

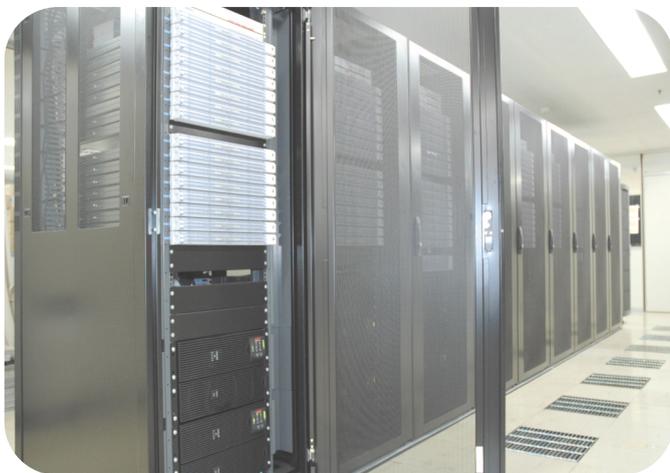


大阪大学サイバーメディアセンター

計算機利用ニュース

Vol. 3 No.1 2007.5
第4号

Cybermedia Center, Osaka University



利 用 案 内

主なサービス内容 (設置機種)	係・室名等	連絡先・電話番号 (大阪大学内からは下4桁が内線番号となります)	月～金	土・日・祝日
--------------------	-------	-------------------------------------	-----	--------

●開館サービス時間

センター利用者の呼出 センター見学の申込等一般受付	情報推進部情報企画課 総務係 soumu@cmc. (吹田本館 1F)	06-6879-8804	9:00～12:00 13:00～17:15	閉
会計事務一般	情報推進部情報企画課 会計係 kaikai@cmc. (吹田本館 1F)	06-6879-8810		
利用者受付 利用案内、利用申請、利用者講習会、広報誌の発行、利用負担金、図書の見出し、貸出	情報推進部情報企画課 利用者サービス係 usersv@cmc. (吹田本館 1F)	06-6879-8808		
大規模計算機システムの運用・管理、セキュリティに関すること	情報推進部情報基盤課 教育・研究システム班 system@cmc. (吹田本館 2F)	06-6879-8813		
ネットワークの運用・管理、セキュリティに関すること	情報推進部情報基盤課 ネットワーク班 network@cmc. (吹田本館 2F)	06-6879-8817		
プログラムのメール等相談	メール利用相談員 toiawase@cmc. 利用者サービス係 (吹田本館1F)	Tel : 06-6879-8808 Fax : 06-6879-8814	常時 毎週月曜日 から 毎週金曜日 まで	室
計算機説明書の閲覧 PC機器、情報コンセント	相談員・利用者控室 (吹田本館 1F)		9:00～17:15※1	

●計算機運転サービス時間

オンライン・サービス〈センター外端末からの利用〉 (注)	
すべての計算機	終日運転

(注) * 平日の17時以降及び土曜日・日曜日・祝日は自動運転を行っていますが、障害が発生した場合は、その時点でサービスを中止することがあります。

* スーパーコンピュータは、毎月の第1月曜日8:30～11:00に保守点検のため停止します。

* サービス時間は原則として上記の時間となっていますが、状況によって変更する場合がありますので、ご了承ください。

* E-mailアドレスの末尾にはosaka-u.ac.jpをお付け下さい。

※1 玄関は原則として午後5時15分に閉館いたしますが、相談員・利用者控室の情報機器をご利用される場合には、延長利用が可能ですので、利用者サービス係窓口までお問合せください。

特集 SX-8R

・スーパーコンピュータ SX-8R の特長と概要	-----	3
・スーパーコンピュータ SX-8R ハードウェア	-----	8
・FORTRAN90/SX の自動並列化機能	-----	15
・FORTRAN90/SX の自動ベクトル化機能	-----	30

スーパーコンピュータSX-8Rの特長と概要

Major Features and Outline of Supercomputer SX-8R

梁川 貴志 山下 晃弘 萩原 孝 篠原 真史 岩田 直樹 長沢 富人

1. まえがき

近年、解決すべき問題がより大規模化していく一方で研究・開発期間の短縮が求められ、コンピュータによる問題解析や設計の高速化、最適化に対する要求がよりいっそう厳しくなっています。このため、スーパーコンピュータの高速化に対するニーズはますます高まっています。

そこでNECでは、最近のスーパーコンピュータに求められる以下のニーズを満足するプラットフォームとして、この度、SX-8Rを開発しました。

- ・容易に高い実効性能が得られること
- ・価格性能比が優れていること
- ・信頼性が高く、運用コストが低いこと
- ・使いやすく、標準化に対応したシステムであること

2. SX-8Rの特長

SX-8Rには、以下のような大きな5つの特長があります。

(1) 世界最高速クラス1チップベクトルプロセッサ

SX-6から採用した1チップベクトルプロセッサを、最先端のCMOS LSIテクノロジーの採用によりさらに強化し、世界最高速クラスのプロセッサあたり35.2GFLOPS（大容量メモリ）という高性能を達成しています。

(2) スケーラビリティに優れたマルチプロセッサシステム

SX-8Rは、シングルノードシステムでの共有メモリアーキテクチャの使いやすさと、マルチノードシステムでの高いスケーラビリティを併せ持つ、共有・分散メモリアーキテクチャを採用しています。シングルノードシステムでは最大8CPU・281.6GFLOPS、マルチノードシステムでは最大4,096CPU・144TFLOPS（いずれも大容量メモリ）のベクトル演算性能を実現しており、最小構成から最大構成までの間で4,000倍もの性能レンジをカバー

しています。

(3) 高密度実装による小型化と優れた設置性

プロセッサの省電力設計により、消費電力・発熱量を削減しているほか、最先端の高密度実装技術により、CPUと主記憶装置をコンパクトな1つのモジュールに収容して筐体をコンパクト化し、設置面積も大幅に削減しています。この結果、電力コストやフロアコスト等の運用コストも大幅に削減しています。

(4) 実績のあるオペレーティングシステムと上位互換性

海洋研究開発機構の世界最速クラススーパーコンピュータ「地球シミュレータ」のハードウェアおよび基本ソフトウェアの基本設計と製造を担当した経験を生かし、超大規模構成での優れた運用性を実現しています。また、従来のSXアーキテクチャを継承することにより、高い上位互換性を確保しています。オペレーティングシステムはSX-3から採用し、豊富な実績を持つ業界標準のUNIXに準拠したオペレーティングシステムSUPER-UXを継続して採用するとともに、さらなる強化をおこなっています。

(5) 充実のソフトウェア開発・利用環境と豊富なアプリケーションソフトウェア

実績あるベクトル処理、および共有メモリ/分散メモリの両方の並列処理に対応するソフトウェア開発環境を提供しています。さらに、グリッドコンピューティングの進展を考慮して、各種UNIX/Linux機やPCが混在する統合環境でもアプリケーションソフトウェアへのアクセスや操作が容易に行えるWebSuperComputing環境を実現しています。

また、高い上位互換性が確保されているため、従来から蓄積されてきたSXシリーズ向けに高度にチューニングされた豊富なアプリケーションソフトウェアを、継続して利用できます。

3. ハードウェアの概要

SX-8Rの製品ラインナップは、図1に示すように、

最大8台の中央処理装置(CPU)と主記憶装置(MMU)を密結合した、メモリ共有型のSX-8Rシングルノードシステムと、複数のシングルノードシステムをノード間接続装置によりクラスタ接続した、SX-8Rマルチノードシステムから構成されます。

SX-8Rシングルノードシステム(大容量メモリ)は、最大8CPU・281.6GFLOPS、主記憶容量256Gバイト、入出力チャンネル数最大27チャンネルのAモデル、および最大4CPU・140.8GFLOPS、主記憶容量128Gバイト、入出力チャンネル数最大27チャンネルのBモデルの2モデルから構成されています。いずれも演算性能、メモリスループット、入出力性能などのトータルバランスに優れ、高い実効性能を發揮します。

マルチノードシステムは、複数のシングルノードシステム間を超高速に接続する専用のノード間接続装置(Internode Crossbar Switch : IXS)により、2台 ~ 512台のノードをクラスタ接続したシステムであり、最大4,096CPU・144TFLOPSの驚異的なベクトル演算性能を發揮します。

(1) 中央処理装置(CPU)

高速に演算を実行するCPUは、ILSIチップ内に実装された1組のスカラユニットと、4組のベクトルユニットから構成されています。クロックサイクルの高速化と1チップ化によるプロセッサ内部デザインの改善により、短いベクトル長でも高性能を実現しています。ベクトルユニットは、乗算/論理演算、乗算、加算/シフト演算、加算/除算/平方根演算、マスク演算、ロード/ストアの各パイプラインから構成され、これらのパイプラインを4セット搭載した多重並列パイプライン方式を採用しています。

SX-8/SX-8Rで追加されたベクトル平方根演算器は、ハードウェアによる平方根演算のベクトル処理を可能とし、本演算機能が有効なプログラムでは、大幅に性能が向上します。スカラユニットも最大4命令を同時実行するスーパースカラ方式や投機実行の採用、データパイプラインの強化などにより、ベクトル化されにくい部分についても高い実効性能が得られるように強化されています。

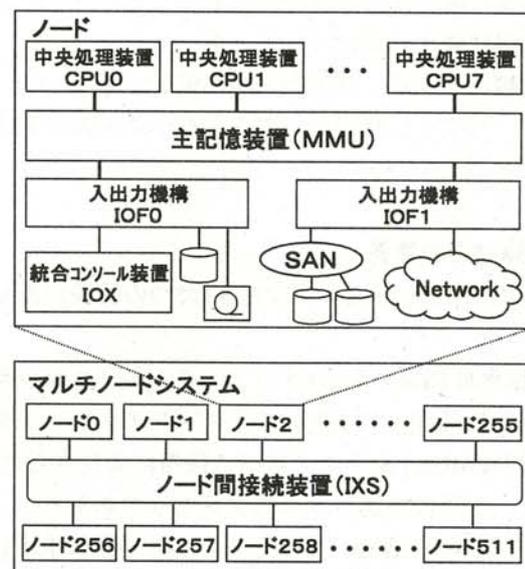
(2) 主記憶装置(MMU)

MMUは、記憶素子に大容量メモリと高速メモリの2種類のRAMを採用し、ユーザの利用条件やアプリケーションの特性によってどちらか一方を選択できるようになっています。シングルノードシステムでは最大256Gバイト、563.2Gバイト/秒、マルチノードシステムでは最大128Tバイト、288Tバイト/秒の

大容量、高メモリスループットを実現しています。この大容量メモリとプロセッサへの高いデータ供給能力により、ベクトル処理を効率よく実行することが可能となっています。

(3) 入出力機構(IOF)

IOFはシステムのスループットを高く保つためにプロセッサ性能・メモリ性能に見合った高速なデータ転送能力を持っています。SX-8Rではシングルノードシステムで最大27チャンネル・6.4Gバイト/秒、マルチノードシステムでは最大13824チャンネル・3276.8Gバイト/秒という高速総合データ転送能力を実現します。入出力インタフェースも、ファイバチャンネルインタフェースを始め、Ultra320 SCSIチャンネルなどの汎用インタフェースをサポートしており、多彩な周辺機器を接続できます。ネットワークインタフェースもジャンボフレームに対応したギガビット・イーサネットなどとの接続が可能です。



(4) ノード間接続装置 (IXS)

シングルノードシステムでは処理が困難な、より大規模な演算を高速に処理するために、複数のシングルノードシステム間をクラスタ接続したマルチノードシステムがあります。マルチノードシステムでは、ノード当たり16Gバイト/秒 ×2 (双方向) の超高速ネットワークで各ノードをクロスバ結合するノード間接続装置(IXS)を使用してノード間のデータ転送を行います。これにより、最大データ転送性能8,192Gバイト/秒の超高速ノード間接続を実現し

ます。

(5) 最先端テクノロジーの採用による設置性向上
SX-8Rは最先端の高密度実装技術の採用、および、

低消費電力設計により、設置性の向上を実現しております。また、全面的な空冷方式を採用し、既存の環境に柔軟に対応します。

表 SX-8Rの主な諸元
Table SX-8R System Specification

項目			SX-8R(シングルノード)						SX-8R(マルチノード)		
モデル名			1B	2B	4B	4A	6A	8A	16M2	128M16	4096M512
大容量メモリ (DDR2-SDRAM)	中央処理装置 (CPU)	CPU数	1	2	4	4	6	8	16	128	4096
		最大ベクトル性能	35.2GFLOPS	70.4GFLOPS	140.8GFLOPS	140.8GFLOPS	211.2GFLOPS	281.6GFLOPS	563.2GFLOPS	4505.6GFLOPS	144TFLOPS
	主記憶装置 (MMU)	容量	32/64/128GB						512GB	4096GB	131072GB
最大データ転送性能		70.2GB/秒	140.4GB/秒	281.6GB/秒	281.6GB/秒	422.4GB/秒	563.2GB/秒	1126.4GB/秒	9011.2GB/秒	288358.4GB/秒	
高速メモリ (Network FCRAM)	中央処理装置 (CPU)	CPU数	1	2	4	4	6	8	16	128	4096
		最大ベクトル性能	32GFLOPS	64GFLOPS	128GFLOPS	128GFLOPS	192GFLOPS	256GFLOPS	512GFLOPS	4096GFLOPS	131TFLOPS
	主記憶装置 (MMU)	容量	16/32/64GB						256GB	2048GB	65536GB
最大データ転送性能		64GB/秒	128GB/秒	256GB/秒	256GB/秒	384GB/秒	512GB/秒	1024GB/秒	8192GB/秒	262144GB/秒	
大容量/高速メモリ 共通	入出力機構 (IOF)	最大チャネル数	27CH						54CH	432CH	13824CH
		最大転送速度	6.4GB/秒						6.4GB/秒	12.8GB/秒	102.4GB/秒
	ノード間接続装置	最大データ転送性能	-						32GB/秒	256GB/秒	8192GB/秒

(6) 豊富なRAS機能

SX-8Rは、高集積設計により、ケーブルなどの部品点数の大幅な削減を実現し、ハードウェアの信頼性を飛躍的に向上させています。また、従来機種同様、主記憶装置における誤り訂正符号の採用はもちろん、回路の二重化、パリティチェックなどによる誤り検出機能が多数組み込まれています。障害発生時には、故障発生箇所を自動的に指摘するビルトイン診断機能(BID)により速やかに故障箇所が検出され、自動再構成機能により故障箇所が自動的にシステムから切り離されて、システムの運転が継続されます。また、障害情報の自動収集、サービスセンタへの自動通報、サービスセンタからの遠隔保守による迅速な故障診断と容易な予防保守を実現し、システムの信頼性・稼働性・保守性を総合的に高めています。

4. ソフトウェアの概要

SX-3での採用以来、17年にわたり信頼と実績を築き上げてきたオペレーティングシステムのSUPER-UXをさらに強化して提供しています。SUPER-UXは、オペレーティングシステムとして業界標準であるUNIXに準拠しています。しかも、BSD機能を取り込み、その特長を継承するとともに、高速入出力機能のサポート、大規模マルチノードシステムへの対応など、スーパーコンピュータにふさわしい多彩な機能を追加しています。さらに、自動ベクトル化、自動並列化コンパイラにより容易にSXシリーズの高性能を引き出すことができます。ロードモジュールは上位互換のため、従来のSXシリーズ上で作成したプログラムを再コンパ

イルすることなくSX-8R上で実行可能です。

SUPER-UXの主要な機能について以下に説明します。

- (1) 強力なリソース活用・管理機能
 - ・マルチプロセッサと並列処理をさらに強化してサポートし、マルチノードにおけるシングルシステムイメージ(SSI)を実現。
 - ・システムの持つ多数のCPUや大規模なメモリ資源を分割管理するリソースブロック機能。
- (2) 高速入出力/大容量ファイル機能
 - ・大容量高速ファイル入出力のためのファイルシステム(SFS)を提供し、入出力性能を大幅に向上。
 - ・大容量の主記憶上に通常のファイルシステムを構築して高速の入出力を可能にし、ディスクキャッシュとして利用するメモリファイル機能(MFF)。
 - ・ファイルアーカイビングシステムSX-BackStore
 - ・テープライブラリ装置を効率的に管理するvolcenter
- (3) 強力なバッチジョブ処理機能/ジョブスケジューリング機能
 - ・SSIを実現したバッチ処理システムNQSII(Network Queuing System II)。

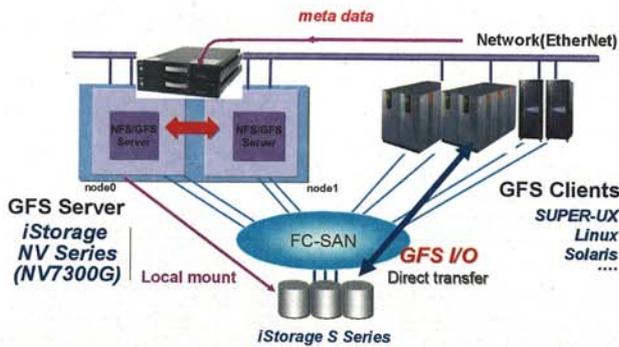


図2 GFS の概念図
Fig.2 GFS Concept

- ・ 拡張リソーススケジューラERSII (Enhanced Resource Scheduler II) によるフェアシェアスケジューリング。
- ・ バックファイルスケジューラJobManipulatorによるマルチノードシステムの効率利用、フェアシェアスケジューリング。
- (4) 多彩な分散ネットワーク環境
 - ・ 分散ファイルシステムNFSの提供。
 - ・ 複数ノード、あるいは異機種間において高速にファイルを共有するGFS(gStorageFS)により、SXシリーズ、TX7シリーズ(OS: Linux)やSUN、HPなどとの間のファイル共有を実現 (図2を参照)。
- (5) 快適な運用管理機能
 - ・ 実行中のプログラムを任意の時点で中断させ、後に再開させることを可能にするチェックポイント・リスタート機能。
 - ・ 分散ネットワーク管理の業界標準であるHP OpenViewを採用し、ネットワーク上の複数のホストマシンを、1台の管理用ホストマシンにより統合的に運用管理するためのソフトウェア MasterScope/SX。
 - ・ 自動運転制御装置(AOC)を使用したシステム運用の省力化・無人化。
- (6) 新標準や技術の進歩への対応
 - ・ 多様なニーズに応える、Fortran95、C++、OpenMP、MPI-2などの最新プログラミング規格のサポート。
 - ・ I/O、ネットワーク関連の最新標準プロトコルのサポート。
- (7) 快適なプログラム開発環境
 - ・ FORTRAN90/SXおよびC++/SXコンパイラは、SXシリーズ上で動作するセルフ版のほか、SUN、Linux製品上でのクロス版も提供。
 - ・ 統合プログラム開発環境PSUITE、分散並列デバuggTotalView、性能解析ツールITAなどによる使いや

すく、効率的なプログラム開発環境。

- ・ POSIX (Portable Operating System Interface for Computer Environment) インタフェースのサポートによるアプリケーションソフトウェアの移植性向上。

5. むすび

以上、SX-8Rのハードウェアとソフトウェアについて、その特長と概要をご紹介しました。SX-8Rはベクトルアーキテクチャならではの高い実効性能と同時に、優れたコストパフォーマンス、経済的な運用コスト、使いやすさを備えたスーパーコンピュータです。今後、多くのSX-8Rシステムが様々な分野で活躍できるよう、積極的にお客様への提案を行っていく所存です。

ビジネスの基本となるものは、利用者の期待と要望に応えることのできる製品と考えています。そのために、今後も最先端技術を取り入れ、スーパーコンピュータ製品の強化・開発を進めてまいります。皆様のご意見とご指導の程、よろしくお願い申し上げます。

*UNIX は The Open Group の米国およびその他の国における登録商標です。

*Linux は Linus Torvalds 氏の米国およびその他の国における商標、または登録商標です。

*Ethernetは米国XEROX社の登録商標です。

*NFSは米国Sun Microsystems社の登録商標です。

*OpenMPはOpenMP Architecture Review Boardの登録商標です。

*TotalViewはEtnus社の米国における登録商標です。

*SUNは米国Sun Microsystems, Inc.の米国およびその他の国における商標または登録商標です。

*HP、OpenViewは米国Hewlett-Packard社の登録商標です。

*POSIXはInstitute of Electrical and Electronics Engineers, Inc. の登録商標です。

筆者紹介

梁川 貴志 日本電気株式会社 コンピュータソフトウェア事業部本部第一コンピュータソフトウェア事業部エキスパート

山下 晃弘 NECシステムテクノロジー株式会社 プ
ラットフォーム事業本部サーバソフトウェア事業部
グループマネージャー

萩原 孝 日本電気株式会社 第一コンピュータ事業
本部コンピュータ事業部エキスパート

篠原 真史 日本電気株式会社 第一コンピュータ事
業本部コンピュータ事業部マネージャー

岩田 直樹 日本電気株式会社 第一コンピュータ事
業本部HPC販売推進本部マネージャー

長沢 富人 日本電気株式会社 第一コンピュータ事
業本部HPC販売推進部主任

スーパーコンピュータ SX-8Rハードウェア

日本電気株式会社 コンピュータ事業部
HPC販売推進本部

1. はじめに

SX-8R は、SX シリーズで実証された高い実効性能と共有メモリの利便性を継承しつつ、さらに増大する科学技術計算ニーズに応えるため、システム全体性能の強化はもとより、単一 CPU 性能を大幅に強化しているシステムです。ノード当り 8 台の中央処理装置 (CPU) を有し、ノード間は専用の超高速クロスバネットワークで接続可能で、実行演算性能、スケーラビリティ、及び使いやすさに優れたスーパーコンピュータです。特長は以下の通りです：

- ① 90nm 銅配線技術を用いた超高速、高集積 CMOS LSI による高密度実装
- ② CPU への、ベクトル平方根演算器の追加による平方根演算性能強化
- ③ スカラユニットにおけるスーパースカラ方式、投機実行、データパイプライン強化、及びクロックサイクル強化による高いスカラ実効性能
- ④ 省電力設計による消費電力・発熱量の削減、及び高密度実装による設置面積の削減

以下では、上記特性を備えた SX-8R のアーキテクチャ、システム構成、及びテクノロジーの概要について説明します。

2. SX-8R のハードウェア

2.1. ハードウェア構成

SX-8R は、CPU と主記憶装置 (Main Memory Unit : MMU) を密結合した共有メモリ型のシングルノードシステムを、超高速なノード間接続装置 (Internode Crossbar Switch : IXS) によりクラスタ接続することにより、分散並列処理を可能としたシステムです。システム構成を図 1 に示します。

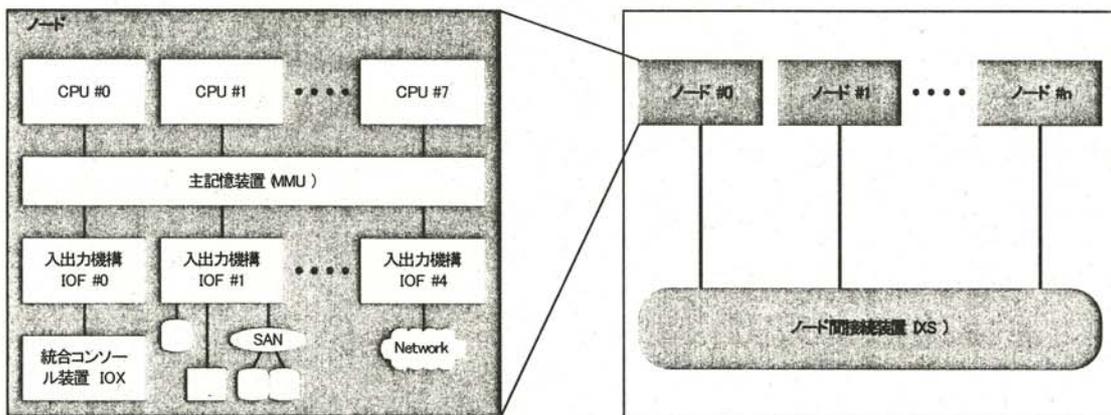


図 1 SX-8R システム構成

表 1 SX-8R システム主要諸元 (シングルノードシステム)

中央処理装置 (CPU)	CPU 数	8
	最大ベクトル性能	281.6* ¹ /256* ² GFLOPS
主記憶装置 (MMU)	最大容量	256* ¹ /128* ² GB
	最大データ転送性能	563.2* ¹ /512* ² GB/s
入出力機構 (IOF)	最大チャンネル数	55 CH
	最大転送速度	12.8 GB/s

* 1 : 大容量メモリ使用時 / * 2 : 高速メモリ使用時

シングルノードシステムは、CPU 数 8 台、最大ベクトル性能 281.6*¹/256*² GFLOPS、最大主記憶容量 256G バイト*¹/128G バイト*²、最大入出力チャンネル数 55 であり、演算性能、メモリスループット性能、入出力性能などのトータルバランスに優れ、高い実効性能を実現します。

設置性と運用面に関しては、1 チップベクトルプロセッサ、及びメモリモジュールを高密度実装技術を用いて装置パッケージングすることにより、設置環境、及び消費電力を従来の SX シリーズと比較して大幅に改善しています。

2.2. CPU

SX-8R の CPU は従来の SX アーキテクチャを継承しつつ、さらなる機能・性能の強化を図っています。図 2 に CPU の構成を示します。CPU は、スカラユニット部、及びベクトルユニット部により構成され、プロセッサ/メモリ間ネットワークを介して MMU と接続されます。スカラユニットは命令の解読、ベクトルユニットへのベクトル命令の供給・起動、及びスカラ命令の実行を行います。一方、ベクトルユニットは 4 セットのベクトルパイプラインを備え、2.2*¹/2*²GHz で動作します。クロックサイクル当たり 8 個の加算と乗算を同時に処理することが可能であり、最大 35.2*¹/32*²GFLOPS のベクトル演算性能を実現します。

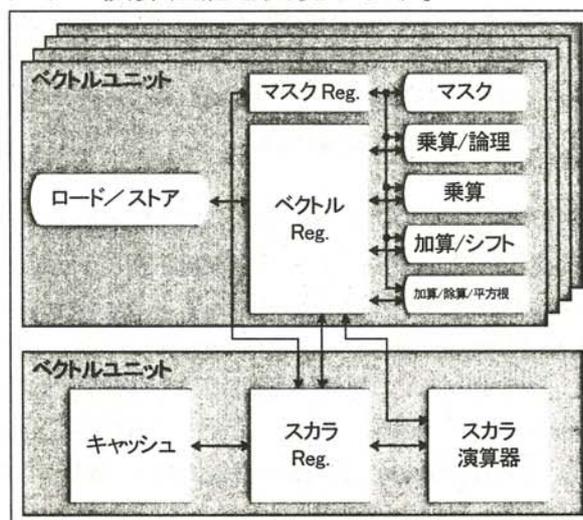


図 2 CPU 構成

* 1 : 大容量メモリ使用時 / * 2 : 高速メモリ使用時

ベクトルユニットはベクトル演算部、及びベクトル制御部から構成されます。ベクトル演算部は、乗算／論理演算、乗算、加算／シフト演算、及び加算／除算／平方根演算の4種のそれぞれ独立に動作可能なパイプライン、マスク演算パイプライン、ロード／ストアパイプライン、マスクレジスタ、及びベクトルレジスタにより構成されています。ベクトル演算部全体は、以上のユニット4セットからなる多重パイプライン構成であり、強力にベクトル演算処理を実行します。特に、SX-8Rは除算パイプラインに平方根演算機能を追加することにより、倍精度浮動小数点平方根演算において $1.1^*1/1^*2$ GFLOPSという高い演算性能を実現しています。

スカラーユニットは64ビットRISCアーキテクチャであり、L1キャッシュ、命令同時デコード数4、命令同時発行数4であるスーパースカラアーキテクチャ、アウトオブオーダー実行、及び命令投機実行の採用により短ベクトル処理性能、及びスカラー性能を大幅に向上させ、 $2.2^*1/2^*2$ GFLOPSの最大演算性能を実現しています。

2.3. 主記憶装置 (MMU)

スーパーコンピュータにおいて高い実効性能を実現するためには、高い演算性能に見合うだけの高いデータ供給性能が必要となります。SX-8RのMMUは、CPUとリモートアクセス制御装置 (Remote access Control Unit : RCU) に接続され、高速、かつ均一にメモリアクセス可能な共有メモリ方式を採用しています。メモリを4096個のバンクに分割し、それぞれ独立に制御するインターリーブ方式、及びメモリバンクキャッシュの採用によりバンク競合を最小限に抑えつつ、高いメモリスループット性能を実現しています。

これによりSX-8Rは、シングルノードシステムにおいて $256^*1/128^*2$ Gバイトのメモリ容量、及び $563.2^*1/512^*2$ Gバイト/秒のメモリスループット性能を実現します (表1参照)。また、MMUは通信レジスタ (Communication Register : CR) を備えており、自動並列処理や、Open MP指示行による共有メモリ内並列処理時のCPU間同期制御を高速に実行することが可能です。

また、お客様の幅広いニーズに応えるために1Gビット/512MビットDDR2-SDRAM (Double Data Rate-SDRAM:大容量メモリ) とFCRAM (Fast Cycle RAM:高速メモリ) の2種類のRAMを採用し、高速アクセス・大容量メモリを提供しています。両者の大きな違いはRAMのスペックであるランダムアクセス時間 (tRC) が異なる点です。DDR2-SDRAMは安価のため大容量のメモリ実装が可能であり、FCRAMは高速であるためバンク競合による待ち時間を最小限に抑えることが出来ます。

SX-8Rは、SX-4以降採用している3次元実装構造のMMUカード実装を引き続き採用しています。これにより、RAMとメモリ制御部間の物理的距離を短くし、RAMの高速動作、及びRAMとCPU間の高速信号伝送を実現しています。一方、メモリ信頼性確保のためのECC (Error Check and Correct) 符号、タイミング、パリティ、2重化回路などの採用による高いメモリ故障検出率の実現、擬似障害によるチェック回路の診断機能、エラー内容から即座にエラー箇所を指摘するビルトイン機能などにより、RAS (Reliability, Availability, Serviceability) 機能の充実を図り信頼性を高めています。

*1 : 大容量メモリ使用時 / *2 : 高速メモリ使用時

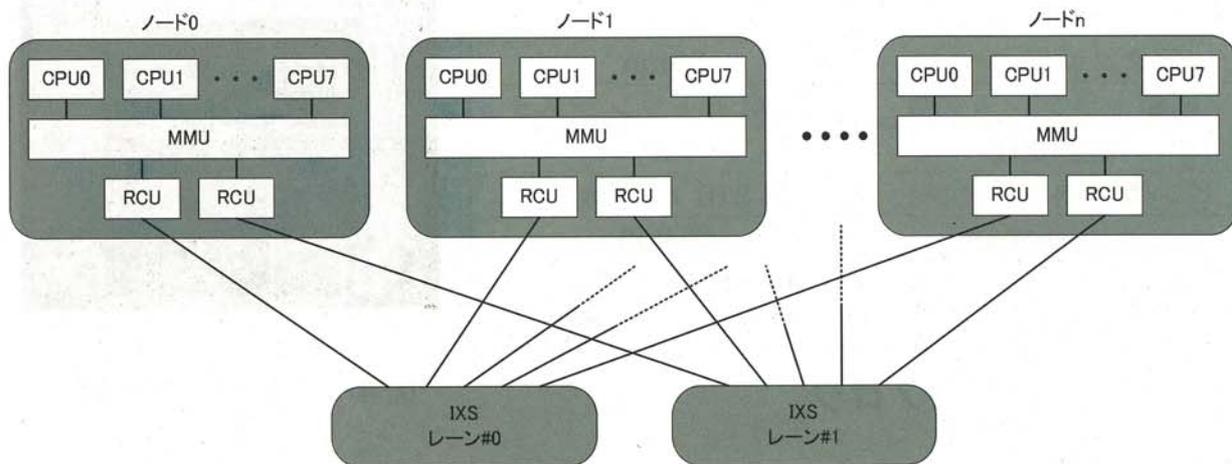


図3 ノード間接続

2.4. 入出力装置

入出力機構 (Input Output Features : IOF) はシステムのスループットを高く保つために、SX-8R の高いプロセッサ性能、及びメモリ性能に見合った高速なデータ転送性能を備えており、ノード当り最大 55 チャンネル、最大 12.8G バイト/秒の性能を有します。入出力インタフェースは、ファイバチャネル、Ultra320 SCSI チャンネルなどの汎用インタフェースをサポートしており、様々な周辺機器を接続することが可能です。また、ネットワークインタフェースとして、ジャンボフレームに対応したギガビット・イーサネットなどが利用可能です。

I/O 動作において、CPU は全ての I/O 装置に対して対等にアクセスすることが可能であり、実行負荷の低い CPU を I/O 制御に割り当てるなど、CPU の効率的利用を可能としています。また、I/O 制御方式として Direct I/O 方式を採用し、低レイテンシ、かつ高スループットな I/O 制御を実現しています。

2.5. ノード間接続装置

SX-8R は図3に示すように、共有メモリ型のシングルノードシステムを超高速専用クロスバススイッチである IXS を介して結合することにより、分散並列処理を可能としています。各シングルノードシステムは、RCU を介して IXS とケーブル接続され、分散並列処理において低通信レイテンシ、及び高通信スループットを実現します。

RCU はノード当り 2 台実装され、各 RCU のデータ受信部、及び送信部はそれぞれ独立に動作可能であり、ノード当り最大 16G バイト/秒×2 (双方向) の通信バンド幅を実現します。また、RCU は CPU とは独立に動作するデータムーバを持つことにより、異なるノードのメモリ間でデータ転送を行なうリモートメモリアクセスを、CPU 動作とは完全に独立して行なうことが可能です。IXS は回線交換型のデータ交換を行なうクロスバススイッチです。各ノードの RCU とのインタフェースを持ち、ポート当り 8G バイト/秒 (双方向) のスループット性能を有しています。

表 2 CPU チップ諸元

項目	
テクノロジロード	90nm
搭載トランジスタ数	8,800 万トランジスタ
電源電圧	1.0V
ピン数 (内信号ピン)	8,210 (1,923)
配線層構成	銅 9 層
I/O インタフェース	1.0V シリアル
実装形態	ベアチップ実装

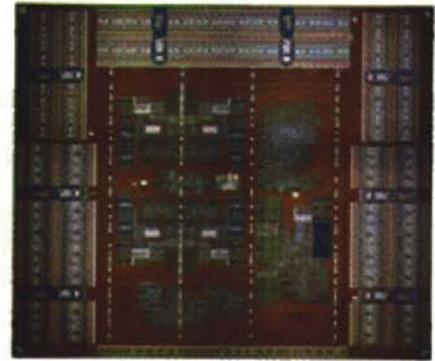


図 4 CPU チップ外観

3. SX-8R のテクノロジー

3.1. LSI 技術

SX-4 以降、SX シリーズでは CMOS テクノロジによる高集積化、及びプロセッサの平行化により高性能化を実現しつつ、コストパフォーマンスを向上させてきました。SX-8R では、さらに高い性能を実現するために LSI 技術、及び回路技術を開発させています。

システムの性能向上のためには、LSI 間信号伝送の高速化も必要になります。SX-8R では、低レイテンシのマルチチャネル・シリアル・インタフェースを開発することにより、LSI 間的高速データ転送を実現しています。また、インタフェース回路を低消費電力化、小面積化することにより LSI への多チャネルの搭載を実現しています。図 4、及び表 2 に SX-8R の CPU チップ外観、及び諸元をそれぞれ示します。

SX-8R は LSI 共通仕様として、90nm CMOS プロセス、9 層銅配線、及び低誘電率層間絶縁膜などの採用による配線遅延の改善、ラジカル窒素プロセスの開発によるゲート酸化膜の薄膜化による高性能な低電圧電源の実現、新規低誘電率層間膜材料 (K=2.9) の採用による配線ピッチ微細化に伴う配線寄生容量増加の低減などを行っています。さらに、チップ当り世界最高の実効データ伝送レートを実現するマルチチャネル・シリアル・インタフェースの開発により、CPU/MMU 間の低転送レイテンシを実現、同時に LSI 上の電源分離、アナログ回路の削減、制御信号のデジタル化などによりノイズ耐力、及びエラーレートの格段の向上を実現しています。

LSI 内部の RAM 回路は、専用設計によりデバイス性能を最大限引き出すことが可能です。一方、LSI の低消費電力化のために、従来のワンショット動作と非選択回路の停止に加え、読み出し回路の制御信号簡略化によるダイナミックパワーの削減、及びリーク電流の少ないトランジスタの利用によるスタティックパワーの削減を実現しています。また、高速なクロック動作を実現するために、LSI 外部からクロックを逡倍する APLL (Analog Phase-Locked Loop) 回路を採用しています。

高速システムにおける処理能力の向上には、LSI 内信号伝送の高速化とともに、LSI 間信号伝送の高速化が必要となります。同様に、信号伝送の高速化を妨げる要因となる電源ノイズ対策も重要になります。SX-8R では高速、かつ安定した信号伝送を実現するために、信号伝送時の減衰が小さい低損失材料を使用した基板、伝送信号の波形を改善するプリエンファシス機能を備えた回路、及び波形ひずみの少ないソケットやコネクタなどを採用しています。また、トランジスタが高速化し、電源電流の時間変化が大きくなることにより電源ノイズが増加するため、デカップリング用コンデンサの搭載数最適化などにより電源ノイズの低減を実現しています。

3.2. 実装技術

SX-8R は、世界最高性能、高コストパフォーマンス、及び優れた設置性を実現するために、高密度 LSI 実装技術、高密度接続技術、高効率冷却技術、及び高性能電源モジュール技術により、従来のワンチッププロセッサをさらに進化させました。

CPU、及び MMU モジュールの外観を図 5、及び図 6 に、CPU、及び MMU モジュールの諸元を表 3 にそれぞれ示します。超高速動作が要求される CPU/MMU モジュールは、高密度実装により大型で多ピンの LSI を搭載可能としています。CPU モジュールは、CPU LSI をビルドアップ基板表面にベアチップ実装し、裏面には入出力用のパッドを並べた LGA (Land Grid Array) が形成されています。MMU モジュールは、メモリ制御用の MMC LSI とメモリキャリア用コネクタをビルドアップ基板表面に実装しています。一方、SX-8R は環境配慮型製品として、ベアチップ接続部を含む全てのはんだ付け部に Sn-Ag-Cu 系の鉛フリーはんだを採用しています。

次にシステム実装技術について述べます。SX-8R はノードボードと呼ばれるメイン

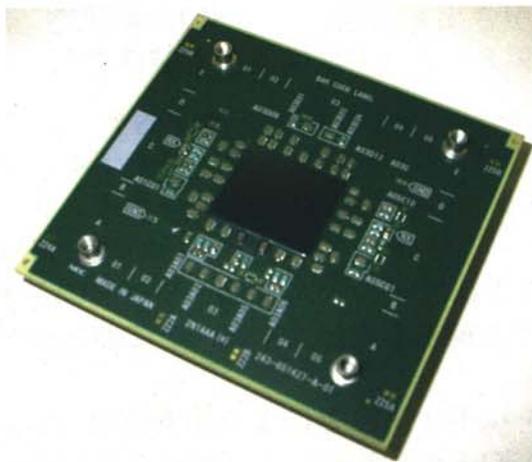


図 5 CPU モジュール外観

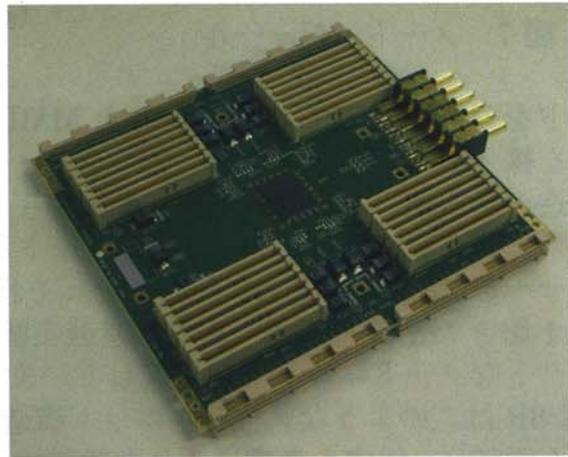


図 6 メモリモジュール外観

表 3 CPU/MMU モジュール諸元

項目		CPU モジュール	MMU モジュール
搭載 LSI (形態)		CPU LSI×1 (ベアチップ)	MMC LSI×1 (ベアチップ)
	ピン数	8210	5284
	IO ピッチ (μm)	182	182
搭載 RAM		—	μBGA×96
配線基板	種類	ビルドアッププリント配線基板	
	基板サイズ (mm)	85×75	130×145
	基板厚 (mm)	1.8	1.7
	基板層数	5 ビルドー8 コアー5 ビルド	
	配線密度 (μm)	配線幅/間隙=20/20	
	via 仕様 (μm)	via 径/ランド径=50/80	
モジュール	入力端子数	5329	824
	冷却	空冷	
	消費電力 (W)	75	95

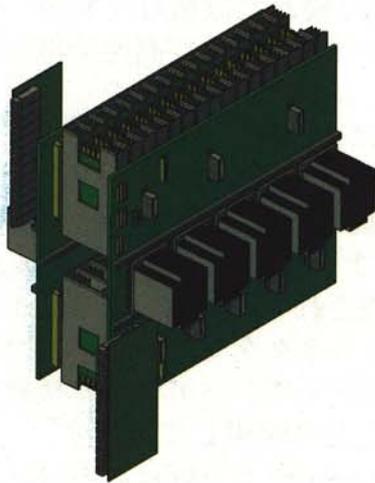
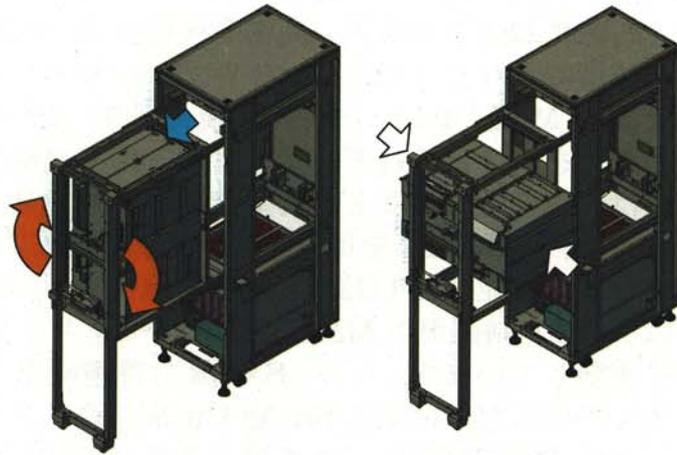


図7 ノードモジュール



(a)引き出し状態 (b)回転状態
図8 ノードユニット実装構造

ボードを介して、CPU モジュールと MMU モジュールを相互接続するケーブルレス構造を採用しています。これにより、1 ノードの実装密度の改善、ケーブルレスによる製造の効率化、及び部品点数の削減によるコスト低減を実現しています。ノードユニットは図7に示すように、2枚のノードボードを向かい合わせ、その間にMMUモジュールを32枚挟み込み、両面にそれぞれCPUモジュール4個、及びRCUモジュール1個をMMUモジュールとは逆面実装する構造となっています。これにより、コンパクトなノードユニットを実現しています。

SX-8Rはこのようなノードユニット構造を採用することにより実装密度を高めていると同時に、保守を考慮に入れた設置性の高い筐体構造となっています。図8にノードユニットの構造を示します。図8(a)に示すように、ノードユニットは筐体から引き出すことにより表面に実装されたCPUモジュールの保守を可能としています。さらに図8(b)に示すように、ノードユニットを回転させることにより、ノードボード間のMMUモジュールの保守を可能にしています。このため、1方向からのアクセスのみによりノードユニットの4方向へのアクセスを可能にし、保守性、及び設置性を大幅に改善しています。

4. おわりに

以上、SX-8Rのハードウェアの概要について紹介させていただきました。SX-8Rは、スーパーコンピュータの要件であるCPUの高い演算性能と、それに見合う主記憶からのデータ供給能力のバランスを重視し、ユーザに使いやすい大規模な分散共有メモリ型スーパーコンピュータとして開発しております。NECは、今後も様々な研究分野の発展を支える強力なツールとしてお客様のご要望を製品開発に反映させたスーパーコンピュータを開発していきます。

皆様がSX-8Rをご利用頂く上で、本稿がお役に立てれば幸いです。

FORTRAN90/SX の自動並列化機能

日本電気株式会社 第一コンピュータソフトウェア事業部

概要

FORTRAN90/SX は、SX-8R のもつ並列処理機能を利用して、その性能を十分に引き出すための高度な自動並列化機能を備えている。本文では、並列処理の概念、自動並列化機能、および並列化促進のための技法について紹介する。

1. はじめに

SX-8R は、既の実績のあるベクトル処理に並列処理機能を融合した「スケーラブル・パラレル・スーパーコンピュータ」です。このハードウェアのもつ高い能力をいかんなく発揮させるためには、コンパイラのベクトル化/並列化機能が重要な役割を果たします。

FORTRAN90/SX は、ハードウェアに密着した高度な最適化、ベクトル化、並列化機能を有した Fortran95 コンパイラです。言語仕様としては、Fortran95 (JIS X3001-1:1998) をサポートしています。SX-8R は、1 ノード内 8 台のプロセッサの範囲内では、利用しやすい共有メモリ方式を採用しており、FORTRAN90/SX の並列処理は、この共有メモリ方式を使った並列化機能を提供しています。

一般に、並列化を行う時には、並列実行しても結果が変わらないことを保証するために、データの依存関係の解析を行い、細心の注意を払ってプログラムの変形や指示行の挿入をしなければなりません。自動並列処理機能を利用すると、それらの作業を自動的にコンパイラが行います。また、ベクトル化と同様に、効果的に並列化を行うためのオプションや指示行が用意されており、十分にプログラムのチューニングを行うことが可能です。本稿では、並列処理の概念と、自動並列化機能および関連する指示行についてご紹介します。

2. 並列処理とは

並列処理とは、1つの仕事をいくつかの小さな仕事に分割し、それを複数のタスク (CPU) で並列に実行することです。FORTRAN90/SX コンパイラが備えている自動並列処理機能とは、「コンパイラがプログラムを解析して、並列に実行可能なループや文の集まりを抽出し、ループの繰り返しや文の集まりを複数のタスクに自動的に割り当てて実行時間を短縮する機能」です。

コンパイラが、do ループを 4 つのタスクに分割して実行するイメージは、次の図ようになります。この例では、外側ループの 100 回の繰り返しを 4 つに分割して、各 CPU 上で各々を並列に実行します。

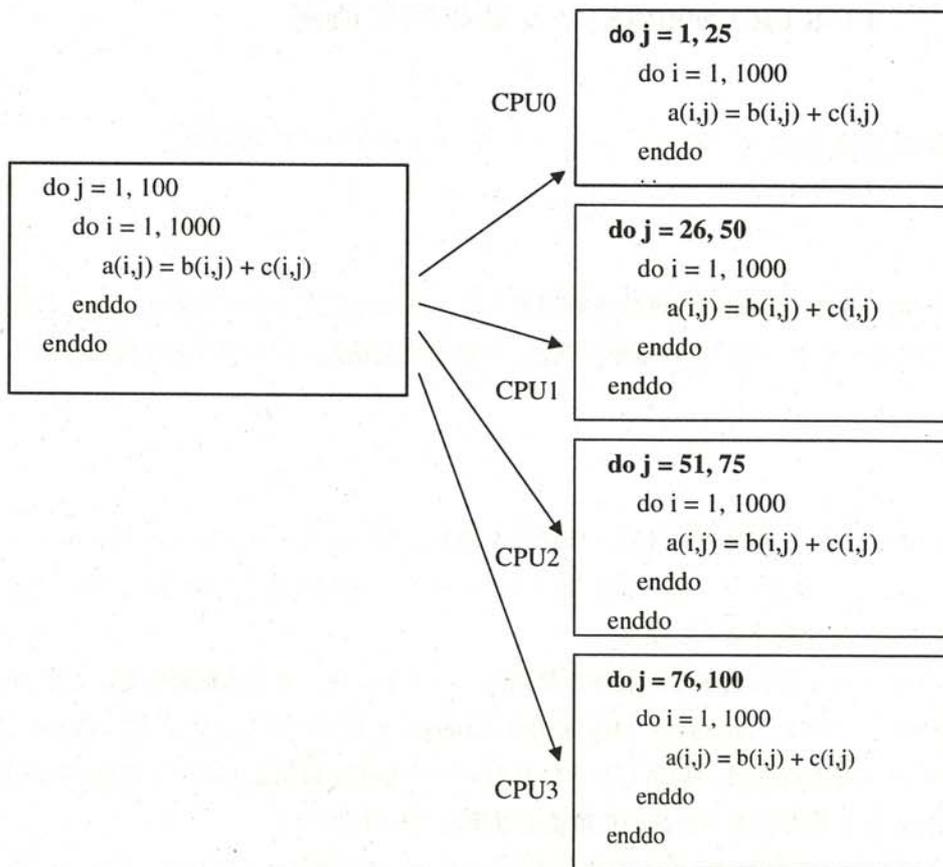


図1 並列実行のイメージ

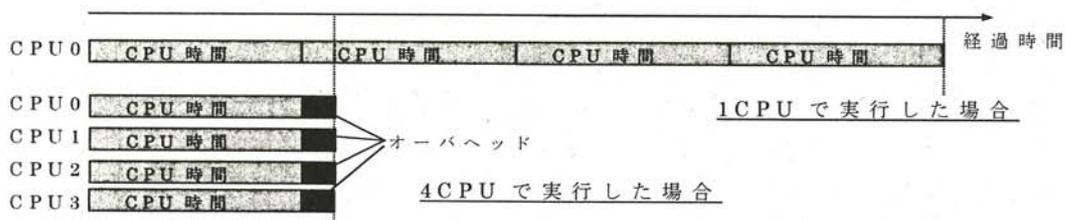
この例の配列 a は分割されて、各タスク毎に値 (a(i,1)~a(i,25)、 a(i,26)~a(i,50)、 a(i,51)~a(i,75)、 a(i,76)~a(i,100)) が計算され、定義されるので、a は各タスクから共通に参照できるグローバルな領域に割り当てられます。このような各タスクから共通に参照できるデータをタスク間共有データと呼びます。これに対して、並列実行される各タスクから、非同期に定義/参照を行うと、結果が不正になってしまうようなデータ (上例では i) は、各タスク毎にローカルな領域 (スタック) に割り当てられ、タスク固有なデータと呼びます。自動並列化機能は、このようなデータの割り当ても適切に行います。

この自動並列化機能は、オプション“-P auto”を指定するだけで利用可能です。

```
sxf90 -P auto program.f
```

ここで、“実行時間の短縮”には注意が必要です。並列処理は、1つの仕事を分割して、並列に実行を行うわけですから、CPU 時間が削減されるわけではなく、経過時間が短縮されることになります。また、仕事を各タスクで並列実行させるための処理 (オーバーヘッド) も必要となり、CPU 時間は、かえって増加することになります。たとえば、CPU 時間と経過時間の関係は、以下のようになります。

図2 並列処理における CPU 時間と経過時間



2.1 並列処理とベクトル処理

ここで、“ベクトル化による実行時間の短縮”と“並列化による実行時間の短縮”との相違を明確にしたいと思います。ベクトル処理とは、規則的に並んだ複数個の配列データを一度に演算する高速なベクトル命令を使って処理を行うことであり、この場合、CPU 時間が短縮され、同時に経過時間も短縮されます。これに対して、並列処理では、先に述べた通り、合計の CPU 時間は並列化のオーバーヘッドにより、単一 CPU で実行した時よりも増加することになりますが、経過時間を短縮することによって高速化を図ります。したがって、ベクトル化の場合は、単一 CPU での実行ですが、上手にベクトル化できれば、スカラで実行した時よりも、一般的に 10 倍以上の性能向上が期待できます。並列化の場合に期待できる性能向上の効果は、最大で使用可能な CPU の個数倍となります。

これらのことより、基本的には、ベクトル化と並列化を組み合わせる利用し、多重ループの内側ループについてはベクトル化を行い、外側ループを並列化することが、高速化を図る最善の方法となります。

また、並列処理した場合には、オーバーヘッド時間が加わりますので、並列に実行される仕事量（粒度と呼びます）が十分に大きくなければ、並列化の効果は期待できません。当然ですが、並列処理のオーバーヘッド時間よりも並列実行される部分の実行時間の方が小さければ、並列化したことにより、実行時間（経過時間）がかえって多くなってしまふこととなります。

ベクトル化できるプログラムは、ベクトル化すればほとんど全ての場合に性能向上が図れますが、並列化できるプログラムは、並列化したからといって必ずしも性能が向上するとは限らないこととなります。すなわち、どんなプログラムでも自動並列化すれば性能が向上するというわけではないことに注意して下さい。以下では効果的な並列化の方法について説明をしていきたいと思ひます。

3. 自動並列化

自動並列化機能を使用すれば、その名の通り、並列用の指示行を直接使って並列プログラミングをする場合に比べ、格段に容易にプログラムを並列化することができます。自動並列化機能は、プログラムを解析し、並列化した場合の効果も調べて、並列化を行います。たとえば、並列化すれば十分性能が向上するだけの粒度をもっているか、do ループの繰り返しを並列実行しても結果不正になるような文を含んではいないかなどを調査し、可能な場合は、並列化できるようにループやデータの定義を書き直します。

自動並列化は、内部的に図 3 のようなイメージで行われます。

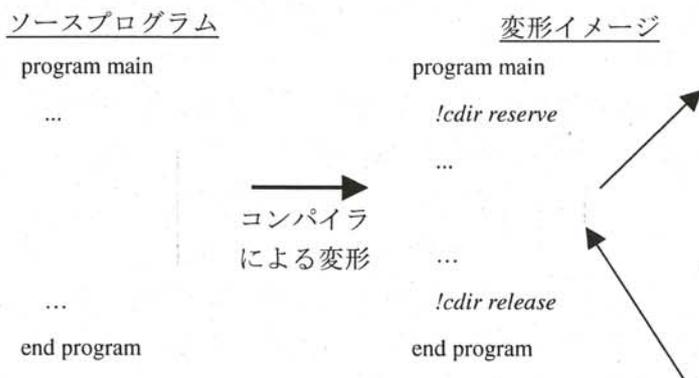


図 3 自動並列化におけるコンパイラの変形イメージ

コンパイラは並列可能なループを検索し、そのループを新たにサブルーチンとして切り出し、マイクロタスク機能を使って並列化します。並列実行されるループをサブルーチンとして切り出すことによって、並列化効率を最大限に引き出すことが可能となります。上記例の!cdir で始まる行はコンパイラによって挿入された指示行で、reserve はタスクの確保、release はタスクの解放、pardo はループを並列実行することを指示します。また切り出されたサブルーチン名には、元のサブルーチン名に\$1、\$2、...とサフィックスが付きます。コンパイル時に、オプション-R1 を指定することによって、並列化された様子を示すリスト（変形リスト）を参照することができます。

3.1 自動並列化の条件

自動並列化の条件は、以下の通りです。

対象となる構文	DO ループ 配列式
対象となるデータの型	構造型、文字型以外の型(4 倍精度も可能)
対象となるループ中に許される文	代入文、IF 文、GOTO 文、CONTINUE 文、CALL 文、SELECT 構文
対象となる演算	加減乗除算、べき算、論理演算、関係演算、型変換、組込み関数

- ループ中のデータに依存関係がある場合に、ベクトル化や並列化ができなくなることがありますが、その条件には違いがあります。次の例では、ベクトル化は可能ですが、並列化はできません。配列 a の引用（演算）される要素 (a(i)) と定義される要素 (a(i-1)) がループの繰り返し間でオーバーラップしているため、ループの繰り返しを並列実行すると、結果不正になる可能性があります。

```
do i = 2, n
  a(i-1) = a(i) * b(i) + c(i)
enddo
```

並列化の場合は、ループの繰り返し間でデータの依存関係があると、並列化できなくなるため、ベクトル化よりも条件が厳しくなります。

- call がループ中にある場合、ベクトル化はできませんが、呼び出すサブルーチンが並列実行可能であれば、そのループを並列化することができます。
- ベクトル化では、4 倍精度用のベクトル命令がないため、4 倍精度のベクトル化は不可能でしたが、並列化においては、ハード的な制約はないため、並列化が可能です。

3.2 自動並列化方法

並列実行される場合は、先にも述べた通り、粒度が十分に大きくなければその効果が期待できません。また、SX-8R はベクトルマシンであるため、ベクトル化が性能向上には欠かせない要因となります。これらのことを考慮し、自動並列化機能は、基本的には、多重ループの内側ループをベクトル化し、外側ループを並列化します。内側ループがベクトル化できない場合は、スカラコードのまま外側ループが並列化されます。

```

subroutine sub(a,h)
  real a(600,100,100)
  integer h(100)

```



```

end

```

さらに、コンパイラは、並列化の効果を高めるため、可能であればループ変形などの最適化を行い、並列化を促進します。これらの並列化の工夫について、いくつか例を紹介します。

◆ 一重ループの場合

基本的には、ベクトル化を行います。ループのコストが大きい、つまり粒度が大きい場合は、ループを分割して、ベクトル化+並列化を行います。ベクトル化できない場合は、並列化だけが行われます。

```

do i= 1,100
  a(i) = sqrt(b(i)**2 + c(i)**2)
enddo

```

ベクトル化

```

do i= 1,10000
  a(i) = sqrt(b(i)**2 + c(i)**2)
enddo

```

ベクトル化+並列化

◆ ループ融合や一重化が可能なループの場合

ループ融合や一重化などのループの最適化を行った後に、並列化を行います。

```

subroutine sub(a,b,c)
  real a(10000,4),b(10000,4),c(10000,4)
  do j= 1,4
    do i=1,10000
      a(i,j) = sqrt(b(i,j))
    enddo
  enddo
  do j= 1,4
    do i=1,10000
      b(i,j) = c(i,j) - a(i,j)
    enddo
  enddo
  return

```

→
ループ融合
+
一重化

```

do j=1,10000*4
  a(j,1) = sqrt(b(j,1))
  b(j,1) = c(j,1) - a(j,1)
enddo

```

並列化

◆ 条件並列化

ループ長（粒度）あるいは依存関係が不明で、並列化の効果がコンパイル時に判断できない場合、実行時に粒度や依存関係を調べて並列コードを実行するかどうかを選択できるように条件並列化を行います。

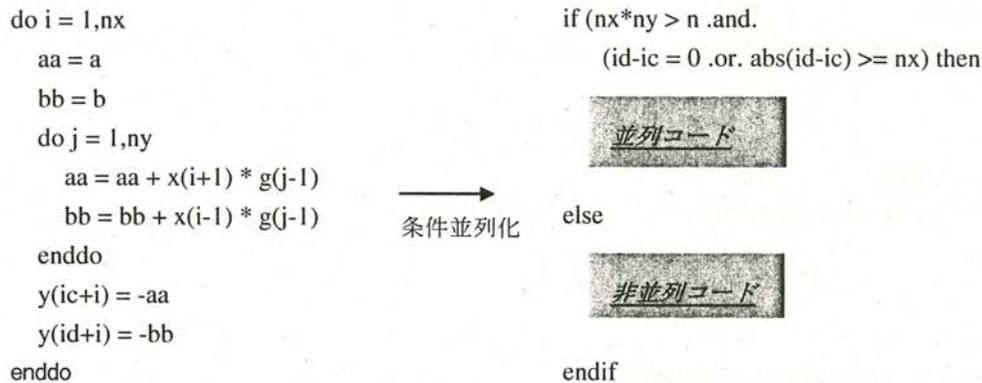
次の例では、 $nx*ny > n$ によって、粒度が並列化するのに十分かどうかを調べています。これは、ループの繰り返し数が十分に大きいかどうかを実行時にチェックしているわけですが、このとき、コンパイラは単純に繰り返し数だけではなく、do ループ中の各演算（加算や乗算など）に対して重み付けを行い、do ループの演算コストから並列化の効果が十分に期待できる値（n）を計算して、条件並列化を行います。

また、ループ中のデータに繰り返しにまたがった依存関係がある場合は並列化すると正しい結果が得られませんが、この例では、配列 y を定義している $y(ic+i)$ と $y(id+i)$ の添字に現れる変数 ic と id において

$id=ic$ または id と ic の値の差が nx 以上

という条件が満たされていれば、ループの実行中に $y(ic+i)$ と $y(id+i)$ により値が変更される領域に重なりがないことが保証できます。

この条件 ($id-ic=0$.or. $abs(id-ic) \geq nx$) を実行時に調べ、並列実行可能な場合のみ並列コードを実行するようにしています。



◆ 手続き呼び出しを含むループの場合

ループ内に手続き呼び出しがある場合、その手続き呼び出しにより、ループの繰り返しのまたがるデータの依存関係が生じていなければ、後で説明する CNCALL 並列化指示行を指定するか、手続きをインライン展開することによって並列化を行うことができます。以下に示す例の2つのサブルーチン abc, def 中のループは、展開イメージからわかるように、データに依存関係が存在しないため、並列化が可能です。

```

real x(100,100),y(100,100),z(100,100)
do i = 1,100
  call abc(x(1,i),y(1,i),100)
  call def(y(1,i),z(1,i),100)
enddo
end

```

インライン展開

```

subroutine abc(a,b,n)
real a(n), b(n)
do i = 1,n
  b(i) = a(i) + 1.0 / a(i)
enddo
end

```

```

subroutine def(a,b,n)
real a(n), b(n)
do i = 1,n
  b(i) = b(i) + sqrt(a(i))
enddo
end

```

```

do i = 1,100
!サブルーチン abc を呼び出したイメー
ジ
  do i1 = 1,100
    y(i1,i) = x(i1,i) + 1.0 / x(i1,i)
  enddo
!サブルーチン def を呼び出したイメー
ジ
  do i2 = 1,100
    z(i2,i) = z(i2,i) + sqrt(y(i2,i))
  enddo
↓ 並列化イメージ

```

```

do i = 1,100
  do i1 = 1,100
    y(i1,i) = x(i1,i) + 1.0 / x(i1,i)
  enddo
  do i2 = 1,100
    z(i2,i) = z(i2,i) + sqrt(y(i2,i))
  enddo
enddo

```

並列化

◆ 配列構文の場合

配列構文の場合は、内部的には、do ループのイメージに展開され、その後で、ベクトル化、並列化が行われます。

```
real a(999, 1000), b(999, 1000), c(999, 1000)
```

```

a = b * c
b = sin(c)

```

イメージ

```

do j = 1, 1000
  do i = 1, 999
    a(i, j) = b(i, j) * c(i, j)
    b(i, j) = sin(c(i, j))
  enddo
enddo

```

並列化
ベクトル化

4. 並列化の阻害要因と並列化指示行

4.1 並列化の阻害要因

先にも述べた通り、ループの繰り返し間にデータの依存関係がある場合は、並列化はできません。並列化を妨げる要因をいくつか紹介します。

◆ 添字に重なりがある場合

```

do i = 1,n
  a(i) = b(i+1)
  b(i) = c(i)
enddo

```

この例も3節で述べた例と同様に、ループの繰り返し間にデータの依存関係があるために並列化

ができない例です。ただし、ベクトル化は可能です。依存関係によるベクトル化可/並列化不可の理由をもう少し具体的に説明します。

ベクトル化の場合は、ループの繰り返しの実行順序は保証されますが、並列化の場合は、ループの実行順序は保証されません。上記例においてループの繰り返しと配列 b の定義/参照関係に着目すると以下ようになります。この例では、ベクトル化の場合は、参照と定義の順番は保証され、たとえば、b(3)の値は必ず参照してから定義されることとなります。

ループの繰り返し	参照	定義
1	b(2)	b(1)
2	b(3)	b(2)
3	b(4)	b(3)

すなわち、ループの繰り返しの繰り返しの定義/参照関係は、プログラム通りの正しい関係が保持されることとなります。次に並列化の場合ですが、簡単のために、ループの繰り返し 2 回毎に並列化する場合を例に考えてみます。

ループの繰り返し	参照	定義	
1	b(2)	b(1)	
2	b(3)	b(2)	タスク 1 で実行
3	b(4)	b(3)	
4	b(5)	b(4)	タスク 2 で実行
...

この場合は、タスク 1、タスク 2、... が並列に実行されることになるため、b(3)の値がタスク 1 で参照されるタイミングとタスク 2 で定義されるタイミングの順序は保証できません。すなわち、先にタスク 2 で定義された値 (c(3)の値) をタスク 1 で参照して、a(2)に代入してしまう可能性があるわけです。

◆ 定義と引用が閉じていない場合

```
do i = 1,n
  c(i) = t
  t = b(i)
enddo
```

ループ中で、変数 t を引用してから、定義しているため、ループの繰り返しの並列実行すると、結果不正となります。文の意味は変わってしまいますが、次のように変数 t を定義してから、引用していれば、並列化が可能です。

```
do i = 1,n
  t = c(i)
  ...
  b(i) = t
enddo
```

◆ if 文下に do 変数以外のインデックス変数がある場合

```
do j = 1,n
  do i = 1,n
    if (a(i,j) >= del) then
      ii = ii + 1
      ic(ii,j) = ii
    endif
  enddo
do i = 1,ii
  b(i,j) = ic(i,j) + sin(c(ii,j))
enddo
enddo
```

ii の更新 (if 文 then 節の実行) が a(i,j) の値に左右され、ii は外側ループの繰り返しのにおいても加算されていきますが、外側ループで並列化された場合、各タスクで並列実行されるループ毎に ii が加算されてしまい、ii の値が正しく計算されなくなり、b の結果が不正となります。

◆ ループからの飛び出しがある場合

```
do j = 1,n
  do i = 1,n
    a(i,j) = sqrt(b(i,j))
    if (a(i,j) >= del) go to 100
    if (c(i,j) >= 0) then
      b(i,j) = c(i,j) - a(i,j)
    else
      b(i,j) = c(i,j) + a(i,j)
    endif
  enddo
enddo
100 continue
```

並列実行されると、ループの繰り返しの実行順序が保証されないため、ループから飛び出すタイミングがプログラム通りにならない場合があります、配列 b の値が不正になる可能性があります。

4.2 指示行の利用とソースプログラムの書き換えによる並列化促進

並列化の阻害要因は、上記以外にもありますが、比較的発生しやすい状況は、依存関係によって並列化が妨げられる場合や、各タスクで同じデータに書き込みを行ってしまうことによって結果不正を引き起こしてしまうような場合です。これらは、プログラミング上の工夫によって、回避することが可能な場合も多々あります。また、ベクトル化の場合と同じように、プログラマには、依存関係がないことがわかっており、それをコンパイラに教えてやることによって並列化が可能になる場合もあります。このような状況に対応できるように、FORTRAN90/SX コンパイラには、指示行が用意されています。本節では、指示行の紹介と利用法、およびプログラミング上の工夫の例を紹介します。

(1) 並列化指示行

並列化指示行は、カラム 1 から

!CDIR オプション

の形式で指定します。並列化用の主な指示行のオプションとその利用法は、次の通りです。

◆ CONCUR / NOCONCUR

直後のループを自動並列化の対象とする/しないを指定します。

たとえば、並列化するとかえって性能が劣化するループをプログラムが含んでいる場合に、そのループの先頭に NOCONCUR を指定します。

◆ INNER / NOINNER

最内側ループあるいは一重ループを自動並列化の対象とする/しないを指定します。

最内側ループは、既定値では自動並列化の対象とはならないため、最内側ループを並列化できると効果がある場合に、INNER を指定します。また、一重ループの場合も、並列化効果がコンパイル時に不明な場合は、並列化の対象とはなりません。INNER を指定することによって並列化をすることが可能となります。

!CDIR INNER

```
do i = 1,n
  a(i) = a(i)+b(i)*c(i)
enddo
```

→
イメージ

```
if (n > 1000) then
  ベクトル+並列コード
else
  ベクトルコード
endif
```

この例では、コンパイラは、ループ中のコストを計算して、条件並列化を行っています。

◆ NOSYNC

ループ中の配列要素に重なりがないことを指定します。

次の例で、k1 と k2 の値がコンパイル時に不明な場合は、k1=k2 であった場合に結果不正になる可能性があるため、このループを自動並列化することはできません。k1 と k2 の値が同じでないことがわかっている場合は、NOSYNC を指定することによって並列化が可能となります。

!CDIR NOSYNC

```
do j = 1,ny
  do i = 1,nx
    a(i,k1,j+1) = a(i,k2,j) + b(i)
  enddo
enddo
```

並列化

◆ SELECT(CONCUR)

多重ループにおいて、指定されたループを優先して並列化します。

多重ループにおいて、並列化した場合に最も効率のよいループを指定することができます。たとえば、最外側ループの繰り返し回数が 1 や 2 である場合、最外側ループで並列化しても、その効果は期待できません。このような場合は、次の例のように、SELECT(CONCUR)を指

定することによって、より効率よく並列化することが可能となります。

```
do k = 1,nz
!CDIR SELECT(CONCUR)
do j = 1, ny
do i = 1,nx
c(i) = b(i,j,k) / dble(nx)
a(i,j,k) = a(i,j,k) + c(i) / 2.0
enddo
enddo
enddo
```

イメージ

```
do k = 1,nz
do j = 1, ny
do i = 1,nx
c(i) = b(i,j,k) / dble(nx)
a(i,j,k) = a(i,j,k) + c(i) / 2.0
enddo
enddo
enddo
```

並列化

◆ CNCALL

手続き呼び出しを含むループを並列化してもよいこと指定します。

あらかじめ、ループ中で呼び出しているサブルーチンが並列実行されても問題がないことがわかっている場合、CNCALL を指定することによって並列化が可能となります。自動並列化機能では、引数として渡す変数が、ループ中で明示的に定義されていない（ループ中で値が代入されない）場合は、タスク間共有データ（タスク間での共通の領域を参照するデータ）となるため、渡されたデータがサブルーチン中で更新されないということが、並列実行しても結果不正にならない重要な条件となります。

!CDIR CNCALL

```
do i = 1,n
call sub(a(i), x)
enddo
```

並列化

たとえば、この例の場合、x はタスク間共有データであるため、複数のタスクでサブルーチン sub が実行され、それぞれで x が更新されると、結果不正を引き起こすことになります。したがって、x が sub 中で変更されない場合のみ、CNCALL 指示行を指定して並列化することが可能です。

◆ 強制並列化指示行

上記指示行等を指定してもコンパイラにそのループが並列化可能であることが認識できず並列化が行なわれない場合でも、利用者には、そのループが並列化可能であることがわかっている場合があります。このような場合にコンパイラに強制的に並列化を行なわせることを指定する指示行が、強制並列化指示行です。

ただし、この指示行を指定した場合、コンパイラはデータの依存関係などのチェックは行なわずに並列化をするので、並列化したときの動作の妥当性については利用者が保証しなければなりません。

!CDIR PARALLEL DO PRIVATE(wk)

```
do j = 1, 10
do i = 1, 100
wk(i) = a(i) + b(i)
enddo
call sub(a(i), wk)
enddo
```

並列化

PARALLEL DO 指示行は直後のループを並列実行することを指定します。各ループの繰り返しで作業用に使用している配列 (wk) のようにタスク固有のデータは、PRIVATE オプションで指定します。ただし、do 変数はコンパイラが自動的にタスク固有データと解釈するので、明示的に PRIVATE で指定する必要はありません。PRIVATE に指定されなかったデータはタスク間共有データとなります。

```
!CDIR PARALLEL DO
do i = 1, 100
  call sub(a(i), b(i), x)
!CDIR ATOMIC
  s = s + a(i) * b(i)
enddo
```

並列化

ATOMIC 指示行は、PARALLEL DO 指示行で並列化されたループ中の総和や内積など、複数のタスクが同時に実行してはならない代入文に指定します。

(2) プログラミング上の工夫による並列化促進

◆ ループの入れ換えによる並列化促進

次のループは外側ループで並列化すると、 $a(i,j)$ と $a(i,j-1)$ の間に依存関係があり、正しい実行結果が得られません。この場合は、ループを入れ換えることによって並列化が可能となります。

```
do j = 2,m
  do i = 1,n
    a(i,j) = a(i,j-1) * b(i,j)
  enddo
enddo
```

→

```
do i = 1,n
  do j = 2,m
    a(i,j) = a(i,j-1) * b(i,j)
  enddo
enddo
```

ループ入れ換え
並列化可能

◆ 仮引数の配列サイズ変更による並列化促進

引数として渡ってきたデータは、タスク間で共有となるため、次の例では、配列 c がタスク間共有変数となります。このため、最外側ループ (do k=1,nz) で並列化を行うと、配列 c の領域を各タスクで書き換えることになり、結果不正となってしまいます。

そこで、配列 c の次元の宣言を変更し、最外側ループで異なる領域を使用することにすれば、並列化が可能となります。もちろん、この sub を呼び出しているサブルーチン中の対応する配列に対しても修正が必要です。

```

subroutine sub(a,b,c,nx,ny,nz)
real*8 a(100,100,100),b(0:100,100,100)
real*8 c(0:100)
do k = 1,nz
  do j = 1, ny
    do i = 1,nx
      c(i) = b(i,j,k) / dble(nx)
    enddo
    do i = 1,nx
      a(i,j,k) = a(i,j,k) + (c(i-1)+c(i))
                                     / 2.0
    enddo
  enddo
enddo
end

```



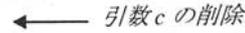
```

subroutine sub(a,b,c,nx,ny,nz)
real*8 a(100,100,100),b(0:100,100,100)
real*8 c(0:100,100)
do k = 1,nz
  do j = 1, ny
    do i = 1,nx
      c(i,k) = b(i,j,k) / dble(nx)
    enddo
    do i = 1,nx
      a(i,j,k) = a(i,j,k) + (c(i-1,k)+c(i,k))
                                     / 2.0
    enddo
  enddo
enddo
end

```

◆ 作業領域の受け渡しをしないことによる並列化促進

上の例において、サブルーチン sub を呼び出している側が、配列 c を参照していない、すなわち配列 c を単なる作業領域として確保している場合、配列 c を引数として渡さずに、サブルーチン sub 側で配列宣言することによって、並列化が可能となります。



```

subroutine sub(a,b,c,nx,ny,nz)
real*8 a(100,100,100),b(0:100,100,100)
real*8 c(0:100)
do k = 1,nz
  do j = 1, ny
    do i = 1,nx
      c(i) = b(i,j,k) / dble(nx)
    enddo
    do i = 1,nx
      a(i,j,k) = a(i,j,k) + (c(i-1)+c(i))/2.0
    enddo
  enddo
enddo
end

```

5. 並列プログラムの性能解析ツール

並列プログラムの実行性能を調べる機能として、プログラム特性情報(proginf 情報)が利用できます。

proginf 情報は、プログラムの実行時に環境変数 F_PROGINF を YES あるいは DETAIL と設定することによって採取できます。以下に、並列プログラム実行時の proginf 情報の出力例を示します。なお、図中の※は、並列処理使用時に表示される情報です。

```

***** Program Information *****
Real Time (sec)      :      2241.800319   経過時間
User Time (sec)     :      8903.817840   ユーザCPU時間
Sys Time (sec)      :           0.848539   システムCPU時間
Vector Time (sec)   :      8713.069899   ベクトル命令実行時間
Inst. Count         :      716897080005.  全命令実行数
V. Inst. Count      :      331530260017.  ベクトル命令実行数
V. Element Count    :      81369094603294.  ベクトル命令処理要素数
FLOP Count          :      35064202883826.  浮動小数点データ処理要素数
MOPS                :           9181.955751  1秒あたりの実行演算数
MFLOPS              :           3938.108743  1秒あたりの浮動小数点データ処理要素数
MOPS (concurrent)   :           36469.877067  1秒あたりの実行演算数(実効時間) ※
MFLOPS (concurrent) :           15641.802862  1秒あたりの浮動小数点データ処理要素数(実効時間) ※
VLEN                :           245.434895   平均ベクトル長
V. Op. Ratio (%)    :           99.528629   ベクトル演算率
Memory Size (MB)    :           5440.000000   メモリサイズ
Max Concurrent Proc. :                4.      実行プロセッサ数 ※
  Conc. Time(>= 1)(sec):           2241.698300  少なくとも1台のCPUが同時に(並列に)動いた時間 ※
  Conc. Time(>= 2)(sec):           2220.965807  少なくとも2台のCPUが同時に(並列に)動いた時間 ※
  Conc. Time(>= 3)(sec):           2220.934612  少なくとも3台のCPUが同時に(並列に)動いた時間 ※
  Conc. Time(>= 4)(sec):           2220.223084  少なくとも4台のCPUが同時に(並列に)動いた時間 ※
Event Busy Count    :                0.      event待ち回数(マクロタスク用) ※
Event Wait (sec)    :           0.000000    event待ち時間(マクロタスク用) ※
Lock Busy Count     :                0.      lock待ち回数(マクロタスク用) ※
Lock Wait (sec)     :           0.000000    lock待ち時間(マクロタスク用) ※
Barrier Busy Count  :                0.      barrier待ち回数(マクロタスク用) ※
Barrier Wait (sec)  :           0.000000    barrier待ち時間(マクロタスク用) ※
MIPS                :           80.515695   1秒当たりの命令実行数
MIPS (concurrent)   :           319.800876   1秒当たりの命令実行数(実効時間) ※
I-Cache (sec)       :           0.239609   命令キャッシュミス時間
O-Cache (sec)       :           5.363951   データキャッシュミス時間
Bank (sec)          :           9.214590   バンクコンフリクト時間

Start Time (date)   : 2002/08/21 21:21:08   開始時刻(日付)
End Time (date)     : 2002/08/21 21:58:29   終了時刻(日付)

```

図4 並列処理使用時の proginf 情報

並列処理の性能を分析するときに重要な情報は、proginf 情報の Conc. Time です。図4の例で網掛けされた Conc. Time の部分を見ると、このプログラムが4つのCPUで約2200秒づつ実行されたことが分かります。

この例では、Conc. Time がほぼ均等となっているので、うまく並列化されていると言えます。これが、以下の例

```

Conc. Time(>= 1)(sec):      74.154168
Conc. Time(>= 2)(sec):      8.549322
Conc. Time(>= 3)(sec):      8.292376
Conc. Time(>= 4)(sec):      8.071275

```

のように、Conc. Time(>= 1)だけ実行時間が大きく残りが小さい場合は、プログラムが十分並列化できていないと言えます。このようなプログラムは、4.2で紹介した方法などを用いて、より多くのループが並列化されるようにする必要があります。

6. あとがき

以上、FORTRAN90/SX で利用可能な自動並列化機能について、簡単に説明いたしました。

並列化は、ベクトル化と異なり、すべてを自動にまかせにしても、必ず性能が向上するというわけにはいきません。したがって、SX の並列処理方式を理解した上で、よりよい並列処理が行われるようにプログラムのチューニングを行うことが、性能向上には大変重要です。

本稿が、自動並列化利用の手助けになれば幸いです。

参考文献

- [1] FORTAN90/SX プログラミングの手引き 日本電気 G1AF07
- [2] FORTAN90/SX 並列処理機能利用の手引き 日本電気 G1AF08

FORTRAN90/SX の自動ベクトル化機能

日本電気株式会社 第一コンピュータソフトウェア事業部

概要

SX-8R は豊富なベクトル演算命令を有しており、そのハードウェア性能を十分に引き出すためには、コンパイラの自動ベクトル化機能によりプログラムのベクトル化を行い、ベクトル命令によって処理される時間の割合(ベクトル化率)をできるだけ高くする必要がある。本文では、SX-8R の FORTRAN90/SX コンパイラがもつ自動ベクトル化機能について、その特長と性能向上の観点を紹介する。

1. はじめに

SX-8R は 8 個の CPU をもつノードを複数結合した「スケーラブル・パラレル・スーパーコンピュータ」であり、そのハードウェア性能を十分に引き出すためには、

- ・複数の CPU を効率的に使用する、並列化

と共に、

- ・個々の CPU の中で効率的に計算を行う、ベクトル化
- が大変重要となります。

並列化につきましては別稿に譲り、本稿では、FORTRAN90/SX コンパイラがもつ自動ベクトル化機能についてご紹介致します。

2. 自動ベクトル化機能

SX-8R におけるベクトル化技法は、SX-5 の場合と基本的には同じですが、今回はスーパーコンピュータを初めて使われる方にもご理解いただけるよう、ベクトル化の基本概念からご紹介させていただきます。

2.1 ベクトル化の基本概念

通常の演算命令は、一度に一組のデータに対する演算処理を行います。(これを、ベクトル命令と対比させるためにスカラ命令と呼びます。)

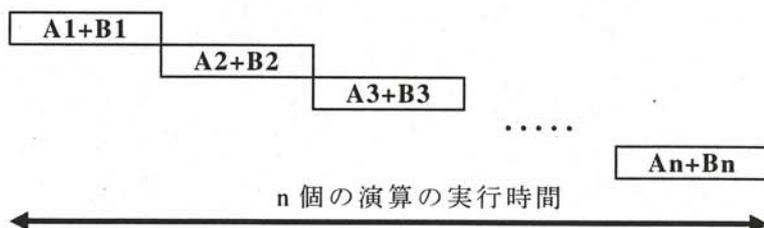
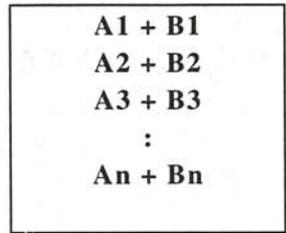


図 1 スカラ命令(スカラ加算)の実行イメージ

これに対して、ベクトル命令は、複数の組のデータに対する演算処理を一つの命令で一度に行うことができます。




 ベクトル命令の実行時間

図2 ベクトル命令(ベクトル加算)の実行イメージ

ループ中で計算される行列の要素など、規則的に並んだ配列データ(ベクトルデータ)に対してベクトル命令を適用することを自動ベクトル化と呼び、ベクトル化することによって高速な演算が可能となります。

2.2 自動ベクトル化の例

例えば、次の DO ループは、2つの配列からのデータのロード、ベクトル加算、メモリへのストアの4つのベクトル命令で実行されます。

例1 配列代入文と、ベクトル化されて生成される命令の概念

```
DO I=1,100
  C(I) = A(I) + B(I)
END DO
```

↓

```
VR1 ← 配列 A (配列 A からベクトルレジスタにデータをロード)
VR2 ← 配列 B (配列 B からベクトルレジスタにデータをロード)
VR3 ← VR1 + VR2 (ベクトル加算)
配列 C ← VR3 (配列 C に結果をストア)
VRn : ベクトルレジスタ
```

2.3 ベクトル化の対象範囲

FORTRAN90/SX コンパイラは、Fortran プログラムの以下の範囲を対象として自動ベクトル化を行います。

ベクトル化の対象となるループ	配列式、DO ループ、DO WHILE ループ、FORALL ループ、IF 文と GOTO 文によるループ
ベクトル化の対象となる文	代入文、CONTINUE 文、GOTO 文、CYCLE 文、EXIT 文、IF 文、SELECT 構文 (CALL 文、入出力文 等は不可)
ベクトル化の対象となるデータの型	4 バイト/8 バイトの整数型・論理型 単精度/倍精度の実数型・複素数型 (文字型、4 倍精度、2 バイト整数型 等は不可)
ベクトル化の対象となる演算	加減乗除算、べき算、論理演算、関係演算、型変換、組込み関数 (利用者定義演算、ポインタ代入 等は不可)

詳細につきましては、「FORTRAN90/SX プログラミングの手引 5.1 ベクトル化の条件およびベクトル化の例」を参照してください。

2.4 ベクトル化可能な条件

しかし、上記の範囲内であれば、どんなプログラムでもベクトル化できる訳ではありません。次の例を見てください。

```
例 2   DO I=1,99
        A(I) = 2.0      ! 代入文 2-1
        B(I) = A(I+1)   ! 代入文 2-2
      END DO
```

図 3 に、このプログラムをベクトル化しない場合、ベクトル化した場合の演算の実行順序を示します。

ベクトル化しない場合 の実行順序	ベクトル化した場合 の実行順序
A(1)=2.0 ! 代入文 2-1	A(1)=2.0 ! 代入文 2-1
B(1)=A(2) ! 代入文 2-2	:
A(2)=2.0 ! 代入文 2-1	A(2)=2.0 ! 代入文 2-1
B(2)=A(3) ! 代入文 2-2	B(1)=A(2) ! 代入文 2-2
:	:
A(99)=2.0 ! 代入文 2-1	A(99)=2.0 ! 代入文 2-1
B(99)=A(100)! 代入文 2-2	B(2)=A(3) ! 代入文 2-2
	:
	B(99)=A(100)! 代入文 2-2

図 3 ベクトル化の有無による実行順序の相違

例 2 のループをベクトル化して実行すると、全ての要素に対して 2-1 の代入がベクトル命令で実行され、次に 2-2 の代入がやはり全ての要素に対して実行されます。このため、配列 B(1)~B(98) の値は全て 2.0 となり、プログラムの意図と異なる結果となってしまいます。

また、N 回目の繰り返しで定義したスカラ変数を N+1 回目の繰り返しで引用する、例 3 のようなプログラムも、全ての要素に対して 3-1 の代入が先に実行されるため、ベクトル化すると配列 A(1)~A(100) の値が全てゼロとなってしまいます。

```
例 3   S=0.0
      DO I=1,100
        A(I) = S      ! 代入文 3-1 (スカラ変数 S の引用)
        S = B(I)+C(I) ! 代入文 3-2 (スカラ変数 S の定義)
      END DO
```

これらのプログラムのように、ベクトル化によって配列の定義・引用関係に変化が生じてしまう場合には、ベクトル化することができません。

FORTRAN90/SX の自動ベクトル化機能は、コンパイラがソースプログラムを解析して、ベクトル命令で実行できる部分を自動的に検出するとともに、必要ならベクトル化に適合するようにプログラムを変形して、その部分に対してベクトル命令を生成します。

2.5 Fortran95 配列構文のベクトル化

例 2 のプログラムは、FORTRAN77 と同じ DO ループでしたが、Fortran95 配列構文ではどうでしょうか。

```
例 4   A(1:99)=2.0      ! 代入文 2-1
        B(1:99)=A(2:100) ! 代入文 2-2
```

配列構文は DO ループと異なり、それぞれの文毎に全ての要素を演算することが、規格で定められています。即ち例 4 では、まず代入文 2-1 によって $A(1:99)=2.0$ を全て実行してから、代入文 2-2 を実行します。これはすなわち図 3 に示した「ベクトル化した場合の実行順序」と同じです。従って Fortran95 配列構文では、例 2 の様に複数の文の前後関係によってベクトル化が阻害されることはありません。(もちろん他の要因でベクトル化できないことはあります。) なお、Fortran95 配列構文については「FORTRAN90/SX 言語説明書」を参照してください。

2.6 ベクトル長

個々のベクトル命令は、演算処理に入る前にある程度の準備処理が必要となります。この準備処理に要する時間を立ち上がり時間と呼びます。即ち、実際のベクトル演算に要する時間が余りにも小さいと、立ち上がり時間の影響が大きくなり、ベクトル化による高速化が行えません。

ベクトル化した場合とベクトル化しない場合とで実行時間が等しくなるループの繰り返し数(ループ長)を交叉ループ長と呼び、立ち上がり時間と交叉ループ長には、図 4 に示す関係があります。

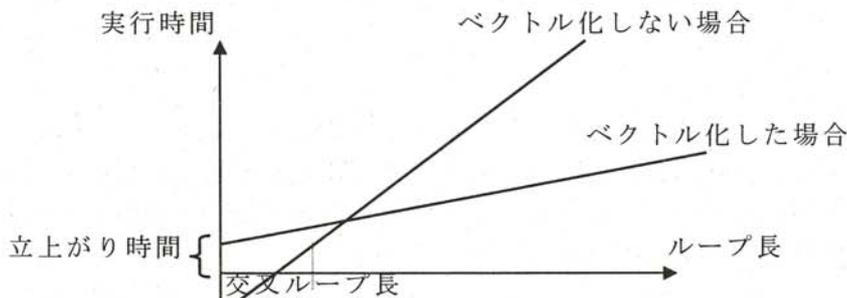


図 4 立ち上がり時間と交叉ループ長

この図を見ると、ループ長をできるだけ長くした方が、ベクトル化による高速化の効果が大きいことが、おわかりいただけると思います。

FORTTRAN90/SX コンパイラの自動ベクトル化では、ループの繰り返し回数が 5 未満の場合には、ベクトル化による効果が少ないと判断し、ベクトル化していません。

3. 拡張ベクトル化機能

前述の通り、FORTRAN90/SX の自動ベクトル化機能は、ベクトル命令で実行可能な部分を検出しますが、そのままではベクトル化できない場合にプログラムを変形してベクトル化したり、プログラムを変形することによってベクトル化の効果をさらに高めたりします。これを拡張ベクトル化機能と呼びます。ここでは、FORTRAN90/SX が持つ種々の拡張ベクトル化機能のうち主なものを紹介します。

3.1 文の入れ換え

先ほどの例 2 は、「ベクトル化できない」と書きましたが、二つの文を入れ換えるとベクトル化できることに気づかれたでしょうか？

例 5 ソースプログラム

```
DO I=1,99
  A(I)=2.0      ! 代入文 5-1
  B(I)=A(I+1)  ! 代入文 5-2
END DO
```

コンパイラによる変形

```
DO I=1,99
  B(I)=A(I+1)  ! 代入文 5-2
  A(I)=2.0     ! 代入文 5-1
END DO
```

このプログラムを右のように変形すると、ベクトル化した場合でも配列の定義・引用関係が保存されますので、FORTRAN90/SX コンパイラは自動的に 5-1 と 5-2 の文を入れ換えてベクトル化を行います。

3.2 ループの一重化

ループ長が長いほどベクトル化の効果が高いとご説明しましたが、例 6 のように多次元配列を 1 次元配列と見なすことができる場合には、多重ループを一重ループに変形することでループ長を拡大すると同時に、外側ループの繰り返し制御の時間を削減します。

例 6 DIMENSION A(M,N), B(M,N), C(M,N)

```
DO I=1,M
  DO J=1,N
    A(J,I) = B(J,I) + C(J,I)
  END DO
END DO
```

↓ コンパイラによる変形のイメージ

```
DO IJ=1,M*N
  A(IJ,1) = B(IJ,1) + C(IJ,1)
END DO
```

3.3 ループの入れ換え

多重ループの場合には、通常は最も内側のループをベクトル化します。しかし、ループを入れ換えることにより定義・引用関係の矛盾が解消されてベクトル化できるようになったり、内側のループよりも外側のループの方がループ長が長く、入れ換えた方が速いと判断した場合には、コンパイラがループを入れ換えてベクトル化を行います。

例 7 配列 A に依存関係がありベクトル化できない

```
DO J=1,M           ! 外側ループ
  DO I=1,N         ! 内側ループ
    A(I+1,J) = A(I,J) + B(I,J)
  END DO
END DO
```

↓

依存関係がなくなりベクトル化できる

```
DO I=1,N           ! 元の内側ループ
  DO J=1,M         ! 元の外側ループ
    A(I+1,J) = A(I,J) + B(I,J)
  END DO
END DO
```

3.4 部分ベクトル化

ループ構造や配列式に、ベクトル化できる部分とベクトル化できない部分が含まれている場合、ベクトル化可能な部分と不可能な部分に分割し、可能な部分だけをベクトル化します。

例 8 4 倍精度の式が含まれているため、ベクトル化できない

```
REAL, DIMENSION (M) :: A, B, C
REAL(KIND=16), DIMENSION (M) :: Q
A(1:M) = B(1:M) * REAL(C(1:M) * Q(1:M))
```

↓

二つの配列式に分割することで、ベクトル化する。
作業用の配列 WK は、コンパイラが用意する。

```
WK(1:M) = REAL(C(1:M) * Q(1:M)) ! この配列代入文はベクトル化できない
A(1:M) = B(1:M)*WK(1:M)         ! この配列代入文をベクトル化する
```

3.5 条件ベクトル化

配列の依存関係がベクトル化に適合しているかどうかコンパイル時に不明であったり、ループ長がコンパイル時に不明で、ベクトル化したほうが速いかどうかわからない場合に、ベクトル化したコードとベクトル化しないコードの両方を生成しておき、プログラムを実行するとき、そのどちらかを選択して実行するものです。

例 9 K が 0 以上であるか、または K が -10 未満ならばベクトル化しても配列 A の各要素の定義参照順序が変わらないのでベクトル化できる

```
DO I=N, N+10
  A(I) = A(I+K) + B(I)
END DO
↓ コンパイラによる変形のイメージ
IF (K .GE. 0 .OR. K.LT. -10) THEN
  A(N:N+10) = A(N+K:N+10+K)+B(N:N+10) ! ベクトル化する
ELSE
  DO I=N, N+10 ! ベクトル化しない
    A(I) = A(I+K) + B(I)
  END DO
END IF
```

例 10 繰り返し回数がコンパイル時に不明なため、ベクトル化したほうが速いかどうか判断できない場合

```
A(1:N) = B(1:N) + C(1:N)
↓ コンパイラによる変形のイメージ
IF (N .GE. 5) THEN
  A(1:N) = B(1:N) + C(1:N) ! ベクトル化する
ELSE
  A(1:N) = B(1:N) + C(1:N) ! ベクトル化しない
END IF
```

3.6 マクロ演算の認識

次のようなパターンは、変数や配列要素が繰り返しにまたがって定義・引用されるため、本来はベクトル化できませんが、コンパイラが特別なパターンであることを認識し、専用のベクトル命令を用いることで、ベクトル化を行います。

例 11 総和: ひとつ前の繰り返しで定義したスカラ変数 S の値を、次の繰り返しで引用するため、通常はベクトル化できないが、配列 A の総和を求めるパターンであるとコンパイラが認識して、総和ベクトル命令を用いることでベクトル化する

```
DO I=1, N
  S = S + A(I)
END DO
```

例 12 漸化式: 配列 A のひとつ前の繰り返しで定義された値を次の繰り返しで参照するので、専用のベクトル命令を用いてベクトル化する

```
DO I=1, N
  A(I) = A(I-1) * B(I) + C(I)
END DO
```

備考:もし $A(I-1)=A(I)*B(I)+C(I)$ であれば、前の繰り返しで定義した値を後の繰り返しでは参照しないため、そのままベクトル化できる。

例 13 最大値、最小値を求める

```
DO I=1, N
  IF (XMAX .LT. X(I)) THEN
    XMAX = X(I)
  END IF
END DO
```

3.7 ループ融合

拡張ベクトル化機能ではありませんが、コンパイラは同じ形状(次元数と各次元のサイズ)を持つ複数の配列式、同じ繰り返し回数を持つ複数のループ構造を一つにまとめてベクトル化します。これをループ融合と呼びます。

例 14 次の二つの配列代入文は形状が一致しているので、下の二重ループと同じように解釈、最適化されてベクトル化されます。

```
A(1:M, 1:N) = B(1:M, 1:N) + C(1:M, 1:N)
D(1:M, 1:N) = E(1:M, 1:N) * F(1:M, 1:N) + S
      ↓      コンパイラによる変形のイメージ
DO J=1,N
  DO I=1,M
    A(I, J) = B(I, J) + C(I, J)
    D(I, J) = E(I, J) * F(I, J) + S
  END DO
END DO
```

コンパイラは、同じ形状の配列式・ループ構造が連続していれば融合しますが、間に形状の異なる配列式・ループ構造や、他の文があると融合できません。

高速化のためには、出来るだけ同じ形状の配列式・ループ構造を連続させるようにしてください。

例 15 二つの配列代入文の間に、形状の異なる配列代入文があるために、二つの配列式は融合されません。このような場合には、文の順序を入れ換えて、同じ形状の配列代入が連続するように書き換えてください。

下のよう書き換えることにより、ループ融合される

```
A(1:M, 1:N) = B(1:M, 1:N) + C(1:M, 1:N)
X(1:L) = 0.0
D(1:M, 1:N) = E(1:M, 1:N) * F(1:M, 1:N) + S
      ↓
A(1:M, 1:N) = B(1:M, 1:N) + C(1:M, 1:N)
D(1:M, 1:N) = E(1:M, 1:N) * F(1:M, 1:N) + S
X(1:L) = 0.0
```

4. ベクトル化率向上のための手法

4.1 ベクトル化率

プログラムをスカラ命令だけで実行させた場合の実行時間に占める、ベクトル命令で実行可能

な部分の時間の割合をベクトル化率と呼びます。

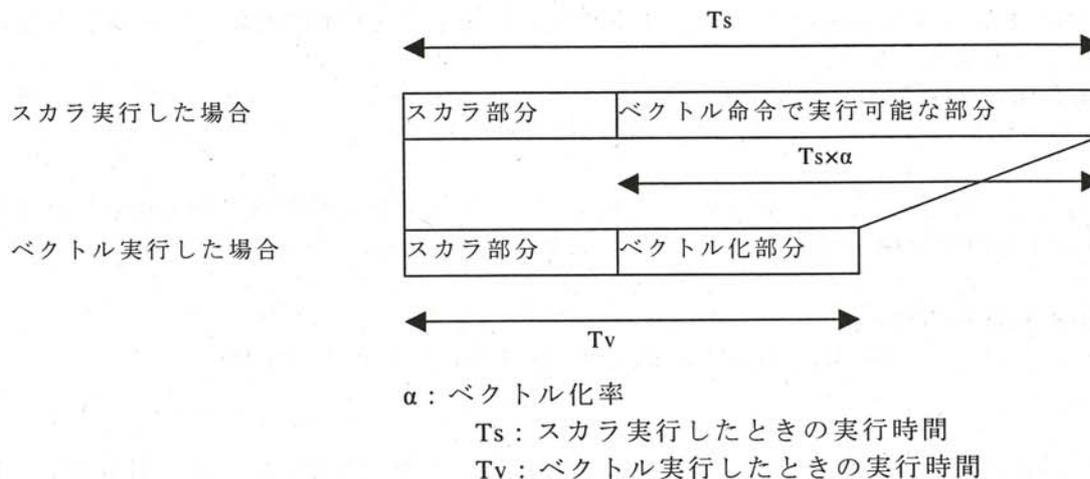


図5 ベクトル化率

プログラムをベクトル化することにより実行性能が向上しますが、プログラムの一部だけをベクトル化しても、ベクトル化の効果はあまり期待できません。ベクトル化部分の実行時間が非常に小さくても、ベクトル化率が50%程度では、高々2倍の性能にしかならないことは、図5からもわかると思います。一般にはベクトル化率が90~95%以上ないと、ベクトル化による大きな効果は期待できません。

すなわち、ベクトル化による高速化技法の一つは、ベクトル化率を高め、100%に近づけることにあります。

しかし、一般にベクトル化率を正確に求めることは困難であるため、SX-8Rでは、ベクトル化率に近い値として、プログラム特性情報(proginf)に表示されるベクトル演算率を用いています(5.1 proginf 情報参照)。ベクトル演算率は、実行された命令の数をハードウェアがカウントすることで、プログラムで処理された全演算数に占める、ベクトル演算命令で処理された数の割合を求めたものです。

SX-8Rでは、このベクトル演算率を高くすることを目標に、チューニングを行ってください。

ここでは、ベクトル化率(ベクトル演算率)を向上させる手法の一つとして、コンパイラ指示行についてご紹介します。

4.2 コンパイラ指示行の挿入による高速化

コンパイラは、プログラムに対して自動的に最適なベクトル命令を生成しますが、例えば変数のもつ値のように、コンパイラがプログラムを解析してもわからない情報があると、必ずしも十分なベクトル化が行われるとは限りません。

このような場合に、コンパイラが知り得ない情報を利用者が与えることにより、ベクトル化の効果を一層促進させるものが、コンパイラ指示行です。

コンパイラ指示行は、

!CDIR オプション[,オプション]

*CDIR オプション[,オプション] (ソースが固定形式の場合のみ)

のいずれかの形式で、!CDIR, *CDIR は、1桁から5桁に書かなければなりません。

詳しくは「FORTRAN90/SX プログラミングの手引 3.2 コンパイラ指示行」を参照してください。

以下に、主な指示行のオプションとその使い方について説明します。

a) VECTOR/NOVECTOR

直後の配列式 または DO ループを自動ベクトル化の対象とする(VECTOR)か、対象としない(NOVECTOR)ことを指定します。

一般に VECTOR を指定する必要はありませんが、ベクトル長が小さく、ベクトル化しない方が効率の良いことがわかっていような場合に、NOVECTOR を指定します。

例 16 例えば「M は、1 又は 2 にしかなり得ない」ことを利用者が知っている場合、以下の様に NOVECTOR を指定して、ベクトル化を抑止したほうが効率がよい

!CDIR NOVECTOR

```
A(1:M) = B(1:M) * C(1:M) + D(1:M) * E(1:M) - F(1:M) * G(1:M)
```

b) NODEP

配列の定義・引用関係がコンパイル時に不明で、自動ベクトル化できない場合に、利用者が「定義・引用関係に矛盾が無いから、ベクトル化するように」指示するものです。

例 17 NK の値が正であればベクトル化できる。条件ベクトル化で NK の値を判断するコードが出力されるが、「NK の値が常に正である」ことを利用者が知っている場合、NODEP を指定することにより無条件にベクトル化される。

!CDIR NODEP

```
DO I=1, N
  A(I) = A(I + NK)
END DO
```

例 18 IP(I)の値に重複するものが無ければベクトル化できるが、もし重複するものがある場合にはベクトル化できない。通常コンパイラにはどちらか判断できず、またこの場合には条件を判断する条件ベクトル化もできない。

もし利用者が、「IP(I)の値に重複するものがない」ことを知っている場合には、NODEP を指定することによってベクトル化することができる。

!CDIR NODEP

```
DO I=1, N
  A( IP( I ) ) = A( IP( I ) ) + B( I )
END DO
```

c) SELECT(VECTOR)

指定した DO ループをベクトル化することを指示します。

例 19 ループの一重化ができない場合、コンパイラには変数 L,M,N の値がわからないので、通常最内側ループでベクトル化する。

しかし、例えば L の値がとても大きく、M や N の値がごく小さいことを利用者が知っているならば、次のように SELECT(VECTOR)指示行を挿入することにより、右のように変形されて、ループ長の長い DO I=1,L のループでベクトル化される。

!CDIR SELECT(VECTOR)

```
DO I = 1, L
  DO J = 1, M
    DO K = 1, N
      A(K, J, I) = B(K, J, I) + C(K, J, I)
    END DO
  END DO
```

```

      . END DO
END DO
      ↓ コンパイラによる変形のイメージ
DO J = 1, M
  DO K = 1, N
    DO I = 1, L      ! このループがベクトル化される
      A(K, J, I) = B(K, J, I) + C(K, J, I)
    END DO
  END DO
END DO

```

上の例で、もし、M の値がとて大きく、L や N の値が小さいならば、DO J=1, M の直前に
!CDIR SELECT(VECTOR)
 の指示行を挿入すればよいことは、もうおわかりのことと思います。

d) SHORTLOOP

SX システムのベクトル演算命令では、一つの命令で一度に最大 256 要素のデータを処理することができます。それでは、処理したいデータの要素数が 256 を超える場合にはどうでしょうか？この場合には次のように 256 回ずつの繰り返しを持つループ(例 21 では配列式)に分割してベクトル化を行います。

例 20 繰り返し回数が 256 を超える配列式のベクトル化の概念

```

C(1:1000) = A(1:1000) + B(1:1000)
      ↓
DO I = 1, 1000, 256      ! コンパイラが生成するループ
  C(I:MIN(I+255, 1000)) = A(I:MIN(I+255, 1000)) + B(I:MIN(I+255, 1000))
      ! この配列代入文は常にベクトル化される
END DO

```

このようにコンパイラが生成するループを、**ストリップマイニンググループ**と呼びます。

しかし、ループ長が常に 256 以下であることがわかっているならば、このストリップマイニンググループの処理が不要となりますので、このための処理時間を削減できると同時に、レジスタを効率的に使用することができます。

SHORTLOOP は、ループ長が必ず 256 以下であることを指示します。

例 21 変数 M の値が、常に 256 以下であることがわかっているならば、次の指示行を挿入することで、ストリップマイニンググループを作成しないようにします。

```

!CDIR SHORTLOOP
A(1:M) = B(1:M) + C(1:M)

```

FORTRAN90/SX コンパイラのコンパイラ指示行には、ここで紹介した他にもいろいろなオプションが指定でき、ベクトル化を制御することができます。詳細につきましては、「FORTRAN90/SX プログラミングの手引 3.2 コンパイラ指示行」をご参照ください。

5. 性能解析のツール

5.1 proginf 情報

先に述べた、ベクトル演算率を始め、プログラムの実行性能を調べる最も簡単で有効な情報がプログラム特性情報(proginf 情報)です。

proginf 情報は、プログラムの実行時に、

```
setenv F_PROGINF YES
```

または

setenv F_PROGINF DETAIL

を指定することで、表示されます。

図 6 は、setenv F_PROGINF DETAIL で採取した proginf 情報の例ですが、ベクトル演算率(図 6 の②)、ベクトル長(図 6 の①)ともに非常に大きく、とても効率よくベクトル化されたプログラムでの例を示しています。

もしベクトル演算率が低ければ、コンパイラや後述の簡易性能解析機能の出力情報をもとに、ベクトル化できていないループ構造を抽出し、ベクトル化できるように変形したり、指示行を挿入するなどして、ベクトル演算率の向上を図ります。

次に、平均ベクトル長が短い場合には、2重以上のループ構造を抽出し、SELECT(VECTOR)指示行を挿入してループ長の一番長いループがベクトル化されるようにすることで、ベクトル長を拡大します。また、ループ長が極端に短いループ構造がベクトル化されている場合には、NOVECTOR 指示行を挿入して、ベクトル化を行わないことも検討します。

```
***** Program Information *****
Real Time (sec)      :      23.579208      経過時間
User Time (sec)     :      22.588967      ユーザ時間
Sys Time (sec)      :      0.227827      システム時間
Vector Time (sec)   :      22.044968      ベクトル命令実行時間
Inst. Count         :      1680849884.     全命令実行数
V. Inst. Count      :      964802200.     ベクトル命令実行数
V. Element Count    :      215295433316.   ベクトル命令実行要素数
FLOP Count          :      95763953865.   浮動小数点データ実行要素数
MOPS                 :      9562.698298   MOPS 値
MFLOPS              :      4239.412620   MFLOPS 値
①→VLEN             :      223.149816     平均ベクトル長
②→V. Op. Ratio (%) :      99.668514     ベクトル演算率
Memory Size (MB)    :      240.031250     メモリ使用量
MIPS                 :      74.410213     MIPS 値
I-Cache (sec)       :      0.057218     命令キャッシュミス
O-Cache (sec)       :      0.043932     オペランドキャッシュミス
Bank (sec)          :      0.006768     バンクコンフリクト時間

Start Time (date)   : 2002/08/28 16:42:34   プログラムの開始日時
End Time (date)     : 2002/08/28 16:42:57   プログラムの終了日時
```

図 6 proginf 情報の出力例

5.2 簡易性能解析機能(fttrace)

proginf 情報では、プログラム全体のベクトル化率(ベクトル演算率)等を知ることができますが、手続き単位の性能を知ることはできません。

これに対して、簡易性能解析機能(fttrace 機能)を使用すれば、手続き単位や指定した範囲の性能情報を表示することができ、この情報を元にプログラムの性能上の問題点を調べることができます。

簡易性能解析機能の使い方を以下に示します。

- 1) プログラムを `-fttrace` オプションをつけてコンパイル・リンク
% sxf90 -fttrace test.f90
- 2) できた実行ファイルを実行
% a.out
- 3) `fttrace` コマンドを実行
% fttrace

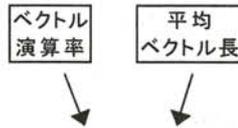
このようにするだけで、第 7 図に示すように手続き単位の平均ベクトル長、ベクトル演算率などの性能情報が出力されます。

なお、3)の `fttrace` コマンドを実行する代わりに、2)の実行の前に環境変数 `F_FTRACE` に `YES` を設定しておく、プログラムの実行が終了したときに、性能情報が自動的に表示されます。

FLOW TRACE ANALYSIS LIST

Execution : Wed Aug 28 16:42:57 2002
Total CPU : 0:00'22"569

手続き名	Call回数	実行時間
------	--------	------



PROG.UNIT	FREQUENCY	EXCLUSIVE	AVER.TIME		.MQPS	MFLOPS	V.OP	AVER
	BANK	TIME[sec](%)	[msec]		RATIO		V.LEN	
CONF								
calc2 0.0000	800	7.594(33.6)	9.492	9778.0	4874.4	99.72	222.3	
calc1 0.0000	800	7.053(31.2)	8.816	10119.8	4642.8	99.76	222.3	
calc3 0.0000	798	5.565(24.7)	6.973	9212.8	3827.9	99.77	222.3	
shallow 0.0067	1	2.346(10.4)	2346.201	8122.2	1998.5	98.87	232.1	
initial 0.0000	1	0.007(0.0)	6.979	6084.6	2188.4	98.93	222.4	
calc3z 0.0000	1	0.005(0.0)	5.479	3930.8	0.0	99.31	222.5	
total 0.0067	2401	22.570(100.0)	9.400	9570.7	4243.0	99.67	223.2	

図 7 ftrace コマンドの出力例

より詳しい使い方については、「FORTRAN90/SX プログラミングの手引 10.3 簡易性能解析機能」をご参照ください。

6. おわりに

以上、FORTRAN90/SX コンパイラの自動ベクトル化機能を中心にご紹介させていただきました。FORTRAN90/SX コンパイラは、他にも本稿でご紹介できなかった種々のベクトル化機能を持っています。詳細につきましては、「FORTRAN90/SX プログラミングの手引」を参照して下さるようお願い致します。

皆様が SX-8R と FORTRAN90/SX を使っていただくうえで、本稿が、多少なりともお役に立てれば幸いです。

参考文献

- (1) FORTRAN90/SX 言語説明書 (G1AF06)
- (2) FORTRAN90/SX プログラミングの手引(G1AF07)

センター報告

・2006年度大規模計算機システム利用者論文、研究成果一覧	-----	45
・徳島大学高度情報化基盤センター スーパーコンピュータ利用説明会報告		52
・2006年度前期・夏季利用者講習会 アンケート集計結果	-----	53
・大規模計算機システム Q&A・リンク集	-----	59
・2007年度大型計算機システム利用相談員・指導員一覧	-----	62
・2007年度大型計算機システム利用相談員・指導員自己紹介	-----	63

2006年度大規模計算機システム利用者論文、研究成果一覧

この論文、研究成果等一覧は、本センター大規模計算機システムを利用して2006年4月から2007年3月までに得られた研究成果等について、利用者から報告されたものを一覧として掲載しています。

1 学術雑誌掲載論文

- [1] 池野勉, 梶島岳夫: “管群内乱流の LES と二次流れの駆動力の解析,” 日本機械学会論文集 B 編 Vol.72, No.716, pp. 853-860, Apr. 2006.
- [2] 井上洋平, 梶島岳夫: “回転流路における乱流構造遷移の時系列解析,” 日本機械学会論文集 B 編 Vol.72, No.724, pp. 2933-2939, Dec. 2006.
- [3] Takeo Kajishima and Takayuki Nomachi: “One-equation subgrid scale model using dynamic procedure for the energy production,” *Journal of Applied Mechanics*, Vol.73, Issue 3, pp. 368-373, May. 2006.
- [4] Shintaro Takeuchi, Takahiro Yamazaki and Takeo Kajishima: “Study of solid-fluid interaction in body-fixed non-inertial frame of reference,” *Journal of Fluid Science and Technology*, Vol.1, No.1, pp. 1-11, May. 2006.
- [5] Tsutomu Ikeno and Takeo Kajishima: “Decay of swirling turbulent flow in rod-bundle,” *Journal of Fluid Science and Technology*, Vol.1, No.1, pp. 36-47, May. 2006.
- [6] Tsutomu Ikeno, Takeo Kajishima, T.Murata: “Efficient immersed boundary method for strong interaction problem of arbitrary shape object with the self-induced flow,” *Journal of Fluid Science and Technology*, Vol.2, No.1, pp. 1-11, Jan. 2007.
- [7] S. Furui and H. Nakajima: “Infraed features of unquenched lattice Landau gauge QCD,” *JFew-Body Systems*, Vol.40, pp. 101-128, 2006.
- [8] S. Furui and H. Nakajima: “Unquenched Kogut-Susskind quark propagator in lattice Landau gauge QCD,” *Phys. Rev. D*, Vol.73, p. 074503, 2006.
- [9] S. Furui and H. Nakajima: “Effects of the quark field on the ghost propagator of lattice Landau gauge QCD,” *Phys. Rev. D*, Vol.73, p. 094506, 2006.
- [10] Y. Koma, M. Koma and H. Wittig: “Nonperturbative determination of the QCD potential at $O(1/m)$,” *Phys. Rev. Lett.*, Vol.97, p. 122003, 2006.
- [11] Y. Koma, M. Koma: “Spin-dependent potentials from lattice QCD,” *Nucl. Phys. B*, in press, 2006.
- [12] T. Takaishi, Ph.de Forcrand: “Nonperturbative determination of the QCD potential at $O(1/m)$,” *Phys. Rev. E*, Vol.73, p. 036706, 2006.
- [13] T. Inakura, H. Imagawa, Y. Hashimoto, S. Mizutori, M. Yamagami and K. Matsuyanagi: “Mixed representation RPA calculation for octupole excitations on superdeformed states in the 40Ca and neutron-rich sulfur regions,” *Nucl. Phys.*, Vol.A768, pp. 61-79, 2006.
- [14] H. Iida, T. Doi, N. Ishii, H. Suganuma and K. Tsumura: “Charmonium properties in deconfinement phase in anisotropic lattice QCD,” *Phys. Rev. D*, Vol.74, p. 074502, 2006.
- [15] P. de Forcrand and O. Philipsen: “The chiral critical line of $N(f) = 2+1$ QCD at zero and non-zero baryon density,” *JHEP*, Vol.0701, p. 077, 2007.
- [16] K. Nawa, H. Suganuma and T. Kojo: “Baryons in Holographic QCD,” *Phys. Rev. D*, in press, 2006.
- [17] K. Nawa, H. Suganuma and T. Kojo: “Brane-induced Skyrmions –Baryons in Holographic QCD–,” *Prog. Theor. Phys. Suppl.*, in press, Vol. 168, 2006.
- [18] M.Garcia Perez and A.Gonzalez-Arroyo: “Gluino zero-modes for non-trivial holonomy calorons,” *JHEP*, Vol.0611, No. 091, 2006.
- [19] K. Yoshida, M. Yamagami, K. Matsuyanagi: “Pairing and continuum effects on low-frequency quadrupole vibrations in deformed Mg isotopes close to the neutron drip line,” *Nucl. Phys.*, Vol.A779, pp. 99-115, 2006.
- [20] Sugihara-Seki, Masako: “Transport of spheres suspended in the fluid flowing between hexagonally arranged cylinders,” *Journal of Fluid Mechanics*, Vol.551, pp. 309-321, 2006.
- [21] Daisuke Inazu, Naoki Hirose, S. Kizu, Kimio Hanawa: “Zonally asymmetric response of the Japan Sea to synoptic pressure forcing,” *Journal of Oceanography*, Vol. 62, pp. 909-916, 2006.
- [22] Hirose Naoki, Kenichi Fukudome: “Monitoring the Tsushima Warm Current improves seasonal prediction of the regional snowfall,” *SOLA*, 2, doi:10.2151/sola.2006-016, pp. 61-63, 2006.
- [23] A. Masago, K. Shirai and H. Katayama -Yoshida: “Crystal stabilities of α - and β -borons,” *Phys. Rev. B*, Vol.73, p. 104102, 2006.
- [24] K. Shirai, A. Masago, and H. Katayama -Yoshida: “High-pressure properties and phase diagram of boron,” *phys. stat. Vol. solidi (b)*, No. 244, p. 303, 2006.

- [25] K. Shirai, I. Hamada, and H. Katayama -Yoshida: "Vibration Problem of H in Silicon," *Physica*, Vol. 41, pp. 376-377, 2006.
- [26] K. Matsukawa, K. Shirai, and H. Katayama -Yoshida: "Gettering Mechanism of Transition Metals in Silicon Calculated from First Principles," *Physica B*, No. 224, pp. 376-377, 2006.
- [27] N. Mitsuda, K. Shirai, and H. Katayama -Yoshida: "Ab-initio Material Design for Control of DB in a-Si," *Physica B*, No. 240, pp. 376-377, 2006.
- [28] H. Nakayama, K. Shirai, and H. Katayama -Yoshida: "Instability of graphite structure induced by excited holes," *Physica B*, No. 296, pp. 376-377, 2006.
- [29] Kiyosi Horiuti: "Transformation Properties of Dynamic Subgrid-scale Models in a Frame of Reference Undergoing Rotation.," *J. of Turbulence*, 7, N17-1-27, 2006.
- [30] Hitoshi Iwamoto, Hidefumi Sakamoto, Keiichi Kimura: "Evaluation of Metal Ion Extractability of Crowned Spirobenzopyrans by Molecular Dynamics Calculation," *J. Computer Aided Chem.*, No. 7, pp. 1-9 (2006)., July 2006.
- [31] Kiyosi Horiuti and Takeharu Fujisawa: "Intermittent Dissipation Field in Multi-mode Stretched-spiral Vortex," *Bulletin of the American Physical Society*, Vol. 51, No. 9, p. 214, Sept. 2006.
- [32] 古賀裕明 大野隆央: "First-principles study of Ag adsorption on the H-passivated Si(001) surface with Bi nanolines," *Physical Review B*, Vol. 74, No. 12: p. 125405, Sept. 2006.
- [33] 重森啓介, 大谷一人, 境家達弘, 砂原淳, 中井光男, 藤岡慎介, 白神宏之, 疇地宏, 三間罔興: "二波長レーザー照射による Rayleigh-Taylor 不安定性の制御," *プラズマ・核融合学会誌* Vol. 82, No. 8: pp. 520-525, 2006.
- [34] Kiyosi Horiuti, Takeharu Fujisawa and Tetsuya Ozawa: "Multi mode stretched spiral vortex in homogeneous turbulence.," *J. Fluid Mech.*, under review, 2007.
- [35] S. I. Anisimov, V. V. Zhakhovskii, N. A. Inogamov, K. Nishihara, Yu. V. Petrov, and V. A. Khokhlov: "Ablated matter expansion and crater formation under the action of ultrashort laser pulse.," *JOURNAL OF EXPERIMENTAL AND THEORETICAL PHYSICS* vol. 103, no.2: pp. 183-197, 2006.
- [36] Mizuho Nagata, Takayoshi Norimatsu, Mitsuo Nakai, Keiji Nagai, Nobuyoshi Ueda, Shinsuke Fujioka, Tatsuya Aota, Hiroaki Nishimura, Katsunobu Nishihara, Noriaki Miyanaga, Yasukazu Izawa and Kunioki Mima: "Fabrication of Low Density Solid Xenon Composed of Microstructured powder.," *Jpn. J. Appl. Phys.*, No. 45 (33), pp. L884-L886, 2006.
- [37] H. Nagatomo, T. Johzaki, A. Sunahara, H. Shiraga, K. Mima: "Computational Study of Implosion Physics and Target Design for the Fast Ignition Experiment FIREX-I.," *Journal de Physique IV* France No. 133, pp. 397-400, 2006.
- [38] Norimatsu, T; Johzaki, T; Azechi, H; Mima, K; Nakao, Y; Sakagami, H; Shiraga, H; Kodama, R; Nagatomo, H; Endo, T; Iwamoto, A; Yoshida, H; Nakai, M: "Conceptual design of laser fusion reactor KOYO-fast - Target design and the fueling system.," *JOURNAL DE PHYSIQUE IV*, 2006.
- [39] T. Johzaki, H. Nagatomo, H. Sakagami, Y. Sentoku, T. Nakamura, K. Mima, Y. Nakao and T. Yokota: "Core heating analysis of fast ignition targets by integrated simulations.," *Journal de physique IV* No. 133, 2006.
- [40] Sakagami, H; Johzaki, T; Nagatomo, H; Mima, K: "Integrated simulations for fast ignition with cone-guided targets.," *JOURNAL DE PHYSIQUE IV*, 2006.
- [41] Sunahara, A; Mima, K; Johzaki, T; Nagatomo, H; Mima, K: "Non-local electron transport in laser-produced plasmas.," *JOURNAL DE PHYSIQUE IV*, 2006.
- [42] Mima, K; Tanaka, KA; Kodama, R; Johzaki, T; Nagatomo, H; Shiraga, H; Miyanaga, N; Murakami, M; Azechi, H; Nakai, M; Norimatsu, T; Nagai, K; Taguchi, T; Sakagami, H; Nagatomo, H; Mima, K: "Recent results and future prospects of laser fusion research at ILE, Osaka.," *JOURNAL DE PHYSIQUE IV*, 2006.
- [43] V. Zhakhovskii and K. Nishihara; Mima, K: "Self-consistent Monte Carlo simulation of particle motion and photon transport in the Argon positive column.," *J. Plasma Physics* vol. 72, pp. 1005-1009, 2006.
- [44] Chihiro Matsuoka and Katsunobu Nishihara: "Vortex sheet model for Rayleigh-Taylor and Richtmyer-Meshkov instabilities.," *JOURNAL DE PHYSIQUE IV* No. 133: pp. 171-173, 2006.
- [45] Chihiro Matsuoka and Katsunobu Nishihara: "Fully nonlinear evolution of a cylindrical vortex sheet in incompressible Richtmyer-Meshkov instability.," *Physical Review E* Vol. 73 (5), Art. No. 055304, 2006.
- [46] Akio HATA, Kunioki MIMA, Astushi SUNAHARA, Hideo NAGATOMO and Akio NISHIGUCHI: "Dynamics of Self-Generated

- Magnetic Fields in Stagnation Phase and their Effects on Hot Spark Formation.," Plasma and Fusion Research Vol. 1, No. 020, 2006.
- [47] Chihiro Matsuoka and Katsunobu Nishihara: "Analytical and numerical study on a vortex sheet in incompressible Richtmyer-Meshkov instability in cylindrical geometry.," PHYSICAL REVIEW E vol.74, no.6 Art. No. 066303, 2006.
- [48] H. Nagatomo, T. Johzaki, T. Nakamura, H. Sakagami, K. Mima: "Computational Studies and Designs for Fast Ignition.," The Physics of Ionized Gases 876, pp. 361-372, 2006.
- [49] M. B. Agranat, S. I. Anisimov, S. I. Ashitkov, V. V. Zhakhovskii, N. A. Inogamov, K. Nishihara, Yu. V. Petrov, V. E. Fortov, V. A. Khokhlov: "Dynamics of plume and crater formation after action of femtosecond laser pulse.," Applied Surface Science Journal, 2006.
- [50] H. Sakagami, T. Johzaki, H. Nagatomo, K. Mima: "Holistic Simulations for Fast Ignition with Cone-Guided Targets.," Europhysics conference abstracts 30I, 2006.
- [51] T. Johzaki, H. Nagatomo, H. Sakagami, T. Nakamura, K. Mima, Y. Nakao: "Integrated Simulations for Imploded Core Heating in Fast Ignition.," Europhysics conference abstracts 30I, 2006.
- [52] V. Zhakhovskii and K. Nishihara: "Self-consistent Monte Carlo simulation of particle motion and photon transport in the Argon positive column.," JOURNAL OF PLASMA PHYSICS vol. 72, No. 6, pp. 1005-1009, 2006.
- [53] S. I. Anisimov, V. V. Zhakhovskii, N. A. Inogamov, K. Nishihara and Yu. V. Petrov: "Simulation of the expansion of a crystal heated by an ultrashort laser pulse.," Applied Surface Science Journal, 2006.
- [54] H. Nagatomo, T. Johzaki, A. Sunahara, K. Mima: "The Formation of High Density Core Plasma in Non-spherical implosion using high-resolution 2-D Integrated Implosion Code.," Journal of Plasma Physics 72, 006.
- [55] 長友英夫、城崎知至、砂原淳、三間圀興: "レーザープラズマにおける流体力学的不安定性解析手法の検証.," 第20回数値流体力学シンポジウム講演論文集 D1-4, 2006.
- [56] 疇地 宏、兒玉了祐、坂上仁志、城崎知至、白神宏之、中尾安幸、長友英夫、三間圀興: "高速点火の物理と炉心プラズマ設計.," プラズマ・核融合学会誌 vol. 82, No. 12, pp. 823-828, 2006.
- [57] Ayush Gupta, T. M. Antonsen, T. Taguchi, and J. Palastro: "Effect of pulse duration on resonant heating of laser-irradiated argon and deuterium clusters.," Phys. Rev. E, Vol. 74, No. 4, pp. 046408-046501, 2006.
- [58] Toshihiro Taguchi, Toshikazu Matsumoto and Kunioki Mima: "Microscopic Analysis of Merging Processes of Relativistic Electron Beams Propagating in a High Density Plasma.," Laser Physics, Vol. 16, No.7, pp. 1116-1122.501, 2006.
- [59] Toshikazu Matsumoto, Toshihiro Taguchi and Kunioki Mima: "Simulation of the nonlinear evolution of large scale relativistic electron flow in dense plasmas.," Phys. Plasmas Vol. 13, No. 5, p. 052701, 2006.
- [60] A. Mizuta, T. Yamasaki, S. Nagataki, and S. Minoshige: "Collimated Jet or Expanding Outflow: Possible Origins of Gamma-Ray Bursts and X-Ray Flashes flow in dense plasmas.," The Astrophysical Journal Vol. 651, No. Issue 2, pp. 960-978, 2006.
- [61] Muneyuki Tsuda, Hideaki Kasai: "Ab initio study of alloying and straining on CO interaction with Pt.," Physical Review Vol. B73, No. 155405, pp. 1-8, Apr. 2006.
- [62] Muneyuki Tsuda, Nelson B. Arboleda Jr., Hideaki Kasai: "Initial driving force for proton transfer in Nafion.," Chemical Physics Vol. 324, pp. 393-397, Jun. 2006.
- [63] Nelson B. Arboleda Jr., Hideaki Kasai, Wilson Agerico Dino, Hiroshi Nakanishi: "Quantum Dynamics Study on the Interaction of H2 on a Pt(111) Surface.," Thin Solid Films Vol. 509, No. 1-2, pp. 227-229, 2006.
- [64] Muneyuki Tsuda, Hideaki Kasai: "Solvent effects on anionic and acid forms of Nafion side chain.," Japanese Journal of Applied Physics Vol. 45, No. 6A, pp. 5121-5125, 2006.
- [65] Tanglaw Roman, Hiroshi Nakanishi, Willson Agerico Dino, Hideaki Kasai: "Hydrogen atom quantum migration on platinum.," e-Journal Surface Science and Nanotechnology Vol. 4, pp. 619-623, 2006.
- [66] Nelson B. Arboleda Jr., Muneyuki Tsuda, Hideaki Kasai: "A quantum dynamics study on the H+ transmission in a Pt-Nafion interface.," e-Journal Surface Science and Nanotechnology Vol. 4, pp. 640-643, 2006.
- [67] Muneyuki Tsuda, Hideaki Kasai: "H2 dissociative adsorption on strained/CO-precovered Pt.," Japanese Journal of Applied Physics Vol. 45, No. 45, pp. L1219-L1221, 2006.

- [68] Nelson B. Arboleda Jr., Hiadeaki Kasai, Wilson Agerico Dino, Hiroshi Nakanishi: "Molecular Orientation Dependence of the Dissociative Adsorption of H₂ on a Pt(111) Surface.," *European Physical Journal D* 印刷中, 2006.

2 国際会議会議録掲載論文

- [1] M. Sugihara-Seki: "Flow structure and fluid shear stresses in water transport across the microvessel wall" *Proc. 2006 BMES Annual Fall Meeting (CD-ROM)*, , 2006.
- [2] S. Furui and H. Nakajima: "Correlation of the ghost and the quark in the lattice Landau gauge QCD" *Proc. 2007 Braz. J. Phys.*, , p. 37, 2007.
- [3] M. Koma Y. Koma and H. Wittig: "Determination of the spin-dependent potentials with the multi-level algorithm" *Proc. 2006 PoS*, , Vol. LAT2005, p. 216, 2006.
- [4] M. Koma: "Optimization of lattice QCD codes for the AMD Opteron processor" *Proc. 2006 Nucl.Instrum.Meth.*, , Vol. A559, pp. 95-98, 2006.
- [5] Toru Sekido, Katsuya Ishiguro, Yoshihiro Mori and Tsuneo Suzuki: "Abelian dominance in local unitary gauges and without gauge-fixing in pure SU(2) QCD" *Proc. 2006 PoS*, , Vol. LAT2006, p. 073, 2006.
- [6] Y. Taniguchi, M. Kimura and H. Horiuchi: "Coexistence of cluster structure and mean-field-type structure in medium-weight nuclei" *AIP Conf. Proc.*, , Vol. 831, p. 245, 2006.
- [7] Y. Taniguchi, M. Kimura, Y. Kanada-En'yo and H. Horiuchi: "Coexistence of cluster structure and mean-field-type structure in medium-weight nuclei" *Nucl. Phys. A, in press*, , 2006.
- [8] H. Iida, H. Suganuma and T.T. Takahashi: "Bound states of (anti)-scalar-quarks in SU(3)_c lattice QCD" *AIP Conf. Proc.*, *in press*, 2006.
- [9] T.T. Takahashi, T. Doi and H. Suganuma: "Nuclear force in Lattice QCD" *AIP Conf. Proc.*, , 2006.
- [10] T.T. Takahashi, T. Doi and H. Suganuma: "Meson-Meson and Meson-Baryon Interactions in Lattice QCD" *AIP Conf. Proc.*, , Vol. CP842, pp. 246-248, 2006.
- [11] F. Okiharu, H. Suganuma, T.T. Takahashi and T. Doi: "Multi-Quarks and Two-Baryon Interaction in Lattice QCD" *AIP Conf. Proc.*, , Vol. CP842, pp. 231-233, 2006.
- [12] N. Ishii, T. Doi, H. Iida, M. Oka, F. Okiharu: "Anisotropic Lattice QCD Studies of Penta-Quarks and Tetra-Quarks" *AIP Conf. Proc.*, , Vol. CP842, pp. 492-494, 2006.
- [13] H. Suganuma, K. Tsumura, N. Ishii and F. Okiharu: "Lattice QCD Evidence for Exotic Tetraquark Resonance" *Proc. of Sci.*, , Vol. LAT2005, p. 070, 2006.
- [14] H. Iida, N. Ishii, T. Doi and H. Suganuma: "J/Psi at High Temperatures in Anisotropic Lattice QCD" *Proc. of Sci.*, , Vol. LAT2005, p. 184, 2006.
- [15] T. Doi, N. Ishii, Y. Nemoto, M. Oka and H. Suganuma: "Anisotropic Lattice QCD Study of Pentaquark Baryons in Spin 3/2 Channel" *Proc. of Sci.*, , Vol. LAT2005, p. 064, 2006.
- [16] H. Suganuma, H. Ichie, F. Okiharu and T.T. Takahashi: "Lattice QCD Study for Confinement and Hadrons" *Moroccan Publisher*, , 2006.
- [17] F. Okiharu, T. Doi, H. Ichie, H. Iida, N. Ishii, M. Oka, H. Suganuma and T.T. Takahashi: "Tetraquark and Pentaquark Systems in Lattice QCD" *Quark Nuclear Physics: Nuclear and Hadronic Systems and Quark Degrees of Freedom, Pyoungchang, Kangwondo, Korea, 22-24 Feb. 2005*, 2006.
- [18] A. Diaz-Gil, J. Garcia-Bellido, M. Garcia Perez and A. Gonzalez-Arroyo: "Magnetic field production after inflation" *PoS*, , Vol. LAT2005, p. 242, 2006.
- [19] T. Doi, T. Blum, M. Hayakawa, T. Izubuchi and N. Yamada: "The isospin breaking effect on baryons with $N_f = 2$ domain wall fermions" *PoS*, , Vol. LAT2006, p. 174, 2007.
- [20] T. Doi: "Theoretical Status of Pentaquarks" *Prog. Theor. Phys. Suppl., YKIS06, YITP, Kyoto Univ.*, November. 2006.
- [21] Y. Nakagawa, A. Nakamura, T. Saito, H. Toki: "Infrared divergence of the color-Coulomb self-energy in Coulomb gauge QCD" *Proc. of Science*, , Vol. LAT2006, p. 071, 2006.
- [22] S. Negishi, H. Matsufuru, T. Onogi, T. Umeda: "Study of 1/m corrections in HQET" *PoS*, , Vol. LAT2005, p. 208, 2006.
- [23] K. Shirai, and H. Katayama -Yoshida: "Atom relaxation of H in silicon," *Proc. 28th Int. Conf. Physics of Semiconductors*, , FrM2f.21, 2006.
- [24] K. Matsukawa, K. Shirai, and H. Katayama -Yoshida: "Gettering Mechanism of 3d Transition Metals Studied by First Principles Calculation," *Proc. 28th Int. Conf. Physics of Semiconductors*, , FrM2g.4, 2006.

- [25] K. Shirai, H. Yamaguchi, and H. Katayama - Yoshida: "Control of Impurity Diffusion in Silicon by IR Laser Excitation," *Proc. International Conference on Quantum Simulators and Design, Hiroshima*, P. 24, 2006.
- [26] H. Dekura, K. Shirai, and H. Katayama - Yoshida: "Valence Control of α -rhombohedral Boron by Electronic Doping," *Proc. International Conference on Quantum Simulators and Design, Hiroshima*, P. 31, 2006.
- [27] K. Mima, H. Azechi, H. Nagatomo, T. Sakaiya, S. Fujioka, H. Shiraga, M. Nakai, S. Obenschain, A. Schmidt, M. Karasik, J. Gardner, J. Bates, D. Colombant, J. Weaver: "29th European Conference on Laser Interaction with Matter Spain," *Proc. Hyper-velocity Acceleration for Impact Fast Ignition*, June 11-16, 2006.
- [28] K. Mima, H. Sakagami, T. Johzaki, H. Nagatomo: "Cone-geometry dependence on characteristics of high energy charged particles for fast ignition," 2006.
- [29] H. Sakagami, T. Johzaki, H. Nagatomo, K. Mima: "Electron surface acceleration by oblique irradiation of intense laser pulses," 2006.
- [30] A. Mizuta, T. Yamasaki, S. Nagataki, and S. Mineshige Swift and GRBs: "Collimated Jet or Expanding Outflow: Possible Origins of GRBs and X-Ray Flashes," *Proc. Unveiling the Relativistic Universe* June 5-9, 2006 Venice (Italy).
- [31] M. Murakami, H. Nagatomo, T. Sakaiya: "1st Japan-Russia workshop on Impact Fast Ignition Russian Federation," *Proc. Recent activities on Impact Fast Ignition* July 10-13, 2006.
- [32] H. Nagatomo, H. Sakagami, T. Nakamura, K. Mima, Y. Nakao: "33rd European Physical Society Conference on Plasma Physics Italy," *Proc. Integrated Simulations for Imploded Core Heating in Fast Ignition* June 19-23, 2006.
- [33] Chiro Matsuoka, Vasilii Zhakhovskii: "10th International Workshop on the Physics of Compressible Turbulent Mixing French," *Proc. Nonlinear Evolution of Richtmyer-Meshkov Instability in Cylindrical Geometry* July 17-21, 2006.
- [34] M. Murakami, H. Nagatomo, T. Sakaiya: "1st Japan-Russia workshop on Impact Fast Ignition Russian Federation," *Proc. Recent activities on Impact Fast Ignition* July 10-13, 2006.
- [35] H. Nagatomo: "23rd Summer School and International Symposium on the Physics of Ionized Gases Serbia," *Proc. Computational Studies and Designs for Fast Ignition* August 28-September, 2006.
- [36] Murakami M., Nagatomo H., Sakaiya T., Azechi H., Saito H., Obenschain S., Karasik M., Bates J., Weaver J., Guskov S., Stepanov R., Rozanov V: "5th Int. Conf. Plasma Physics and Plasma Technology Belarus," *Proc. HYPER-VELOCITY ACCELERATION OF FOIL TARGETS FOR IMPACT IGNITION* September 18-22, 2006.
- [37] M. Murakami, H. NAGATOMO, T. SAKAIYA, T. JOHZAKI, S. FUJIOKA, H. SAITO, H. SHIRAGA, M. NAKAI, K. SHIGEMORI, H. AZECHI, M. KARASIK, J. GARDNER, J. BATES, D. COLOMBANT, J. WEAVER, A. SCHMITT, A. VELIKOVICH, Y. AGLITSKIY, J. SETHIAN, and S. OBENSHAIN: "21st IAEA Fusion Energy Conference China," *Proc. Hyper-velocity Acceleration of Foil Targets for Impact Fast Ignition* October 16-21, 2006.
- [38] T. Johzaki, T. Nakamura, H. Sakagami, Y. Nakao, T. Taguchi, A. Sunahara, K. Nishihara, K. Mima: "Fast Ignition Integrated Interconnecting Code (FI3) - Integrated Simulation and Element Physics," 2006.
- [39] H. Nagatomo: "48th Annual Meeting of the APS Division of Plasma Physics USA," *Proc. Simulation and design study of cryogenic cone shell target for fast ignition FIREX project* October 30-November 3, 2006.
- [40] T. Taguchi and K. Mima: "4Beam-plasma interaction analyzed by a hybrid simulation code," 2006.
- [41] H. Sakagami, H. Nagatomo, T. Nakamura, K. Mima, Y. Nakao: "9th International Workshop on Fast Ignition of Fusion Targets USA," *Proc. Fast Ignition Integrated Simulation for FIREX Project* November 3-5, 2006.
- [42] T. Johzaki, T. Nakamura, H. Sakagami, A. Sunahara, K. Mima: "Target designs of cone-guided implosion for fast ignition," 2006.
- [43] T. Shioya: "Twenty Years after SN 1987A (超新星1987爆発20周年記念国際会議) USA," *Proc. The structure of disks formed by merging two white dwarf*, 2006.
- [44] Sho Kusuda, Shintaro Takeuchi and Takeo Kajishima: "Parallel optimization of fish shape and swim mode by generic algorithm," *Proc. International Conference on Parallel Computational Fluid Dynamics* Busan, Korea, pp. 393-394, May. 2006.
- [45] Edouard Boujo, Tuan Mohammad Yusoff Shah Tuan Ya, Shintaro Takeuchi and Takeo Kajishima: "Study of fluid-structure interaction problems involving deformable objects by a new finite element - immersed boundary approach," *Proc. 3rd International Symposium on Aero Aqua*

Bio-Mechanisms (ISABMEC2006) Ginowan, Okinawa, No. P07 (CD-ROM), July. 2006.

- [46] Hiroki Ugajin, Masafumi Kawai, Kohei Okita, Yoichiro Matsumoto, Takeo Kajishima, Satoshi Kawasaki, and Hiroshi Tomaru: "Numerical simulation of unsteady cavitating flow in a turbopump inducer," *Proc. 42nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference* Sacramento, California, July. 2006.
- [47] Hiroki Sakai, Takeo Kajishima and Takashi Ohta: "Direct numerical simulation of vortex cavitations in turbulent shear layer," *Proc. International Symposium on Cavitation (CAV2006)* Wageningen, the Netherlands, No. 54 (electronic media), Sept. 2006.
- [48] Takashi Ohta, Kazuhiko Mizobata and Takeo Kajishima: "Influence of density fluctuation on DNS of turbulent channel flow in the presence of temperature stratification," *Proc. 5th International Symposium Turbulence, Heat and Mass Transfer* Dubrovnik, pp. 297-300, Sept. 2006.
- [49] Takeo Kajishima, Takashi Ohta, Hiroki Sakai and Kie Okabayashi: "DNS of vortex cavitations in turbulent separated layer," *Proc. Korea-Japan CFD workshop in 2006 Autumn Meeting of Korean Society of Computational Fluids Engineering* Daejeon, Korea, pp. 11-12, Oct. 2006.
- [50] Takeo Kajishima: "LES of turbulent flow around an airfoil at low Mach number," *Proc. K7th Sino-Japan Workshop on Turbulence* Huangshan, China, (CD-ROM) Nove. 2006.
- ### 3 国内研究会等発表論文
- [1] 関真佐子, 秋永剛, 板野智昭: "微小流路内の浸透流の及ぼす電荷の影響," 第56回理論応用力学講演会講演論文集, pp. 595-506, 2007.
- [2] Daisuke Inazu, Hirose Naoki, S. Kizu, Kenichi Fukudome, Kimio Hanawa: "Response of the Japan Sea to synoptic atmospheric forcing," EGU General Assembly, Apr. 2006.
- [3] Kiyosi Horiuti, Yohei Takagi and Shoji Abe: "Multi Modes for the Vortex Sheet-Tube Transformation Process and Viscoelastic Effect.," in *Proc. IUTAM Symposium in Elementary Vortices and Coherent Structures*, June. 2006.
- [4] Daisuke Inazu, Hirose Naoki, S. Kizu: "Response of the Japan/East Sea to synoptic wind forcing," Program of Korea-Japan Seminar on Local Physical Oceanographic Problems, July. 2006.
- [5] 大濱 瑛輔, 井門 英司: "枝付加による最大木数グラフの構成と同形グラフについて," 平成18年度電気関係学会四国支部連合大会, pp. 1-7, Sept. 2006.
- [6] 辻本 公一, 狩谷修次, 社河内 敏彦, 安藤 俊剛: "DNSによる噴流混合指標に関する研究," 日本流体力学学会年會 2006 講演論文集, AM06-11-013, pp. 1-4, Sept. 2006.
- [7] 辻本 公一, 狩谷修次, 社河内敏彦, 安藤俊剛: "DNSを用いた複合噴流の混合特性に関する研究," 日本機械学会第84期流体工学部門講演会概要集, p.251, Oct. 2006.
- [8] 藤澤竹春, 堀内潔: "多重モード spiral vortex における散逸構造," 第20回数値シンポジウム講演論文集, A7-5, Dec. 2006.
- [9] 笹島雄一郎, 広瀬直毅: "並列化されたネスティング RIAMOM の作成," 海洋中・深層循環のモデリング-水塊形成から大循環まで-, Jan. 2007.
- [10] Satoshi Nakada, Naoki Hirose, Jong Hwan Yoon, Yutaka Yoshikawa: "Topographically Induced Seasonal Upwelling by horizontal advection of the Tsushima Warm Current along the Japanese shelf area," Program of Korea-Japan Seminar on Local Physical Oceanographic Problems, Mar. 2007.
- [11] 辻本 公一, 藤井秀崇, 狩谷修次, 社河内敏彦, 安藤俊剛: "円形平行噴流についての DNS," 日本機械学会東海支部第56期総会講演会 講演論文集, pp. 257-258, Mar. 2007.
- [12] 梶島岳夫: "乱流の数値解析の基礎," 日本機械学会関西支部第284回講習会 「設計に使える熱流体解析の基礎と応用」 pp. 1-8, July. 2006.
- [13] 梶島岳夫: "分散混相流の直接数値シミュレーション," 化学工学会 粒子・流体プロセス部会セミナー 「現象のシミュレーション解析と装置開発・プロセス操作への応用」 pp. 1-8, Janu. 2007.
- [14] 大島翼, 太田貴士, 梶島岳夫: "剪断乱流の素過程に対する界面活性剤の効果," 日本機械学会流体工学部門講演会 講演論文集 CD-ROM, No.705, 2006.
- [15] 宇賀神裕樹, 河合理文, 沖田浩平, 松本洋一郎, 梶島岳夫, 都丸裕司, 川崎聡: "ターボポンプ用インデューサ内の非定常キャビテーション流れの数値計算," 日本機械学会流体工学部門講演会 講演論文集, CD-ROM, No.G704, Oct. 2006.
- [16] 井上洋平, 梶島岳夫: "壁面吹き出しを伴う回転流路における乱流構造," 第20回数値流体力学シンポジウム 講演論文集 (CD-ROM) No.A10-1, Dec. 2006.
- [17] 岩田隆一, 池野勉, 梶島岳夫: "境界埋め込み法を用いた気泡乱流の LES," 第20回数値流体力学シンポジウム 講演論文集 (CD-ROM) No.F8-2, Dec. 2006.
- [18] 森田功, 竹内伸太郎, 梶島岳夫: "埋め込み境界法を用いた粒子クラスター化過程における粒子間力依存性の解析," 第20回数値流体力学シンポジウム 講演論文集 (CD-ROM) No.F8-4, Dec. 2006.

- [19] 乙社知也, 太田貴士, 梶島岳夫: “1 方程式型ダイナミックモデルによる円柱周りの流れの LES 解析,” 第 20 回 数値流体力学シンポジウム 講演論文集 (CD-ROM) No.A4-5, Dec. 2006.
- [20] Tuan Mohammad Yusoff Shah Tuan Ya, 竹内伸太郎, 梶島岳夫: “Study of hydrodynamic force acting on an elastic object using immersed boundary and finite element methods,” 第 20 回 数値流体力学シンポジウム 講演論文集 (CD-ROM) No.B5-4, Dec. 2006.
- [21] 結城芳彦, 竹内伸太郎, 梶島岳夫: “IBM-FEM カップリング法による可変形粒子群を含む流れ場の直接数値シミュレーション,” 日本機械学会関西支部 第 82 期定時総会講演会 講演論文集, pp. 10-12, 2007.
- [22] 岩田隆一, 竹内伸太郎, 梶島岳夫: “VOF-IBM による固気液三相流の直接数値シミュレーション,” 日本機械学会関西支部 第 82 期定時総会講演会 講演論文集, pp. 9-16, 2007.
- [23] 森田功, 竹内伸太郎, 梶島岳夫: “集団的挙動する粒子群と流体の相互作用における粒子間力の及ぼす影響の解析,” 日本機械学会関西支部 第 82 期定時総会講演会 講演論文集, pp. 10-8, 2007.
- [24] 中村一樹, 溝端和彦, 太田貴士, 梶島岳夫: “弱圧縮性流れ解法による密度変動を考慮した温度成層溝乱流に関する研究,” 日本機械学会関西支部 第 82 期定時総会講演会 (大東) 講演論文集, pp. 9-20, 2007.
- [25] Hideaki Kasai: “マテリアルデザイン,” 第 2 回「計算科学による新たな知の発見・統合・創出」シンポジウム「計算科学の戦略と次世代コンピュータ」つくば国際会議場, Apr. 2006.
- [26] Hideaki Kasai, Tanglaw Roman: “環境調和と計算機ナノマテリアルデザイン: 水素エネルギーと燃料電池,” 第 4 回ナノ学会「ナノテク最前線—ナノ物質からナノ医療まで—」京都大学, May. 2006.
- [27] 笠井秀明: “次世代のデバイスをデザインする Computational Materials Design,” 第 20 回 CAE 懇話会, 日本原子力研究開発機構 関西光科学研究所, May. 2006.
- [28] 笠井秀明, 中西寛: “水素クリーンエネルギー基盤技術での計算機マテリアルデザインの展開,” 第 9 回コンピューテーショナル・マテリアル・デザイン (CMD) ワークショップ 国際高等研究所・日本原子力研究機構 関西光科学研究所, Sep. 2006.
- [29] 笠井秀明: “ナノ構造のダイナミクスとナノ触媒反応 燃料電池・水素貯蔵材料のデザイン,” 高度学際教育プログラム ナノマテリアル・ナノデバイスデザイン学 大阪大学中之島センター, Dec. 2006.
- [30] S. Furui and H. Nakajima: “Infrared features of unquenched finite temperature lattice Landau gauge QCD,” Preprint (hep-lat/0612009), 2006.
- [31] Toru Sekido, Katsuya Ishiguro, Yoshiaki Koma, Yoshihiro Mori and Tsuneo Suzuki: “Abelian dominance and the dual Meissner effect in local unitary gauges,” Preprint, submitted, 2006.
- [32] H. Iida, H. Suganuma and T. Takahashi: “Scalar-Quark Systems and Chimera Hadrons in SU(3)_c Lattice QCD,” Preprint, submitted to Phys. Rev. D, 2006.
- [33] S. Negishi, H. Matsufuru, T. Onogi: “Precision Study of $B^*B\pi$ Coupling for the Static Heavy-light Meson,” Preprint (hep-lat/0612029), 2006.

4 その他の特許

- [1] Hideaki Kasai, Hiroshi Nakanishi, Muneyuki Tsuda: “一酸化炭素の吸着性が制御された触媒の吸着性が制御された触媒および触媒の製造方法、並びにその利用,” K20060025, 2006 年 5 月 9 日, 特願 2006-130825.

5 その他の論文

- [1] Kazuhito Mizuyama: “Realistic continuum quasi-particle linear response theory with Skyrme interaction and multipole response in neutron rich nuclei,” Niigata University, 2006.
- [2] Toru Sekido: “The quark confinement and the dual Meissner effect in SU(2) lattice QCD,” Kanazawa University, 2006.
- [3] Shunsuke Negishi: “The Determination of the $B^*B\pi$ Coupling in Lattice QCD,” Department of Physics, Kyoto University, 2006.
- [4] Kenichi Yoshida: “Pair correlation and continuum coupling effects on low-frequency modes of excitation in deformed neutron-rich nuclei,” Department of Physics, Graduate School of Science, Kyoto University, 2006.

6 著書

7 その他の業績

徳島大学高度情報化基盤センター スーパーコンピュータ利用説明会報告

説明会講演者：大阪大学サイバーメディアセンター 下條、東田

大阪大学情報推進部 宮永、橋本

開催場所：徳島大学高度情報化基盤センター 1階 会議室

説明内容及び説明時間（質疑応答含む）：出席者（19名）

1. 大阪大学サイバーメディアセンターの近況（下條センター長） 13:30～14:15
2. 新スーパーコンピュータシステムの紹介について（東田助手） 14:15～15:00
3. スーパーコンピュータ利用デモンストレーションと、運用・サービス内容について
（情報基盤課教育・研究システム班 宮永） 15:10～16:10
4. 利用についての申請種類・方法について（情報企画課利用者サービス係 橋本） 16:10～16:20

2007年3月7日、徳島大学高度情報化基盤センターにご協力いただき、スーパーコンピュータの利用説明会を開催しました。

出張説明会の開催趣旨は、スーパーコンピュータの概要などを知っていただき、利用促進を図ることです。今回は徳島大学でシステム改変が行われたこともあり、是非開催してほしいという声をいただいたため、実現しました。その経緯から、大勢の熱心なスーパーコンピュータ利用者の皆様にご出席いただくことができました。

はじめに徳島大学高度情報化基盤センターの大恵教授にご発声いただき、続いて下條センター長よりセンターの概要および近況説明がありました。大阪大学サイバーメディアセンターが取り組んでいる事業に、積極的に関わりたいと大恵教授よりご意見をいただきました。



次に東田助手より、平成19年1月に更新されたばかりの新スーパーコンピュータについて詳細が説明され、出席者との熱心な質疑応答が交わされました。その後事務部の職員より、スーパーコンピュータのデモ画面を表示して実際の利用方法が示され、続け

て各種サービスや申請方法など、様々な説明が行われました。どちらも講演後活発な質疑応答がありました。最後に大恵教授より挨拶をいただき、その中で、徳島大学で19年度に定額制で利用申請いただけると話され、非常に有意義な説明会となりました。



説明会終了後にアンケートを実施し、期待していた内容と異なっていたという意見もいただきましたが、大半の出席者にはおおむね好評をいただきました。このような、出張による説明会は必要であるとの意見も多々ありましたので、今後も同様の説明会を開催し、一人でも多くのかたにスーパーコンピュータの利用者になっていただきたいと思います。

2006年度前期・夏季利用者講習会アンケート 集計結果

◎受講者数、アンケート回収数等

講習会名	開催日時	申込者数	受講者数	学内		学外	アンケート回収数
Nastran	6月13日～14日 10:00～17:00	1	1			1	1
Marc	6月19日～21日 10:00～17:00	受講希望者無し					
Gaussian03+Mol-Studio入門	6月23日 10:00～17:30	3	3	3			3
LS-DYNA	7月4日 14:00～17:00	受講希望者無しにつき中止					
Windowsライクの簡単操作でスーパーコンピュータを利用する	7月5日 14:00～16:00	受講希望者無しにつき中止					
AVS画像処理入門& 画像処理プログラミング	7月6日 10:00～17:00	10	9	9			9
スーパーコンピュータ初心者入門	7月12日 15:00～17:00	6	8	3		5	5
スーパーコンピュータ: ベクトル化、並列化編	7月13日 15:00～17:00	2	1	1			1
スーパーコンピュータ: パフォーマンス・チューニング編	7月14日 15:00～17:00	1	1	1			1
Patran	7月24日～26日 10:00～17:00	受講希望者無し					
スーパーコンピュータ初心者入門	9月14日 13:00～15:00	1	1	1			1
AVS画像処理入門& 画像処理プログラミング	9月25日 10:00～17:30	2	2			2	2
合計		26	26	18		8	23

[問1] あなた(参加者)は、◆本センターの利用者ですか? ◆所属は? ◆職種は?

講習会名	利用者			所属		職名						
	はい	いいえ	無回答	学内	学外	教員	技官	事務官	院生	学生	研究生	その他
Nastran	1				1	1						
Marc	受講希望者無し											
Gaussian03+Mol-Studio入門	2	1		3			1					2
LS-DYNA	受講希望者無しにつき中止											
Windowsライクの簡単操作でスーパーコンピュータを利用する	受講希望者無しにつき中止											
AVS画像処理入門& 画像処理プログラミング	6	3		9					1	8		
スーパーコンピュータ初心者入門	3	2			5					5		
スーパーコンピュータ: ベクトル化、並列化編	1			1					1			
スーパーコンピュータ: パフォーマンス・チューニング編	1			1					1			
Patran	受講希望者無し											
スーパーコンピュータ初心者入門	1			1						1		
AVS画像処理入門& 画像処理プログラミング	1	1			2				1			1
合計	16	7	0	15	8	1	1	0	4	14	0	3

[問2, 3]

講習会名	[問2] どのような方法でこの講習会を知りましたか？(複数回答可)							[問3] 開催口時は適当ですか？			
	速報	センター掲示のポスター	その他掲示のポスター	WWW	研究室の教員	研究室の知人	その他	適当	普通	不適当	わからない
Nastran	1							1			
Marc	受講希望者無し							受講希望者無し			
Gaussian03+Mol-Studio入門	1						2	3			
LS-DYNA	受講希望者無しにつき中止							受講希望者無しにつき中止			
Windowsライクの簡単操作でスーパーコンピュータを利用する	受講希望者無しにつき中止							受講希望者無しにつき中止			
AVS画像処理入門&画像処理プログラミング					6	3		7	1		1
スーパーコンピュータ初心者入門					5			1	4		
スーパーコンピュータ: ベクトル化、並列化編					1			1			
スーパーコンピュータ: パフォーマンス・チューニング編					1			1			
Patran	受講希望者無し							受講希望者無し			
スーパーコンピュータ初心者入門					1				1		
AVS画像処理入門&画像処理プログラミング					2			2			
合計	2	0	0	0	16	5	0	16	6	0	1

[問4, 5] 内容等

講習会名	[問4] 希望した内容に満足しましたか？					[問5] これからの研究に役立ちますか？				
	大変満足	満足	普通	悪い	わからない	大変役立つ	役立つ	普通	役立たない	わからない
Nastran	1					1				
Marc	受講希望者無し									
Gaussian03+Mol-Studio入門	1	2				2	1			
LS-DYNA	受講希望者無しにつき中止									
Windowsライクの簡単操作でスーパーコンピュータを利用する	受講希望者無しにつき中止									
AVS画像処理入門&画像処理プログラミング	1	7	1			3	6			
スーパーコンピュータ初心者入門			5				5			
スーパーコンピュータ: ベクトル化、並列化編			1				1			
スーパーコンピュータ: パフォーマンス・チューニング編			1				1			
Patran	受講希望者無し									
スーパーコンピュータ初心者入門		1					1			
AVS画像処理入門&画像処理プログラミング	1	1					2			
合計	4	11	8	0	0	6	17	0	0	0

[問6, 7] 内容等

講習会名	[問6] 講習内容は理解できましたか？				[問7] 講習会資料は適当で満足しましたか？			
	よく理解できた	大体理解できた	まったく理解できなかった	無回答	大変満足	満足	普通	悪い
Nastran	1				1			
Marc	受講希望者無し							
Gaussian03+Mol-Studio入門	2	1			1	1	1	
LS-DYNA	受講希望者無しにつき中止							
Windowsライクの簡単操作でスーパーコンピュータを利用する	受講希望者無しにつき中止							
AVS画像処理入門&画像処理プログラミング	2	7			3	6		
スーパーコンピュータ初心者入門	1	4				4	1	
スーパーコンピュータ: ベクトル化、並列化編	1						1	
スーパーコンピュータ: パフォーマンス・チューニング編	1						1	
Patran	受講希望者無し							
スーパーコンピュータ初心者入門	1					1		
AVS画像処理入門&画像処理プログラミング	2				1	1		
合計	11	12	0	0	6	13	4	0

[問8] 利用予定

講習会名	[問8] 差し支えなければ、どのような対象にご利用されるでしょうか？		回答内容
	回答	無回答	
Nastran	1		流動・熱伝導解析
Marc	受講希望者無し		
Gaussian03+Mol-Studio入門	1	2	サーバーの管理を行うため、ユーザ対応をする
LS-DYNA	受講希望者無しにつき中止		
Windowsライクの簡単操作でスーパーコンピュータを利用する	受講希望者無しにつき中止		
AVS画像処理入門&画像処理プログラミング	7	2	・研究、・流体の数値計算、・流体解析結果の可視化、・研究結果のプレゼン
スーパーコンピュータ初心者入門		5	
スーパーコンピュータ: ベクトル化、並列化編	1		電磁界シミュレーション
スーパーコンピュータ: パフォーマンス・チューニング編	1		電磁界シミュレーション
Patran	受講希望者無し		
スーパーコンピュータ初心者入門	1		三次元境界要素法による電磁場解析
AVS画像処理入門&画像処理プログラミング	2		・数値計算(電場)結果の出力 ・流体の数値計算の可視化
合計	14	9	

[問10, 12, 13,14] 内容等

講習会名	[問10] あなたは、スーパーコンピュータを使ったことがありますか？			[問12] 今後、大規模計算機システムのスーパーコンピュータの科学技術計算を利用されますか？			[問13] 今後、大規模計算機システムのアプリケーションサーバの科学技術計算を利用されますか？			[問14] 今後、大規模計算機システムのアプリケーションを利用されますか？		
	ある	ない	その他	する	しない	その他	する	しない	その他	する	しない	その他
Nastran		1			1		1			1		
Marc	受講希望者無し											
Gaussian03+Mol-Studio入門	2	1		2		1		2	1	2		1
LS-DYNA	受講希望者無しにつき中止											
Windowsライクの簡単操作でスーパーコンピュータを利用する	受講希望者無しにつき中止											
AVS画像処理入門&画像処理プログラミング	2	7		6	1	2	4	2	3	4	2	3
スーパーコンピュータ初心者入門	1	4		2		3	2		3	2		3
スーパーコンピュータ: ベクトル化、並列化編		1		1				1			1	
スーパーコンピュータ: パフォーマンス・チューニング編		1		1				1			1	
Patran	受講希望者無し											
スーパーコンピュータ初心者入門	1			1					1			1
AVS画像処理入門&画像処理プログラミング	1	1		2			2			1		1
合計	7	16	0	15	2	6	9	6	8	10	4	9

平成18年度前期・夏季利用者講習会 アンケート集計結果

([問4~7,9,11,15]に対する意見等)

講習会名：Nastran (講師：MSC ソフトウェア)

[問6] 講習内容は理解し満足できましたか？

- ・ よく理解できた (良かった) 点
補足資料も使って、説明が丁寧であったこと。

[問9] 今後、どのような機能の追加・改良を行えば、満足して利用できると思われますか。

マニュアルの日本語化

[問15] その他、ご意見・ご要望があれば、些細なことでも結構ですのでご記入ください。

(ソフトウェア、ハードウェア、通信関係)

Nastran も Marc のように、外部から手軽に使えるようにしていただきたいです。
とても役に立つ講習でした。

講習会名：Gaussian03+Mol-Studio (講師：NEC)

[問4] 希望した内容に満足しましたか？

- ・ 満足できた点
まったく知らなかったもので、input file の書き方がよくわかりました。

[問15] その他、ご意見・ご要望があれば、些細なことでも結構ですのでご記入ください。

(ソフトウェア、ハードウェア、通信関係)

サンプルファイルを使用する際 (Win 上で) あまりにもファイルが多く見つけにくかったもので、講習会で使用するファイルのみ用意しておくほうがよいかと思えます。

講習会名：AVS 画像処理入門&画像処理プログラミング (講師：KGT)

[問4] 希望した内容に満足しましたか？

- ・ 満足できた点
AVS の使い方がよくわかった。
AVS の様々な使い方がわかった。
実習が多く、質問もしやすかった。

[問6] 講習内容は理解し満足できましたか？

- ・ よく理解できた (良かった) 点
実習がスムーズにできた。CAVE がとても面白く、是非使いたいです。
少人数で丁寧に指導していただいた。

[問15] その他、ご意見・ご要望があれば、些細なことでも結構ですのでご記入ください。

(ソフトウェア、ハードウェア、通信関係)

アンケートの質問の最後のほうをわかりやすくしてください。
非常にわかりやすく、有意義な講習でした。

講習会名：スーパーコンピュータ「ベクトル化・並列化」編 (講師：NEC)

[問6] 講習内容は理解し満足できましたか？

- ・ よく理解できた (良かった) 点
ベクトル化・並列化の条件等。

[問15] その他、ご意見・ご要望があれば、些細なことでも結構ですのでご記入ください。

(ソフトウェア、ハードウェア、通信関係)

演習形式のほうがよかったかも。

講習会名：スーパーコンピュータ「パフォーマンス・チューニング」編（講師：NEC）

[問6] 講習内容は理解し満足できましたか？

- ・ よく理解できた（良かった）点

FTRACE の使い方、パフォーマンス改善のポイントがわかった。

[問15] その他、ご意見・ご要望があれば、些細なことでも結構ですのでご記入ください。

（ソフトウェア、ハードウェア、通信関係）

実機を触りながらのほうがよいと思う。

講習会名：スーパーコンピュータ初心者入門

（講師：レーザーエネルギー学研究所職員，教育・研究システム班職員）

[問6] 講習内容は理解し満足できましたか？

- ・ よく理解できた（良かった）点

エラーメッセージの内容がわかった。基本的なことがわかった。

[問11] スーパーコンピュータではどんなライブラリをご使用ですか。

ASL

講習会名：AVS 画像処理入門&画像処理プログラミング（講師：KGT）

[問6] 講習内容は理解し満足できましたか？

- ・ よく理解できた（良かった）点

基礎的な使用法全般

[問11] スーパーコンピュータではどんなライブラリをご使用ですか。

SSL2

[問15] その他、ご意見・ご要望があれば、些細なことでも結構ですのでご記入ください。

（ソフトウェア、ハードウェア、通信関係）

動画に落とす時の、ファイル形式の種類が増えるとありがたい。

大規模計算機システム Q&A・リンク集

サイバーメディアセンター

1. パスワードを変更したいのですが？

<https://portal.hpc.cmc.osaka-u.ac.jp/> に Web からアクセスし、ログイン ID、パスワードを入力を行って、新パスワードを指定します。この時、MAC O/S を利用されていれば、Safari では問題があります（現在調査中）ので、Firefox か Netscape でご利用ください。

2. 研究室のパソコンからスパコンを利用したいのですが、どうすれば良いのでしょうか？

Windows の PC から利用する場合は、

- ① SSH(Secure Shell)用のソフトウェアを PC にインストールする必要があります。
- ② Tera Term Pro をダウンロードサイトを選択します。フリーソフトウェアなので Web から検索サイトを選べます。または、阪大では下記の URL からダウンロードが可能です。
<http://www.hpc.cmc.osaka-u.ac.jp/j-o/tebiki/getteraterm.html>
- ③ Tera Term Pro をダウンロードします。
- ④ Tera Term Pro を PC にインストールします。
- ⑤ TTSSH を起動して、ログインサーバに接続します。
- ⑥ HOST フィールドにログインサーバのホスト名 : login.hpc.cmc.osaka-u.ac.jp 又は、133.1.65.34 を入力
- ⑦ Service として SSH を選択し、TCP Port#に 22 が表示されるのを確認
- ⑧ OK ボタンをクリックします。
- ⑨ Add this machine and its key to the known hosts list をチェックし、
- ⑩ Continue ボタンをクリックします。
- ⑪ 次に User name Password の画面がでますので、利用者番号とパスワードを入力します。
- ⑫ 正常にログインされると、次のメニューが表示され、接続ホストを選択します。



3. 大規模計算機システムにログインする方法を知りたいのですが？

下記の URL を参照ください。接続方法がわかります。

<http://www.hpc.cmc.osaka-u.ac.jp/j/tebiki/nyumon/koshukai20070509.pdf>

4. 大規模計算機システムのインターネットサービスの設定を知りたいのですが？

SSH によるログインサーバへ接続を行います。

下記の URL をご覧ください。設定がわかります。

http://www.hpc.cmc.osaka-u.ac.jp/j/service/internet_guide.html

5. 大規模計算機システムにファイル転送を行いたいのですが？
WinSCP を使用してファイル転送を行います。下記の URL を参照ください。
<http://www.hpc.cmc.osaka-u.ac.jp/j/tebiki/nyumon/koshukai20070509.pdf>
6. 大規模計算機システムの負担金一覧表を知りたいのですが？
下記の URL をご覧ください。負担金一覧表がわかります。
<http://www.hpc.cmc.osaka-u.ac.jp/j/futankin/index.html>
7. 大規模計算機システムの試用制度及び試用制度の申請をしたいのですが？
下記の URL をご覧ください。試用制度の申し込み及び内容を知ることができます。
<http://www.hpc.cmc.osaka-u.ac.jp/j/futankin/shiyou.html>
8. 大規模計算機システムの申請関係の情報を知りたいのですが？
下記の URL をご覧ください。申請書及び内容を知ることができます。
<http://www.hpc.cmc.osaka-u.ac.jp/j/shinsei/forms.html>
9. 大規模計算機システムを利用する場合の資格を知りたいのですが？
下記の URL をご覧ください。利用資格を知ることができます。
<http://www.hpc.cmc.osaka-u.ac.jp/j/shikaku/index.html>
10. 大規模計算機システムの刊行物の情報を知りたいのですが？
下記の URL をご覧ください。年報を掲載しています。
<http://www.cmc.osaka-u.ac.jp/j/intro/publication.html>
11. 大規模計算機システムのサポート情報を知りたいのですが？
下記の URL をご覧ください。種々の情報等を掲載していますので、ご覧ください。
<http://www.hpc.cmc.osaka-u.ac.jp/j/support/inquiry.html> 問合せ先情報
<http://www.hpc.cmc.osaka-u.ac.jp/j/support/advisor.html> 利用相談員情報
12. 他の6大学情報基盤センターの情報を知りたいのですが？
下記の URL をご覧ください。他大学情報基盤センターへリンクしています。
<http://www.cmc.osaka-u.ac.jp/j/intro/link.html>
13. スパコン会話利用の方法は、どうすれば良いのでしょうか？
下記の手順で行います。
①プログラムファイルを作成（例：sample.f）します。
②SX にログインします。
③SX8%f90 sample.f で翻訳
④SX8%a.out で実行、実行結果は画面に表示されます。

例) Gaussian03 の場合

SX8%g03 < input.dat > out.dat input.dat:入力データ、out.dat:出力ファイル

14. スパコンを NQS (バッチリクエスト) で利用するには、どうすれば良いのでしょうか？
下記の手順で行います。
① フロント端末でプログラム等のエディタ編集を行い、
② 編集後、ftp で ASCII 転送を行い、
③ qsub コマンドで SX のバッチ JOB 投入を行います。
④ nqs のバッチ JOB のファイル（例：nqsfile）の内容は次のようになります。

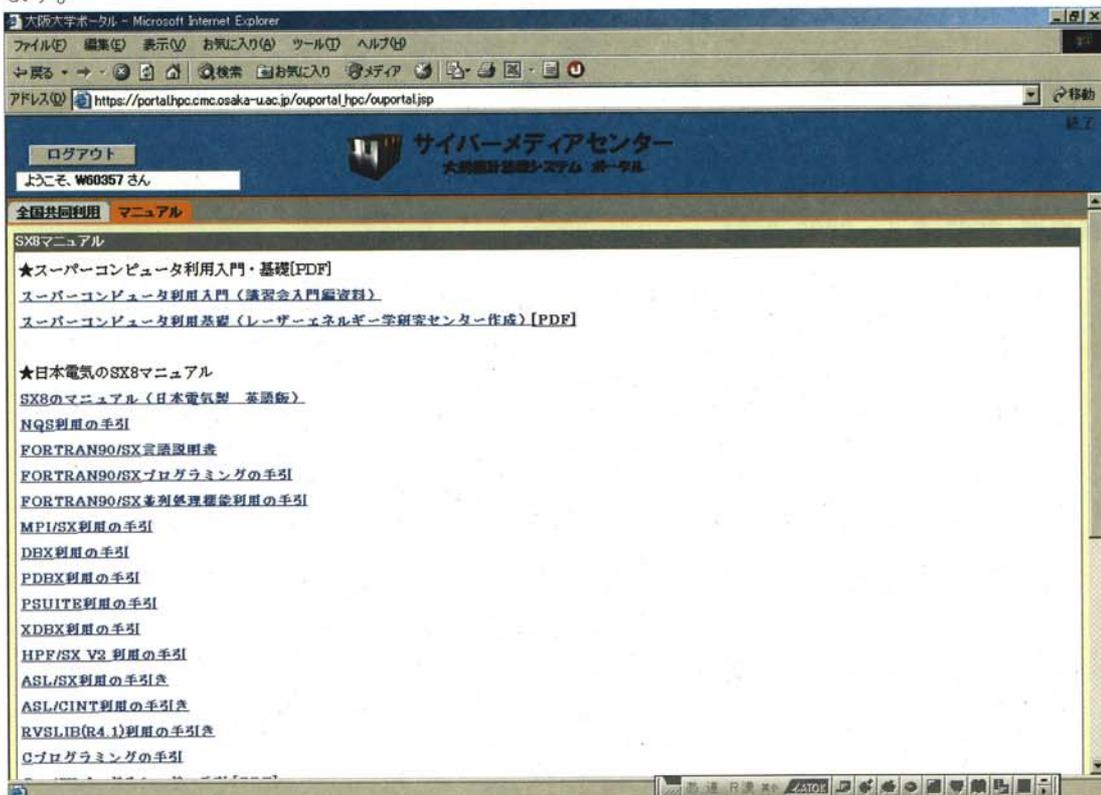
```
#!/usr/bin/csh
#PBS -q SXF@cmc
#PBS -l cpunum_job=4,elapstim_req=5:30:00,
memsz_job=10GB
#PBS -o nqs.out
#PBS -e nqs.err
#PBS -M user@lab.univ.ac.jp
#PBS -m e
cd [実行フォルダ]
setenv F_PROGINF DETAIL
a.out
プログラム翻訳は予め行い、a.outの実行モジュールを作成しておきます。
```

スクリプトを csh で実行宣言
ジョブクラス
使用 CPU 数、経過時間、メモリ容量の申告
標準出力ファイル名
標準エラー出力ファイル名
メールの送信先を指定
ジョブ終了時にメールを送信
実行するフォルダへ移動
解析情報を標準エラーへ出力
実行

- ⑤ front01% qsub nqsfile として JOB を SX に投入します。
- ⑥ 投入後、リクエスト ID が表示されます。
- ⑦ front01% qstat で、JOB の状況を表示させます。
- ⑧ front01% qdel 12345.cmc で JOB をキャンセルすることが可能です。12345.cmc : リクエスト ID です。

15. Web 上のマニュアルを参照できませんか？

Web から大規模計算機システムポータルにログインすることで、スーパーコンピュータシステムのマニュアルを参照することが可能です。下記に大規模計算機システムポータルにログインした時の画面を表示しています。画面にマニュアル名が表示されています。そこをクリックすることで参照することができます。



2007 年度大型計算機システム利用相談員

氏名	所属	職名	相談形態
高木 達也	大阪大学 大学院薬学研究科	教授	メール
山井 成良	岡山大学 総合情報基盤センター	教授	メール (常時)
武知 英夫	阿南工業高等専門学校 機械工学科	准教授	メール (常時)
柳澤 将	大阪大学 産業科学研究所	特任研究員	メール (木曜日 14 時～16 時)

* 委嘱期間 2007. 5. 1～2008. 3. 31

2007 年度大型計算機システム利用指導員

氏名	所属	職名
板野 智昭	関西大学 システム理工学部	講師
藤 堅正	近畿大学 理工学部	講師
武知 英夫	阿南工業高等専門学校 機械工学科	准教授

* 委嘱期間 2007. 5. 1～2008. 3. 31

2007 年度大型計算機利用相談員 自己紹介

たかぎ たつや
高木 達也 ttakagi@phs.osaka-u.ac.jp

(大阪大学大学院薬学研究科生命情報環境科学専攻・教授)

大阪大学大学院薬学研究科の高木達也と申します。宜しくお願ひ申し上げます。

近年は、化学、生物学分野でも計算機の比重が大きくなってきました。それと共に、皆が計算機をいとも簡単に使えるようになり、それはそれで喜ばしいことなのですが、誤った使い方も散見されるようになってきました。実験科学で誤った方法で行えば、何らかの手段で「誤っていること」がわかるのですが、計算科学の場合は何はともあれ結果は出てきますし、一見確かにも思えますので、結構やっかいなものです。新たに計算科学に手を出される皆さんは、是非、この利用相談員制度をご活用下さい。

なお、当方の専門は、本来計量薬学で、データマイニング、機械学習、統計学と言ったところが比較的近い分野だと思われませんが、現実には多いのは、Gaussian などの量子化学計算に対する相談です。専門というわけではありませんが、簡単な質問には答えられると思いますので、お気軽にご相談下さい。

このシステムが CMC の、ひいては大阪大学の活性化に繋がることを期待しております。

たけち ひでお
武知 英夫 x60602@center.osaka-u.ac.jp

(阿南工業高等専門学校 機械工学科 助教授)

平成 18 年度後半にシステムが更新されるという大変革があり、システムのバックアップとリストア、オブジェクトモジュールの互換使用が懸念されていたが、12 月末にネットワークと筐体本体が無事インストールされて、今年の年明けにはシステムが順調に稼働を開始した、更にオブジェクト互換が実現して殆どシームレスな更新が達成できたといつてよい。今後も継続して自動並列化プログラミングは重要であるが、システム性能を極限まで引き出すためには MPI 化が必要であり、重並列化に向けた一層のプログラミングの経験蓄積が不可欠になると思う。

やまい なりよし
山井 成良 yamai@cc.okayama-u.ac.jp

(岡山大学総合情報基盤センター 助教授)

岡山大学総合情報基盤センターの山井と申します。ご存知の方もおられると思いますが、平成 9 年 11 月まで大阪大学大型計算機センター研究開発部に在籍しておりました。現在も原則として月曜日の午前中は月に 2 回程度の頻度でセンターに詰めておりますので、電子メールだけでなくセンターでの直接相談も歓迎いたします。

さて、私の担当領域ですが、SX、APLSV などを含めた UNIX (LINUX を含む) システム全般とさせていただきます。NQS や LSF についての質問も歓迎致します。また、電子メールや SSH など、センターのネットワークサービスに関する質問も歓迎致します。本年度も微力ながら皆様のお役に立てれば幸いです。

柳澤 将 shou@cmp.sanken.osaka-u.ac.jp

(大阪大学 産業科学研究所 特任研究員)

大阪大学産業科学研究所で特任研究員をしている柳澤と申します。専門分野は物性物理・量子化学で、最近、有機分子膜と金属表面から成る界面の原子レベルの構造と電子状態を特定するための第一原理分子動力学シミュレーションを行っています。

このような表面・界面系の計算機シミュレーションでは、数百程度の原子からなる周期的な単位格子のモデルが必要であり、その中の電子の波動関数を数万～数十万の平面波基底で記述します。構造最適化や反応のシミュレーションには、スパコンのような大型の計算機上での並列計算が欠かせません。

計算には、大赤大学の森川良忠教授らが開発した STATE(Simulation Tool for Atom Technology)というプログラムコードを使用しています。この Fortran90 で書かれたプログラムは高いベクトル化率を有し MPI 並列を実装していますので、当センターの SX-8 は研究上大いに役立っています。

私の所属するグループには、他にも同様の計算シミュレーションをしている研究員や学生がいますが、その人数も年を経るごとに増えていますし、計算対象の表面・界面モデルもより現実的にするために、だんだん大きくなっています。したがって SX の使用規模も質・量ともに増加の一途をたどっているため、プログラムの効率化をはじめ、計算機をより効率的に利用できるよう考えていく必要性に迫られているところです。

昨年について利用相談員を勤めさせていただきますが、SX-8 の使用や Fortran90 の相談に関してお役に立つことができればと思っています。また、量子化学プログラム Gaussian03 の使用の経験もありますので、こちらでもお役に立てればと思います。よろしくお願ひします。

2007 年度大型計算機利用指導員 自己紹介

いた の とも あき
板 野 智 昭 itano@kansai-u.ac.jp

(関西大学 工学部 専任講師)

大阪府出身で吹田市にある府立千里高校卒業後、駿台予備校での真っ黒な浪人生活を経て、晴れて京都大学理学部に進学、高校で大好きになった物理学を修めました。趣味は旅と読書とサイクリングで、特に旅では見知らぬ土地で好んで友を作ることが好きです。研究に新しいアイデアをもたらすにも、旅とサイクリングによるリフレッシュが効果的(すなわち仕事の一部)だと信じています。

さて、私の専門は流体物理で、研究と計算機は切っても切り離せない関係にあります。計算機環境は愛用のラップトップにインストールした Linux で、研究時間の大半は計算機の前で過ごしてきました(ただし最近の研究以外の仕事が多すぎて、机の前に座る時間もかなり減っていますが・・・)。大型計算機も使い始めて10年くらいになるのでしょうか。長年の一利用者としてあちこちの計算機センターでお世話になっていますので、その視点から皆さんのお役に立てればと思っています。

たけ ち ひで お
武 知 英 夫 x60602@hpc.cmc.osaka-u.ac.jp

(阿南工業高等専門学校 機械工学科 助教授)

今年の正月にシステムが稼働を始めた際に、初めて `login.hpc.cmc.osaka-u.ac.jp` へアクセスしても手元のルータに情報が無いため接続できない状態が発生しました。またローカル linux の `/etc/hosts` に登録しようとしても ip アドレスが不明で、結局 `isc.tohoku.ac.jp` へ接続してから ping で得られた値を使用したのが、その後 `login.hpc.cmc.osaka-u.ac.jp` は DHCP の影響をうけるのか複数個存在することが判明した。新システムでは SX8 のオンラインアクセスが許可されるようになり、以前から SX コマンドで瞬時に解決できる処理を毎度 front 端末から NQS スクリプトを送信する必要があったのと比較すれば非常に歓迎すべきことだと思う。但し front 端末が以前のような汎用システムから、SX のために機能を大幅に削ぎ落としてクロス環境だけに限定して特化したために、数値計算以外何も出来ないマシンになってしまった。

ふじ けん しょう
藤 堅 正 fuji@ele.kindai.ac.jp

(近畿大学 理工学部 講師)

近畿大学の藤です。所属学科は、理工学部の学科再編で原子炉工学科から電気電子工学科(エネルギー工学コースが原子炉工学科に相当)になり、今年は第3期生が卒業研究を行います。

エネルギー材料(核燃料・原子炉材料)に関する実験系の研究室ですが、核燃料内部の化学状態を検討するための多相化学平衡計算や、燃料被覆管の水蒸気酸化シミュレーションあるいは燃料被覆材と核分裂生成物との固相反応の挙動解析に関する計算でサイバーメディアセンターを利用しております。

また、電子エネルギーレベルの計算をほんの少しだけお手伝いすることもあります。何れも FORTTAN で動いている関係上、研究室で適宜 FORTRAN のご相談を承っております。宜しく申し上げます。

附 表

・大規模計算機システム ホスト一覧	-----	69
・SX-8R のジョブクラスと制限値一覧	-----	69
・大規模計算機システム利用負担額一覧	-----	70

大規模計算機システム ホスト一覧

サーバ名	ホスト名	IP アドレス
ログインサーバ	login.hpc.cmc.osaka-u.ac.jp	133.1.65.34
ファイル転送サーバ	ftp.hpc.cmc.osaka-u.ac.jp	133.1.65.35
Mail サーバ	mail.hpc.cmc.osaka-u.ac.jp	133.1.4.44
Web サーバ	www.hpc.cmc.osaka-u.ac.jp	133.1.4.21
ポータルサーバ	portal.hpc.cmc.osaka-u.ac.jp	—
ライセンスサーバ	license.hpc.cmc.osaka-u.ac.jp	—

- ・スーパーコンピュータなどの演算システムへは、ログインサーバ経由での接続となります。
(ホスト一覧表には明記していません)
- ・Mail サーバ以下の各種サーバについては、用途に応じたサービスのみの接続が可能です。

SX-8R のジョブクラスと制限値一覧

スーパーコンピュータのジョブ資源制限値は次の通りです。また、使用するに当たって使用資源に見合ったクラスへのジョブ投入をお願いします。制限範囲内において CPU 数及びメモリー量を大きく要求すると、システムが資源確保に努めるために、要求値を確保できるまでジョブの実行が開始されませんのでご注意ください。

実行ノード	経過時間		使用 CPU 数		主記憶 (GB)	
	規定値 (分)	最大値 (時間)	規定値	最大値	規定値	最大値
SXF	1	24	1	8	1	120
SXL	1	120	1	32	1	1000
SXL(届出制)	1	240	1	64	1	2000

※ 既定値とは、演算時間・主記憶量を明記せずにジョブを投入した際にシステムがデフォルトで付与する値です。

大規模計算機システム利用負担額一覧

(平成 19 年 4 月 1 日現在)

区分	計算機資源の シェア値	スーパーコンピュータ		ファイル利用 の制限	年間負担額 (後期利用は半額)
		並列実行 CPU 数	メモリ制限		
基本 負担 額	1	4	16GB	10GB	0 円 (備考 7)
	1	4	16GB	10GB	1 万円
	10	4	32GB	300GB	10 万円
	50	8	制限なし	1TB	50 万円
	100	制限なし	制限なし	2TB	100 万円
	260	制限なし	制限なし	3TB	200 万円
	450	制限なし	制限なし	4TB	300 万円
ファイル追加 オプション	ファイル追加 50GB につき				1 万円
消費税額	上記負担額で算出した合計額に 100 分の 5 を乗じて得た額				

備考

- 1 基本負担額は年度の最初の登録時に算出する。
- 2 基本負担額の制限内でスーパーコンピュータ、クラスタシステム、ファイルなど計算機資源を利用できる。なお、スーパーコンピュータ、クラスタシステムにおける CPU・メモリなどの計算機資源は、フェアシェアスケジュール機能により設定したシェア値に応じて割り当てられる。
- 3 基本負担額 1 万円の場合、登録者数は 1 名とする。その他の場合、登録者数は特に制限を設けない。
- 4 後期 (10～3 月) 利用の基本負担額及びファイル追加オプションは、年間負担額の半額とする。
- 5 上記の基本負担額以外に 50 万円単位での申請を 1,000 万円を上限として受け付ける。その場合のシェア値及びファイルシェア利用の制限の設定については以下のとおりとする。
 シェア値は 300 万円未満が基本負担額の 1.3 倍、300 万円以上が基本負担額の 1.5 倍とする。
 ファイル利用の制限は、50 万円につき 0.5TB を加算する。
- 6 ファイルサーバはファイル使用量の制限内で利用できる。なお、制限値以上の利用は 50GB 単位での追加オプションとする。
- 7 別に定める試用制度による利用を認められた者は、基本負担額 1 万円の場合と同じ資源を、登録のあった月から、前期 (4 月～9 月) 3 ヶ月間、又は後期 (10 月～3 月) 1 ヶ月間無料で利用できる。ただし、当該会計年度を超えての利用はできないものとする。
- 8 大学院の学生が基本負担額 1 万円を利用する場合、負担額を半額とする優遇措置を受けられる。

募 集

- ・大規模計算機システムを利用して行った
研究・開発等の記事の募集について ----- 73

大規模計算機システムを利用して行った研究・開発等の記事の募集について

センターでは、大規模計算機システムを利用して研究したことを主体とする内容の広報誌の発行を予定しています（5月、12月）。この広報誌に掲載する次の内容の記事を募集しますので、皆様のご投稿をお待ちします。

1. 随筆
2. 大規模計算機システムを利用して行った研究・開発の紹介
3. プログラムの実例と解説
4. その他、広報誌に掲載するにふさわしいもの

*投稿いただいた方には、ご希望により掲載した広報誌5部をお渡しします。

【原稿執筆に当たってご注意いただく事項、及び原稿の提出方法】

a) Tex の場合（オフセット印刷可能なもの）

- Latex のソースファイルは、MS-DOS 形式のフロッピーあるいは電子メールで提出してください。
- 図面・画像は、別途提出してください。また、図表や写真の挿入箇所を指示しておいてください。
- 書式は、
 フォントサイズ：10point
 スタイルファイル：jarticle
 上余白：20mm、下余白：20mm
 左余白：20mm、右余白：20mm

b) ワードプロセッサ出力（オフセット印刷可能なもの）

- テキストファイルは、MS-DOS 形式のフロッピーあるいは電子メールで提出してください。
- 図面及び画像は、別途提出してください。また、図表や写真の挿入箇所を指示しておいてください。
- 書式は、
 フォントサイズ：10point
 フォント：MS 明朝
 上余白：20mm、下余白：20mm
 左余白：20mm、右余白：20mm

a,b 共通)

- 1 頁 1 段の行数：50 行
 1 行の文字数：24 文字
 1 頁 2 段書き、フォントは MS 明朝
- 数字、英字は半角、フォントは Times New Roman
 数字、英字を括弧で閉じる場合は、括弧も同様に

半角

- 文字、漢字は全角。文字、漢字を括弧で閉じる場合の括弧は全角
- 文中の半角の “，” や “：” は、全て全角の “、” や “。” とする。
- 原稿の題名は、MS ゴシックで 14Point。執筆者氏名は、姓と名の間及び所属機関・学部・専攻名の間を半角スペースとし、フォントサイズは 10Point、フォントは MS 明朝とする。

・注意事項

- a) 投稿原稿は原則として、サイバーメディアセンターホームページに公開されますことをご了承ください。なお、公開を希望されない場合には、その旨をご連絡してください。また、原稿の返却はいたしません。
- b) 執筆者には、ご希望により冊子を5部進呈します。
- c) 初回の校正は、誤植の防止をはかるため執筆者にお願いします。
- d) 原稿の送付先は、下記のとおりです（大阪大学内の方は、学内便でお願いします）。
 郵送：〒567-0047
 大阪府茨木市美穂ヶ丘5-1
 大阪大学サイバーメディアセンター
 情報推進部情報企画課
 利用者サービス係
 TEL 06-6879-8808 FAX 06-6879-8814
 E-mail：usersv@cmc.osaka-u.ac.jp

編集後記

「計算機利用ニュース Vol.3 No.1」をお届けします。

今回はスーパーコンピュータ SX-8R 特集です。

コンピュータの性能は、驚くべき速さで進歩しています。たとえば、1960年代に初めてアポロが月へ飛んだときの制御コンピュータの処理能力は、1980年代に大ヒットしたゲーム機並だという話を聞いたことがあります。なんとなく信じられませんが、それほどコンピュータの進歩はめざましいということなのでしょう。そんな中サイバーメディアセンターでは定期的に新型のスーパーコンピュータを導入し、皆様にサービスを提供してきました。そして、このたび2007年1月、新型のスーパーコンピュータシステムが導入されました。今回の計算機利用ニュースではその性能・特色を存分にご紹介できたものと思います。

新しいものが登場すれば、当然古いものは姿を消します。寂しい気もしますが、その分も新しいスーパーコンピュータをよろしく願います。また、この計算機利用ニュースが計算機を利用する皆様にとって有意義なものとなりましたら幸いです。

最後になりましたが、執筆者の皆様には非常にご多忙な中ご寄稿いただき、誠にありがとうございました。この場をお借りして心よりお礼申し上げます。また、編集作業の遅延により発行時期が遅れましたことをお詫びいたします。

(利用者サービス係 M.I)

広報委員会委員

- 小田中 紳 二 (委員長、大阪大学サイバーメディアセンター)
- 藤 堅 正 (近畿大学理学部)
- 豊 永 昌 彦 (高知大学理学部)
- 前 迫 孝 憲 (大阪大学大学院人間科学研究科)
- 養 老 真 一 (大阪大学大学院法学研究科)
- 小 郷 直 言 (大阪大学大学院経済学研究科)
- 阿 部 浩 和 (大阪大学サイバーメディアセンター)
- 清 川 清 (大阪大学サイバーメディアセンター)
- 竹 蓋 順 子 (大阪大学サイバーメディアセンター)
- 時 田 恵一郎 (大阪大学サイバーメディアセンター)
- 馬 場 健 一 (大阪大学サイバーメディアセンター)

大阪大学サイバーメディアセンター
計算機利用ニュース 2007年度 Vol.3 No.1
2007年 5月発行

編集者 大阪大学サイバーメディアセンター
広報委員会
(小田中紳二、藤堅正、豊永昌彦、前迫孝憲、
養老真一、小郷直言、阿部浩和、清川清、
竹蓋順子、時田恵一郎、馬場健一)

発行者 大阪府茨木市美穂ヶ丘5-1(〒567-0047)
大阪大学サイバーメディアセンター
Cybermedia Center, Osaka University
Tel : 06-6879-8808
URL : <http://www.hpc.cmc.osaka-u.ac.jp/j/>

印刷所 大阪市福島区玉川3-6-4
阪東印刷紙器工業所

目 次

特集 SX-8R	1	2007年度大型計算機システム利用相談員・ 指導員自己紹介	63
スーパーコンピュータSX-8Rの特長と概要	3		
スーパーコンピュータSX-8Rハードウェア	8		
FORTRAN90/SXの自動並列化機能	15	附 表	67
FORTRAN90/SXの自動ベクトル化機能	30	大規模計算機システム ホスト一覧	69
		SX-8Rのジョブクラスと制限値一覧	69
センター報告	43	大規模計算機システム利用負担額一覧	70
2006年度大規模計算機システム利用者論文、 研究成果一覧	45	募 集	71
徳島大学高度情報化基盤センター スーパーコンピュータ利用説明会報告	52	大規模計算機システムを利用して行った 研究・開発等の記事の募集について	73
2006年度前期・夏季利用者講習会アンケート 集計結果	53	編集後記	74
大規模計算機システムQ&A・リンク集	59		
2007年度大型計算機システム利用相談員・ 指導員一覧	62		

(お願い)

このニュースは、本センター利用者（利用登録者）の皆様に配布しています。
お近くの研究者・大学院生の方にも、このニュースをご回覧くださるようお願い申し上げます。