する側は配布先に気を巡らすことなく、手元のファイルを最新に保てばよい。利用者は必要になったときに必要な情報を取り寄せられる。このようなシステムは多くの情報が提供されないと普及は難しいが、現在では完全に普及してしまっている。

一方、WWW を用いた情報提供が簡単にできるようになったため、それこそ無数の情報提供先のアドレスを覚えなければならなくなっている。そこで、NHSE : National HPCC Software Exchange として、<http://www.netlib.org/nse> で Yellow Page のようなサービスが行われている [5].

NHSE では

- High Performance Computing and Communication に関する活動
- Grand Challenges
- ソフトウェア
- データベース
- ベンダー
- テクニカルレポート, 学会誌, 学会案内
- ネットワーク上の情報資源
- HPCC についてのロードマップ
- 国立研究所の研究紹介
- 関連した URL のリスト

などの WWW 情報へのポインタを提供している。ユーザが自分の情報を付け加えることもできるし、有名教授の授業ノートを見ることだってできる。NHSE を利用するには WWW のブラウザが使えることが条件となるが、それさえクリアすれば上記のアドレスだけで大抵のことは済む。

これらの情報の活用方法として、例えば IBM の SP2 用に後述の BLAS をチューニングすることを考えよう。BLAS の原形は netlib から ftp で取ってくればよい。SP2 がどういうコンピュータかは、NHSE を利用して IBM にアクセスすればよいし、ベンダーが提供しているベンチマーク・データは netlib のベンチマーク・データベースから検索できるし、コンパイラ・オプションと性能の関連についてのテクニカルレポートも netlib から入手できる。疑問点は NA-net に電子メールを送れば、誰かが教えてくれるかもしれない。テクニカルレポートを書いたのなら、NHSE を経由して配布することもできる。

はたして国産のコンピュータ、例えば HITAC S3800 でこのような公開情報だけでチューニングができるだろうか？ ネットワーク経由で自作のソフトウェアを提供したり、WWW 上のいろいろな情報を効率的に入手できるだろうか？ 日本のコンピュータ・メーカーの技術者に email で連絡がとれるだろうか？ これらは、高校で使えるだろうか？ 日本では、すべてはこれからである。

4. 共通の作業環境の開発

世間にはいろいろなコンピュータがあふれかえっていて、OS が UNIX といっても少しずつ違う。並列コンピュータ A と並列コンピュータ B のために別々のソフトウェアを開発しなければならないだろうか？ そうだとすると大変な労力がかかる。

このような労力を避けるため、汎用性を強く意識して開発されているのが、PVM, MPI, BLAS, BLACS である。

PVM (Parallel Virtual Machine) はネットワークで結合された多種多様なコンピュータを 1 台の巨大なコンピュータとして活用するためのメッセージパッシングライブラリで、FORTRAN77 と C から呼びだせる [13][14][15][21]。PVM さえ動いていれば、1 台のワークステーションから並列コンピュータまで同一のプログラムがリコンパイルだけで実行できるし、異機種を 1 台のコンピュータに構成できる。並列コンピュータがなくても並列計算ができる。このような分散メモリ形式の MIMD プログラムを書くための支援ツールあるいは支援環境として HeNCE (Heterogeneous Network Computing Environment) [3][4], VPE (A Visual Parallel Programming Environment) [19], Xwindow 上で PVM を実行するための環境 XPVM などが開発されている [17]。

PVM が de facto 標準のひとつとして広く使われるようになると、特別なハードウェアを持つ並列コンピュータ用に高速化が図られるようになる。日本製にワークステーションのように PVM 作成時に想定されていない機種の場合は、動かすためにユーザもしくはベンダーによる修正が必要になる。多くのコンピュータを Heterogeneous な環境で動かすこと、バージョンアップに追随する

ことを考えると、なんらかの規約が必要になってくるだろう。

いろいろな de facto 標準を包含するのが MPI (Message Passing Interface) である [21]. MPI は HPF (High Performance Fortran) のように、関心のある企業・グループの議論から設計がなされ、真に標準的なメッセージパッシングライブラリとして各社で作成されている。

BLAS (Basic Linear Algebra Subprograms) は線形数値計算に現れる演算がサブルーチンの形で定義されたもので [18][8][9], 高速化を達成するための道具立てとして使われている。現在では、ほとんどの行列計算ソフトウェアが BLAS を使っているため、BLAS を高速化しさえすれば一応の高速化が達成される。高水準言語で書かれた BLAS が高性能を発揮しない場合でも、ハードウェアの特性を活かすようにアセンブラで BLAS を作れば高性能を発揮できる。多くベンダーはハードウェアに対応した BLAS を提供している。

逆に BLAS がコンピュータの差異を吸収するため、ベンダーは行列計算ライブラリ全体を作らなくとも、高速化した BLAS と後述の LAPACK を組み合わせて容易に高精度かつ高性能なソフトウェアを提供できる。BLAS は高水準言語版が公開されているため、いかなるコンピュータでも利用できるが、真に高速化を達成し、しかもその成果をユーザ全体で共有するためにはベンダーの協力が不可欠である。

BLACS (Basic Linear Algebra Communication Subprograms) は線形代数用のメッセージパッシングライブラリで [7], 2次元の格子状に配列されたプロセッサに行列データを格子位置の指定だけで送ることができる。BLACS の目指しているところも BLAS と同様で、並列コンピュータの詳細にこだわらずにプログラムが書けて、できあがったプログラムは分かりやすかつどこでも実行できるようにということである。並列行列計算ライブラリがまだ作られていない現状では、行列計算用の標準的なメッセージパッシングライブラリとなるべく Portability を強く意識している。現時点ではベンダー提供の BLACS がないため、一般的な並列コンピュータ用 BLACS と PVM 用 BLACS が netlib を通して公開されている。

5. 線形計算ソフトウェアの作成

このような道具立てで作られている線形計算ソフトウェアが LAPACK, Templates, ScaLAPACK である。実際の作業の多くは UTK で行われているものの、世界中の研究者の研究成果が活用されていること、UCB (University of California, Berkeley) などとネットワークを活用した共同作業が行われていることが特徴である。

LAPACK (Linear Algebra Package, エルエーパックと呼ぶ) は昨年秋に第2版が発行された高精度な線形計算ライブラリで [1], 連立一次方程式の解法, 固有値解析, QR 分解などの FORTRAN77 プログラムからなる。EISPACK, LINPACK の機能を包含し、計算の核に BLAS を使っている。キャッシュ・アーキテクチャのコンピュータで高速となるようブロック構造が導入されている。単・倍精度実数, 単・倍精度複素数の4種類についてテストプログラム, タイミングプログラムなどを合わせるとソースプログラムだけで約 27 MB になる。ディスクスペースの制限がきつい日本的な計算環境では、連立一次方程式ひとつを解くのにも多くのサブルーチンが必要なので不便に感じるかもしれないが、条件数の推定, 誤差の上限予測などの充実した機能をもつ。C++ 言語によるフロント・エンド LAPACK++, C 言語版なども提供されている。

Templates は反復法プログラムの集まりである [2]. 反復法は収束についての知識が必要で、しかも多くの場合、複雑なデータ構造の大規模疎行列をもつ連立一次方程式を対象としているため、ライブラリ化が困難と思われていた。ここでは反復法アルゴリズムとデータ構造を分離して「テンプレート」にまとめている。実際の仕組みは、「行列ベクトルの積」と「前処理をベクトルに施すこと」の2つの行列演算サブプログラムの作成をユーザに任せている。ユーザは自分の問題固有のデータ構造, 使用環境に依存したチューニングなどがアルゴリズムとは独立に行なえる。そのかわり、メインプログラム (メモリの確保) とデータの渡し方 (COMMON 文あるいはファイル) に影響があるので、ライブラリのようにサブルーチンを交換するだけでは済まなくなる。

SIAM (Society for Industrial and Applied Mathematics) の新しい試みとして、この本は出版と同時に印刷原稿のポストスクリプト・ファイルが netlib から提供され、最近では WWW のブラウザを用いてオンラインで読むこともできる。LAPACK の初版から始められた本の収入を SIAM のシンポジウムに参加する学生の旅費援助に使うことのほかに、有料の本と同じ内容を同時に無料で提供することが特徴になっている。学会活動と出版のこのような関係は興味深い。

ScaLAPACK (Scalable LAPACK) は今年 3 月に第 1 版が正式公開された MIMD 型の分散メモリコンピュータ用の行列計算ソフトウェアである [6]。密行列用の 7 本の LAPACK プログラムを MIMD 型の分散メモリコンピュータ用に拡張したもので、BLAS のかわりに PB-BLAS (Parallel Blocked Basic Linear Algebra Subprograms) と PBLAS (Parallel Basic Linear Algebra Subprograms), 通信には BLACS が使われている。現時点では、実用バージョンというより、並列線形計算ライブラリのひとつの実現方法というべきもので、多くの研究者が自分のプログラムとの比較に使っている。

6. 利用者教育

UTK と ORNL, テネシー州のいくつかの大学が協力しあって High Performance Computing に関する研究、訓練、相談、技術移転を行なう学科とは独立した機関 JICS (Joint Institute for Computational Science) が UTK に設けられている [16]。ここでは、学生・研究者のために、休暇中に CM-5, MP-2, iPSC i860, Kendall Square Research KSR-1 64 processors, Intel Paragon, IBM SP2, PVM などのセミナー（原則として午前は講義、午後は実習）を行ない、いろいろな分野の研究に High Performance Computing が役立てられるような支援を行なっている。セミナーの際にももらったアカウントが有効な数週の間は、自分の大学から Internet 経由で利用できるし、いろいろな機能を試すこともできる。

問題をかかえている研究者なら、比較的簡単な手続きでこのような設備が使えるため、コネに

頼って使わせてもらったり、買ってから始めて使うといったことは少ない。また、自分の大学にそのような設備がなくても、このような機会を活用すればだれでも High Performance Computing にふれられるようになっている。これらの点は日本の大学・研究所とは大いに異なる素晴らしい点である。

7. 将来の展望

共通の開発環境や高性能な線形計算ソフトウェアについての評価をするためには詳細な比較・検討が必要なので、ここではこれ以上言及しない。彼らの開発方針は「どこでもだれでも使える」ことなので、興味をもたれた場合はぜひ使ってみてほしい。

ソフトウェアや情報の配布するためのツールの作成・運営に関しては、日本の HPC 分野には netlib のような (1) ソフトウェアを配布するシステムもなく、(2) 各種情報への効率的なアクセス手段もなく、しかも (3) 情報の流通をとりまとめる機関もないのが現状である。問題がこれだけだとは思っていないが、最近手軽に利用できるようになった道具を用いれば、比較的容易に情報流通の促進を図ることができる。

あるところでこのような話をしたところ、その情報がかけ巡って数人の協力者とともに次のようなボランティアベースのサービスを提供できる目処がきつつある。

- 名称: PHASE (Parallel and High-performance Application Software Exchange)
- 対象: High Performance Computing
- 目的
ソフトウェアと情報の共有を図る
情報交換を円滑かつ容易にする
情報の流通に必要なことを明らかにする
- サービス内容
ftp を使ったソフトウェアの配布
WWW を使った情報の提供

● 方針

Free: タダで誰にでも

あるがままの状態 で提供する

営利目的の利用は不可

ユーザに何かを要求したり、料金をとったりしない

作成者名と PHASE から入手したことを明示する

問題点は起きてから検討する

責任はとらない

● ボランティア

東京大学情報科学科 小柳義夫

御茶の水女子大学情報科学科 長嶋雲兵

東京大学大型計算機センター 村尾裕一

電子技術総合研究所 佐藤三久

電子技術総合研究所 関口智嗣

電子技術総合研究所 中田秀基

図書館情報大学 長谷川秀彦

具体的には、(1) 本やプログラム・コンテンツなどで公表されたソフトウェア、ベンチマーク・プログラムなどを ftp で取り寄せられるようにすること、(2) 学会、SIG、国内外の機関の情報へアクセスできるようにすること、(3) ベンダーの提供している情報へアクセスできるようにすること、(4) 各種 mailing list (HPC, PVM 等) への加入方法を列挙すること、(5) ベンチマーク結果を蓄積・公開することである [20]。ユーザは、例えば <http://WWW.etl.go.jp/PHASE/> のようなひとつのアドレスだけを知っていれば、ソフトウェアを取り寄せたり、関連するすべての情報にアクセスできるようになる。(2) から (5) は WWW の使用を前提とする。

このようなことが可能になった背景には、ftp 機能をもつコンピュータなら相手を問わずファイルの転送ができること、WWW は Internet 経由でも電話線経由でも同一形式の情報にアクセスできて、しかも情報を提供する側はインターフェースを準備する必要がないことなどがある。現時点

ではコスト、利用可能性の面での制約もあるが、このようなサービスを開始することでユーザと提供者の両面から情報流通が促進できると考えている。

これは、便利だということが分かれば多くのユーザが付くだろうし、ユーザが多くなればそれ相応の情報提供をせざるをえなくなるといったことを意味している。

とはいえ、専属のスタッフによって恒常的なサービスが行なわれている netlib に比べると、質・量とも劣るのは明らかである。しかし、世界的にみればわずかだが国内では圧倒的なユーザ数をもつパソコン用のソフトウェアとか、日本国内でしか使われていないスーパーコンピュータの情報のように、このような情報サービスだけが対象にできる情報も存在する。しかし、いくら技術が身近になったといっても、このような活動に対してボランティアとして費やせる資源・労力は限られている。

そこで、まずはできることから始めたい。一度使ったらやめられないくらい便利な道具になれるかどうかは、ユーザと提供者にかかっている。

参考文献

- [1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen, *LAPACK Users' Guide, Second Edition*, SIAM, Philadelphia, Pennsylvania, 1995, ISBN 0-89871-345-5.
- [2] M. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donate, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, Philadelphia, Pennsylvania, 1994, ISBN 0-89871-328-5.
- [3] A. Beguelin, J. Dongarra, A. Geist, R. Manček, and V. Sunderam, *Graphical development tools for network-based concurrent supercomputing*, Proceedings of Supercomputing 91, Albuquerque, 1991, pp. 435-444.
- [4] A. Beguelin, J. Dongarra, A. Geist, R. Manček, K. Moore, P. Newton, and V. Sunderam, *HeNCE: A Users' Guide Version 2.0*, 1994. This report is available electronically, the url is <http://www.netlib.org/hence/hence-2.0-doc.html>.

- [5] S. Browne, J. Dongarra, S. Green, K. Moore, T. Rowan, and R. C. Wade, *National Software Exchange Prototype*, Technical Report CS-94-263, Department of Computer Science, University of Tennessee, Knoxville, Nov. 1994.
- [6] J. Choi, J. Dongarra, S. Ostrouchov, A. Petit, D. Walker, and R. C. Whaley, *The Design and Implementation of the ScaLAPACK LU, QR, and Cholesky Factorization Routines*, Technical Report CS-94-246, LAPACK Working Note #80, Department of Computer Science, University of Tennessee, Knoxville, Aug. 1994.
- [7] J. Dongarra and R. C. Whaley, *A User's Guide to the BLACS v1.0*, Technical Report CS-95-281, LAPACK Working Note #84, Department of Computer Science, University of Tennessee, Knoxville, Mar. 1995.
- [8] J. Dongarra, J. Du Croz, S. Hammarling, and R. Hanson, *Algorithm 656: An extended Set of Basic Linear Algebra Subprograms: Model Implementation and Test Programs*, ACM Transactions on Mathematical Software, 14(1)(1988), pp. 18-32.
- [9] J. Dongarra, J. Du Croz, I. Duff, and S. Hammarling, *A Set of Level 3 Basic Linear Algebra Subprograms*, ACM Transactions on Mathematical Software, 16(1)(1990), pp. 1-17.
- [10] J. Dongarra and E. Grosse, *Distribution of mathematical software via electronic mail*, Communications of the ACM, 30(5)(1987), pp. 403-407.
- [11] J. Dongarra, T. H. Rowan, and R. C. Wade, *Software distribution using Xnetlib*, Technical Report ORNL/TM-12318, Oak Ridge National Laboratory, Oak Ridge, June 1993.
- [12] J. Dongarra, T. H. Rowan, and R. C. Wade, *Software distribution using XNETLIB*, To appear in ACM Transactions on Mathematical Software.
- [13] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, and V. Sunderam, *PVM: Parallel Virtual Machine, A Users' Guide and Tutorial for Networked Parallel Computing*, The MIT Press, Cambridge, Massachusetts, 1994, ISBN 0-262-57108-0. This book is available electronically, the url is <ftp://www.netlib.org/pvm3/book/pvm-book.ps>.
- [14] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, and V. Sunderam, *PVM 3 User's Guide and Reference Manual*, Technical Report ORNL/TM-12187, Oak Ridge National Laboratory, Oak Ridge, May 1994.
- [15] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, and V. Sunderam (村田英明訳), *PVM 3 ユーザーズガイド & リファレンスマニュアル日本語版*, Aug. 1994. This translation is available electronically, the file is </pub/pvm/pvm3-jp/jpvmug.ps.Z> at etlport.etl.go.jp.
- [16] The url is <http://www.jics.cs.utk.edu/>.
- [17] The url is <http://www.netlib.org/utk/icl/xpvm/xpvm.html>.
- [18] C. Lawson, R. Hanson, D. Kincaid, and F. Krogh, *Basic Linear Algebra Subprograms for Fortran Usage*, ACM Transactions on Mathematical Software, 5(3)(1979), pp. 308-329.
- [19] P. Newton and J. Dongarra, *Overview of VPE: Visual Environment for Message-Passing*. Private communication.
- [20] 関口智嗣, 佐藤三久, *HPC の性能評価 - 測定から科学へ*, ハイパフォーマンスコンピューティング研究会 93-HPC-46, 93(33) (1993), pp. 21-28.
- [21] 関口智嗣, 長嶋雲兵, 日向寺祥子, *ワークステーションとメッセージパッシングライブラリ*, ハイパフォーマンスコンピューティング研究会 93-HPC-47, 93(51) (1993), pp. 13-20.