ハイパフォーマンス
コンピューティング 67－17
（１９９７．８．２０）

複数の並列計算機上での科学技術計算のための統合利用環境の構築

武宮 博　今村俊幸　太田浩史　川崎琢治　樋口健二　小出 洋

日本原子力研究所　計算科学技術推進センター

科学技術計算は多くの計算機資源を必要とするため，ネットワークに接続された複数の並列計算機を使用して１つの問題を並列分散的に処理する研究が盛んに行われている．科学技術計算プログラムの並列分散化とその実行には，複数の並列計算機を統合して使用できる計算機間通信基盤を備えた環境が必要とされる．本研究では，科学技術計算プログラムの並列分散化とその実行に必要な機能について議論し，それらの機能を実現する新しい並列分散処理のための統合環境，STA 基本ソフトの第２版を提案し，並列分散処理への適用例を示す．

Design and Implementation for a New Integrated User Environment for Scientific Computing on a Parallel Computer Cluster

Hiroshi Takemiya, Toshiyuki Imamura, Hirofumi Ohta,
Takuji Kawasaki, Kenji Higuchi and Hiroshi Koide

Center for Promotion of Computational Science and Engineering,
Japan Atomic Energy Research Institute

For the propose of efficient processing of scientific computation which requires enormous computational resources, distributed parallel computing becomes a major interest in the recent years. To develop codes and to execute them on the distributed parallel environment require new infrastructure such as common communication layer between parallel computers. The authors propose a new integrated environment, STA environment version 2, which has the common communication layer and includes required services for the distributed parallel computation. The authors also apply the proposed environment to instances of scientific computation.

1　はじめに

ほとんどすべての科学技術計算は，非常に多くの計算機資源 (特に計算時間と記憶領域) を必要とする特徴を持つ．このため，計算時間の短縮化と大きな記憶領域を確保することを目的として，ワークステーションクラスタやスーパーコンピュータなどの並列計算機を使用して科学技術計算を並列処理する研究が行われてきた．最近では，ネットワークに接続された複数の並列計算機を使用して科学技術計算を行う研究が盛んである．本論文では，１台以上の並列計算機を使用して互いに連携しながら計算することを並列分散処理と呼ぶ．

著者らは，科学技術計算プログラムの並列分散化と並列分散処理を行うためには，以下の機能が必要であると考えている．

(1) 計算機間通信基盤

並列分散処理を行うためには，各並列計算機ごとに個別に閉じているプロセッサ間通信が可能であることに加え，複数の並列計算機間で通信が可能である必要がある．このため，それを可能にする計算機間で共通に使用できる通信基盤が必要である．

(2) 並列分散開発環境

各計算機を統合的に管理してプログラム開発できる並列分散開発環境が必要である．

(3) 並列分散利用環境

各計算機を統合的に利用して複数のプログラムを連携させて実行可能にする並列分散利用環境が必要である．

本論文の目的は，これらの必要な機能を満たす科学技術計算プログラムを統合的に並列分散化したり，

並列分散化されたプログラムを利用できる新しい計算機環境(以下，単に並列分散統合環境と呼ぶ)を提案することである．

並列分散統合環境は，日本原子力研究所 計算科学推進センター(以下，原研)が開発中の STA (Seamless Thinking Aid) 基本ソフト第 2 版(以下，STA_2 と呼ぶ)として実現されている [1]．STA 基本ソフトは，並列プログラム開発者の途切れの無い思考の支援を目的とし，並列プログラム開発の各段階を容易に移行できるように，開発ツールを統合的に利用できる GUI(グラフィカル・ユーザ・インターフェース)環境を提供している．その第 2 版である STA_2 は，エディタを中心として利用され，エディタ画面から各プログラム開発ツールを統合的に利用できる．また，既存のプログラムを新しい開発ツールとして組み込むことができ，利用者の目的に合わせて GUI をカスタマイズできる特徴を持つ．

STA_2 は，並列分散統合環境に必要とされる計算機間通信基盤を備えている．その計算機間通信基盤の上に，STA_2 のツールとして並列分散処理に必要な機能を実現している．

著者らは，最初に，科学技術計算を並列分散処理する利点について述べ(2.1節)，並列分散処理のための環境に必要とされる機能について述べる (2.2節)．つぎに，並列分散統合環境の実現について述べ(3.2節)，その適用例を示す (3.3節)．最後にまとめを行う (4章)．

2 並列分散統合環境の目的

2.1 並列分散処理の利点

著者らは，科学技術計算を並列分散処理する主な目的には，つぎに示す 2 項目があり，その目的に対する需要は今後さらに高まると考えている．

利点 1：計算機資源の確保

1 台の並列計算機では計算に必要な計算機資源が確保できないとき，複数の並列計算機を使用して並列分散的に 1 つの問題の計算を実行することになる．

利点 2：計算機資源の利用効率の向上

同じひとつの科学技術計算プログラムにおいても，プログラムの各部分と各計算機の組み合わせで，計算機資源の利用効率が異なる場合がある．この性質を持つプログラムを実行する場合，複数の異なる計算機を用意し，プログラムの各部分と各計算機の組合わせを計算機資源の利用効率が最良になるように選び，1 つの科学技術計算を用意した複数の異なる計算機上で並列分散処理することで，計算機資源の利用効率を向上できる．

2.2 並列分散処理に必要な機能

科学技術計算を並列分散処理することは，計算機資源の確保や利用効率の向上の利点があるが，それを実現する環境には以下に示すさまざまな機能が必要である．

2.2.1 計算機間通信基盤

各並列計算機ごとに個別に閉じているプロセッサ間通信が可能であることに加え，複数の並列計算機間で通信が可能である必要がある．

各計算機ベンダは，C や Fortran などの一般的な逐次型言語を使用して並列計算機用のプログラムを作成するための PVM [2] や MPI [3] などのプロセッサ間通信ライブラリ，あるいは，データパラレル型言語などの拡張型言語 [4, 5, 6] を提供している．しかし，各計算機ベンダが実装したプロセッサ間通信ライブラリや拡張型言語は，性能向上やサポートの理由から並列計算機 1 台だけで使用できるのが普通である．そのため，並列分散処理を行うためには，ソケット通信ライブラリ [7]，Nexus [8]，Ninf [9]，Netsolve [10] などの計算機間通信ライブラリと実際に計算機間で通信を行う仕組みである計算機間通信基盤が必要である．

2.2.2 並列分散開発環境

科学技術計算プログラムの並列分散化を行うには，さまざまなプログラム(開発ツール)を使用する必要がある．

科学技術計算に使用される並列計算機のオペレーティング・システム (OS) は，そのほとんどが UNIX [7] であるが，各並列計算機間で少しづつ異なる．特にファイルシステムや並列プログラムの実行の方法の部分で差異が大きい．

各並列計算機ベンダは，さまざまなベクトル化コンパイラや最適化コンパイラを提供している．各コンパイラでコンパイラオプションやライブラリの指定の方法の差異は大きく，利用者は，たがいに依存関係を持ち複雑なコンパイラオプションやライブラリを指定しなくてはならない．

他に使用する必要のある開発ツールには，エディタ，性能解析ツール，デバッガなどがあるが，どれも差異が大きいといえる．

いままでは，科学技術計算プログラムの並列分散化を行う場合，各並列計算機の環境を交互に利用する方法しかなく，プログラム開発者は各環境の相違に混乱させられていた．

そこで，各計算機を統合的に管理して開発できる並列分散開発環境が必要である．並列分散開発環境は，以下の機能を実現するものとする．

- 各計算機の OS に依存しない共通な GUI 環境を提供する．
- 各計算機のファイルシステムに依存しない共通なファイルシステムを提供する．
- 各計算機で統合的に使用できる開発ツールを提供する．
- 各計算機におけるコンパイラのオプション，必要なライブラリの指定の方法などの差異を吸収する．

2.2.3 並列分散利用環境

いままでは，別のプログラムの結果を入力にする場合など，互いに依存関係がある複数の並列分散化されたプログラムを順次実行するためには，利用者が各並列計算機の運用スケジュールなどの情報を調査し，綿密な計画を立てて行う方法しかなかった．

そこで，各計算機を統合的に利用して複数のプログラムを連携させて実行可能にする並列分散利用環境が必要である．以下の機能が，並列分散利用環境で実現されるものとする．

- 実行するプログラム，使用する並列計算機群，入出力の対象となるファイルを GUI を使用して指定可能とする．
- 使用する並列計算機を容易に選択できるように，以下のようなさまざまな情報を提示する．
  - 並列計算機の運用スケジュール
  - 各並列計算機の負荷，キューの状態
  - プログラムの実行に必要な計算機資源

3 並列分散統合環境の提案

本章で，2.2節で述べた，並列分散処理に必要とされる機能を実現した並列分散統合環境を提案する．

表 1: 実験に使用する並列計算機の構成．

Table 1: The organization of the parallel computers.

設置場所	機種	CPU数	主記憶(GB)	ピーク性能(GFLOPS)
東京中目黒	VPP300	16	9.5	35.2
	SR2201	64	16.0	19.2
	T94	4	1.0	8.0
	SP2	48	3.2	12.8
	SX4	6	1.5	12.0
東海	VPP500	42	10.5	67.2
	AP3000	36	8.0	14.4
	Monte4	4	0.5	6.4
那珂	Paragon	256	8.0	19.2
大阪寝屋川	Paragon	834	104.3	125.1
	VPP300	12	6.0	26.4

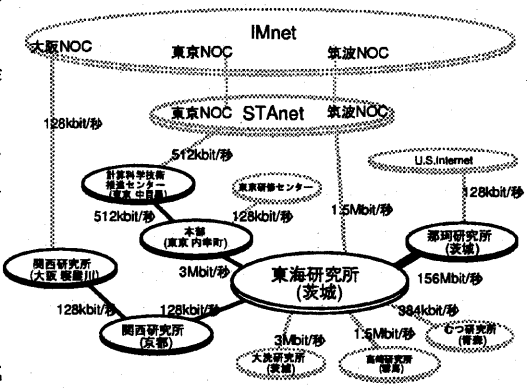

図 1: 実験に使用するネットワークの構成．

Fig. 1: The organization of the network.

3.1 対象とする並列計算機群

著者らが，並列分散統合環境の開発および実験に利用している並列計算機群は計算科学技術推進センターに設置されている 5 台の並列計算機と日本原子力研究所の各事業所に設置されている 6 台の並列計算機から構成されている．さまざまな種類の並列計算機，ローカルエリアネットワーク，大域ネットワークを含んでいる．表 1 に各並列計算機の構成の概略を示し，図 1 に並列計算機間ネットワークの構成の概略を示す．

3.2 並列分散統合環境の実現

本節では，2.2節であげた並列分散処理のための統合環境に必要とされる機能を実現するため，STA_2 に必要な新たなツールについて説明する．

STA_2 は，エディタを中心として利用され，エディタ画面から各開発ツールを統合的に利用できる．また，既存のプログラムを新しいツールとして容易に組み込むことができ，利用者の目的に合わせて GUI をカスタマイズできる特徴を持つ．その基本的構成は，サーバ・クライアント型モデル [7] に基づいており，通信基盤層とツール層の 2 層から構成されている．

通信基盤層は，ツールの実行と終了，ツールとアダプタの通信の中継を行う．ツールは，並列計算機で使用できる既存の開発ツールである．アダプタは，通信基盤層とツールを接続する役割を果たす．

並列分散統合環境を実現するために必要となる基本的なツールは，開発環境提供ツール，作業代行ツール，情報提供ツールである．計算機間通信基盤は，STA_2 に備わっているものを利用すれば良い．以下，各ツールの概要を説明する (図 2)．

3.2.1 開発環境提供ツール

並列分散統合環境とプログラム開発者との界面となる部分である．利用者に各計算機を統合的に管理して科学技術計算プログラムを並列分散化する作業環境を提供する．

Netscape などの既存の WWW ブラウザ，GUI 管理ツール，ファイル管理ツール，エディタなどの各種開発作業用ツールから構成されている．

以下，開発環境提供ツールを構成するツールの概要を説明する．

WWW ブラウザ:

GUI 管理ツールの WWW サーバから各クラスオブジェクトを読み込み，Java アプレット [11] として内部のインタプリタで実行する．

並列分散統合環境の GUI を持つすべてのツールは，それと対になる Java アプレットが存在し，それが WWW ブラウザに読み込まれることにより，各ツールを操作する GUI が実現されていることに注意されたい．

GUI 管理ツール:

既存の WWW サーバ，HTML[12] ファイル，各ツールの GUI を実現する Java アプレットから構成される．WWW サーバは，WWW ブラウザが要求する HTML ファイル，Java アプレットを送信する．

ファイル管理ツール:

利用者に各計算機のファイルシステムに依存しない統合化されたファイル管理のための GUI を提供する．ファイル管理ツールと連携して，ツールが操作する対象となるファイルを操作する機能を提供する．

ファイル管理ツールは，利用者にファイルに関する情報を提供する．利用者からの要求に応じて，ファイルやディレクトリの作成，複写，削除，名称変更の操作を行う．

各種開発作業用ツール:

利用者に各種の開発作業用ツールを操作するための GUI を提供する．

例えば，開発作業用ツールの 1 つであるコンパイラツールは，各並列計算機のコンパイラのコンパイラオプション，ライブラリの指定などの差異を吸収して利用できる GUI を提供する．各並列計算機のコンパイラの起動も行う．

3.2.2 作業代行ツール

並列分散統合環境と科学技術計算プログラム利用者との界面となる部分である．利用者に科学技術計算プログラムを連携して順次処理する作業を支援する．利用者はこのツールを使用して，各ツールを実行する並列計算機を指定し，各ツールで使用するファイルを指定する．

Netscape などの既存の WWW ブラウザ，GUI 管理ツール，ファイル管理ツール，実行計算機指定ツールから構成されている．

以下，作業代行ツールを構成するツールの概要を説明する．開発環境提供ツールの項で説明したツールと同様であるため，実行計算機指定ツール以外のツールは説明を省略する．

実行計算機指定ツール:

利用者が，科学技術計算プログラムを並列分散処理する並列計算機を支援するため，さまざまな情報を提示する．利用者が，各ツールを実行する並列計算機を指定し，各ツールで使用するファイルを指定するための GUI を提供する．

運用情報提供ツール，負荷情報提供ツール，科学技術計算情報提供ツールの各種情報提供ツールから構成されている．

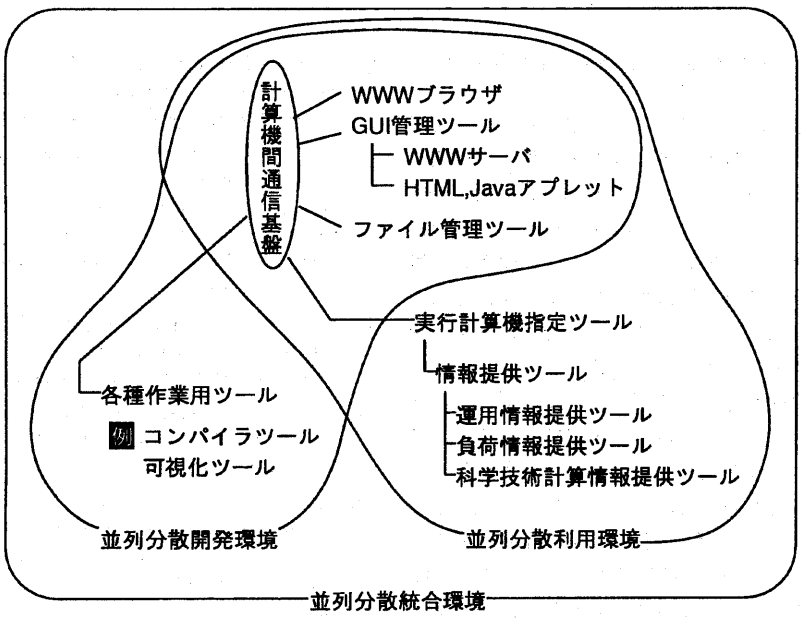

図 2: 提案する並列分散統合環境の構成.
Fig. 2: The organization of the proposed user environment.

3.2.3 各種情報提供ツール

各種情報提供ツールを構成するツールの概要を説明する.

運用情報提供ツール:

休日や保守の日程など運用スケジュールに関する情報, 計算機間ネットワークの構成に関する情報, バッチ運用におけるのキューの構成に関する情報を利用者に提供する.

負荷情報提供ツール:

各並列計算機の負荷情報やバッチ運用におけるキューの長さの情報を利用者に提供する.

科学技術計算情報提供ツール:

各科学技術計算プログラムに関して, 実行に必要な計算時間や記憶領域, 各並列計算機と組み合わせた場合における利用効率の情報を利用者に提供する.

3.3 並列分散処理への適用例

本節では, 並列分散統合環境を使用して, 科学技術計算を並列分散処理する適用例をあげる.

計算機資源の確保を目的とした適用例

中性子, 光子, 電子の輸送問題をシミュレートする粒子輸送モンテカルロコード (代表的なものに, 米国のロスアラモス国立研究所で開発された MCNP コード [13] がある) は, 精度の良い計算結果を得るためには長時間の計算時間を必要とする.

この種のコードの計算時間の大部分を占める粒子ごとの追跡計算は独立に実行できる [14]. MCNP を実行するためのプログラムは, 粒子ごとの追跡計算を行うツール A が, 複数の並列計算機で実行されるように並列分散化され, それ以外の部分 ツール B とに分割されているとする. ツール間の通信は, 並列分散統合環境の計算機間通信基盤 (STA_2 の通信基盤層) を利用して行う.

利用者は, WWW ブラウザを使用して並列分散統合環境の GUI 管理ツールに接続する. WWW ブラウザは並列分散環境の一部となり, 利用者は, MCNP を実行するための GUI を操作できる. 並列分散統合環境用に並列分散化された MCNP が起動されると, 複数の並列計算機でツール A が実行される. ツール B は, ツール A から結果を受け取り集計する.

計算資源の利用効率の向上を目的とした適用例

ITER (国際熱核融合実験炉) [1] などのトカマク装置中のプラズマの巨視的な運動論的振る舞いを観測する計算機シミュレーションが行われている [15, 16]. この種のコード (例えば GYRO3D) には，個々の粒子がつくる電磁場の計算と個々の粒子の軌道の計算が含まれる．前者は行列計算が主であり条件分岐があまり含まれないため，ベクトル型並列計算機を使用するとベクトルユニットを有効に利用できる．後者は粒子の座標などによる場合分けが必要なためベクトル化できず，ベクトル型並列計算機を使用してもベクトルユニットを有効には利用できない．

前者と後者を並列分散統合利用のツール C，D として別々に用意する．利用者は，実行計算機指定ツールで C をベクトル型並列計算機，D をスカラー型並列計算機を実行するように指定すると，システム全体の利用効率を向上できる．

4 まとめ

著者らは，科学技術計算プログラムの並列分散化とその実行に必要な機能を備えた新しい並列分散処理のための統合環境 STA 基本ソフト第 2 版を提案した．また，計算機資源の確保や利用効率の向上を目的とする適用例を示した．

本研究で提案した並列分散統合環境 STA 基本ソフト第 2 版は，すでに設計が完了しており，現在は実装を行っている．

謝辞

早稲田大学教授 笠原博徳氏，本センター次長 相川裕史氏には本研究に関する議論に参加して頂き，ご討論を頂いた．ここに感謝する．

参考文献

[1] 太田浩史, 樋口健二ほか: 途切れの無い思考を支援するプログラミング環境 STA の構築, 計算工学講演会論文集, Vol.2, No.1, pp.97-100 (1997).

[2] Geist, A., Beguelin, A., Dongarra, J. et al.: *PVM: Parallel Virtual Machine, A User's Guide and Tutorial for Networked Parallel Computing,* The MIT Press (1994).

[3] Gropp, W.,Lusk, E. and Skjellum, A.: *Using MPI: Portable Parallel Programming with the Message-Passing Interface,* The MIT Press (1994).

[4] High Performance Fortran Forum: *High Performance Fortran Language Specification,* version 1.1 (1994). `http://www.crpc.rice.edu/HPFF`

[5] Fox, G.,Hiranandani, S.,Kennedy, K. et al.: *Fortran D Language Specification,* COMP-TR901419 (Rice) and SCCS-42c (Syracuse), Department of Computer Science, Rice University, and Syracuse Center for Reseach on Parallel Computation, Syracuse University (1991).

[6] Zima, H., Brezany, P., Chapman, B. et al.: *Vienna Fortran, a language specification,* ACPC/TR 92-4, Austrian Center of Parallel Computation (1992).

[7] Leffler, S., McKusick, M., Karels, M. el al : *The Design and Implementation of the 4.3BSD UNIX Operating System,* Addison-Wesley (1989).

[8] Foster, I., Kesselman, C. and Tuecke, S.: *The Nexus Approach to Integrating Multithreading and Communication,* Journal of Parallel and Distributed Computing, Vol. 37, No. 1, pp. 70-82 (1996).

[9] 関口智嗣, 中田秀基ほか: ネットワーク数値情報ライブラリ *Ninf* – システム実装と評価, 情報処理学会研究報告, HPC-62-22, pp.153-158 (1996).

[10] Casanova, H. and Dongarra, J.: *NetSolve: A Network Server for Solving Computational Science Problems,* UT-CS-95-313, Department of Computer Science, University of Tennessee (1995). `ftp://netlib.org/tennessee/ut-cs-95-313.ps`

[11] Gosling, J. and McGilton, H.: *The Java Language Environment, A White Paper,* Sun Microsystems (1996). `http://java.Sun.COM/docs/white`

[12] Berners-Lee, T.: *Hypertext Markup Language - 2.0,* MIT/W3C, RFC 1866 (1995).

[13] Briesmeister, J. Ed.: *MCNP–A General Monte Carlo N-Particle Transport Code Version 4B,* LA-12625-M, Version 4B, Los Alamos National Laboratory (1997). `http://www-rsicc.ornl.gov/DOCUMENTS.html`

[14] 樋口健二, 川崎琢治: 粒子輸送モンテカルロ・コード MCNP の並列処理, JAERI-Data/Code, 96-019, 日本原子力研究所 (1996).

[15] 内藤裕志, 徳田伸二ほか: 小特集:超並列計算機とプラズマ, プラズマ・核融合学会誌, Vol.72, No.8 (1996).

[16] Imamura, T.,Tokuda, S.,Naitou, H.: *Parallelization of 3D Gyrokinetic Particle Code: GYRO3D Using MPI and Its Performance Evaluation on Parallel Computers,* Proc. M&C+SNA'97 (1997).

[1] `http://www.jaeri.go.jp/~intro/ITER`