

J A E R I - M
92-054

第7回「原子力におけるソフトウェア開発」
研究会報告集

1991年10月30～31日、東海研究所

1992年3月

原子力コード研究委員会・炉物理研究委員会

日本原子力研究所
Japan Atomic Energy Research Institute

JAERI-M レポートは、日本原子力研究所が不定期に公刊している研究報告書です。
入手の問合せは、日本原子力研究所技術情報部情報資料課（〒319-11茨城県那珂郡東海村）あて、お申しこしください。なお、このほかに財団法人原子力弘済会資料センター（〒319-11 茨城県那珂郡東海村日本原子力研究所内）で複写による実費領布をおこなっております。

JAERI-M reports are issued irregularly.

Inquiries about availability of the reports should be addressed to Information Division
Department of Technical Information, Japan Atomic Energy Research Institute, Tokai-mura,
Naka-gun, Ibaraki-ken 319-11, Japan.

©Japan Atomic Energy Research Institute, 1992

編集兼発行 日本原子力研究所
印 刷 いばらき印刷株

第7回「原子力におけるソフトウェア開発」研究会報告集

1991年10月30～31日，東海研究所

日本原子力研究所東海研究所

原子力コード研究委員会・炉物理研究委員会

(1992年3月5日受理)

本報告書は、平成3年10月30日、31日に、日本原子力研究所東海研究所において開催された第7回「原子力におけるソフトウェア開発」研究会での発表論文の内容を収録したものである。研究会は、原子力コード研究委員会と炉物理研究委員会の共催によって行われた。本研究会での発表論文は、(1)計算機シミュレーションの新展開、(2)超並列計算機とその応用、(3)原子炉における3次元核熱計算のシミュレーションに関するものであった。

東海研究所：〒319-11 茨城県那珂郡東海村白方字白根2-4

プログラム実行委員会：

原子力コード研究委員会 高野 誠、横川三津夫、伊藤 伸泰、白土 幸保（以上原研）
植木紘太郎（船研）、湊 明彦（日立）、班目 春樹（東大），
藤村統一郎（NEDAC）

炉物理研究委員会 秋濃 藤義、森 貴正、水本 元治、柴田 勝成（以上原研）
金城 勝哉（動燃）、芳賀 暢（原工試）

Proceedings of the 7th Topical Meeting on Nuclear Code Development
October 30~31, 1991, Tokai, JAERI

Nuclear Codes Committee
and
Committee on Reactor Physics

Tokai Research Establishment
Japan Atomic Energy Research Institute
Tokai-mura, Naka-gun, Ibaraki-ken

(Received March 5, 1992)

This report is the proceedings at the 7th Topical Meeting on Nuclear Code Development held at Tokai Research Establishment, Japan Atomic Energy Research Institute (JAERI), on October 30 and 31, 1991. The meeting was organized in cooperation with the Japanese Committee on Reactor Physics (JCRP) and Japanese Nuclear Codes Committee (JNCC). The topics of the meeting are (1) Computational Science, (2) Massively Parallel Processors and Applications, and (3) Three-Dimensional Numerical Simulation in Nuclear Reactors.

Keywords: Software, Reactor Physics, Nuclear Code, Computational Science, Parallel Computer, Dynamic Behavior Analysis, Proceedings, Meeting

目 次

1. 開会の辞	
佐野川好母（原子力コード研究委員会委員長）	1
2. 原子蒸気流の数値シミュレーション	
鷲木 英雄（日本原子力研究所）	2
3. 陽子リニアック設計の為のコンピューターコード	
加藤 隆夫（高エネルギー物理学研究所）	20
4. Massively Parallel Supercomputing as the Centerpiece of a Diversified, Multidisciplinary Research Program	
Prof. Claudio Rebbi(Boston Univ.)	39
5. 並列計算機による遮蔽安全評価用モンテカルロコードMCACEの高速化	
高野 誠, 他 (日本原子力研究所)	58
6. 知能ロボットにおけるセンサー融合による状況認識 —物理空間法について—	
伊藤 伸泰, 他 (日本原子力研究所)	75
7. 固体物性の計算物理	
宮下 精二 (京都大学)	86
8. 3次元輸送コードの開発の現状とベンチマーク計算	
竹田 敏一, 他 (大阪大学)	96
9. 3次元動特性解析コードによるBWRの安定性の解析	
瀧川 幸夫, 他 (東芝)	112
10. SIMMER-Ⅲコードと高速炉の空間依存動特性	
近藤 悟, 他 (動燃)	129
11. パネル討論会「原子力分野における専用計算機への期待」	
11.1 プラズマ・シミュレーション専用計算機—METIS	
谷 啓二 (日本原子力研究所)	152
11.2 並列計算機とモンテカルロ遮蔽計算	
高野 誠 (日本原子力研究所)	153
11.3 超並列コンピュータへの展望	
栗田 康男 (日本IBM)	155
11.4 スーパーコンピュータの動向について	
小池 誠彦 (日本電気)	156
11.5 リアルワールド情報を実時間処理するための超並列プロセッサ	
浅川 和雄 (富士通)	157

11.6 並列計算のためのソフトウェア	
梅谷 征雄（日立製作所）	158
12. ポスターセッション	
12.1 3次元炉心ヒートアップおよびFP放出源評価コード：SHAPE	
芳賀 輝，他（原工試・安解析）	159
12.2 LESによる乱流数値シミュレーション	
佐竹 信一，他（東京理科大学）	169
12.3 イオンビーム引き出しシミュレーションにおける並列化手法	
久保田昌晴，他（埼玉大学）	170
12.4 MIMD型並列計算機の物理シミュレーションへの応用	
彦坂 聰，他（埼玉大学）	171
12.5 標準遮蔽計算コードシステムINTEL-BERMUDAの開発	
長谷川 明（日本原子力研究所）	174
12.6 知的原子力設計システム（IRDS）の開発	
久語 輝彦，他（日本原子力研究所）	175
12.7 ベクトル化モンテカルロコードの開発と応用	
森 貴正，他（日本原子力研究所）	176
13. 閉会の辞	
石黒 幸雄（炉物理研究委員会 原子炉システム専用部会長）	177
付録 研究会出席者名簿	179

Contents

1. Opening Address	1
K. Sanokawa (Chairman of Japanese Nuclear Codes Committee)	
2. Direct Numerical Simulation of Atomic Vapor Flow	2
H. Kaburaki (JAERI)	
3. Computer Codes for Designing Proton Linear Accelerators	20
T. Kato (KEK, National Laboratory for High Energy Physics)	
4. Massively Parallel Supercomputing as the Centerpiece of a Diversified, Multidisciplinary Research Program	39
C. Rebbi (Boston University)	
5. Speedup of Monte Carlo Code MCACE for Shielding Analysis by Parallel Computer	58
M. Takano, et al. (JAERI)	
6. Multi-sensor Fusion Method for Intelligent Robots - Physical Space Method	75
N. Ito, et al. (JAERI)	
7. Computational Approach to Condensed Matter Physics	86
S. Miyashita (Kyoto Univ.)	
8. Development of a 3D Neutron Transport Code and Benchmark Tests	96
T. Takeda, et al. (Osaka Univ.)	
9. BWR Stability Analysis with Three-Dimensional Transient Code ..	112
Y. Takigawa, et al. (Toshiba)	
10. SIMMER-III Code and Space-Energy Dependent Dynamics of Fast Reactors	129
S. Kondo (PNC)	
11. Panel Discussion	
11.1 An Exclusive-Use Computer for Plasma Simulation "METIS"	152
K. Tani (JAERI)	
11.2 Parallel Computer and Monte Carlo Shielding Calculation	153
M. Takano (JAERI)	
11.3 Towards Super-Parallel Computer	155
Y. Kurita, et al. (IBM Japan Ltd.)	
11.4 Trends in Supercomputing	156
N. Koike (NEC Corporation)	

11.5 Massively Parallel Computer for Real World and Real Time Information Processing	157
K. Asakawa (Fujitsu Lab. Ltd.)	
11.6 Software for Parallel Processing	158
Y. Umetani (HCRL)	
12. Poster Session	
12.1 3D Core Heatup and FP Source Evaluation Code: SHAPE	159
T. Haga, et al. (NUPEC/JINS)	
12.2 Numerical Simulation of a Turbulent Flow by LES	169
S. Satake, et al. (Sci. Univ. of Tokyo)	
12.3 Parallelization Method for Ion Beam Extraction Simulations .	170
M. Kubota, et al. (Saitama Univ.)	
12.4 Performance of Physical Simulation Applied in MIMD Type Parallel Processing Systems	171
S. Hikosaka, et al. (Saitama Univ.)	
12.5 Development of a Standardized Shielding Calculation Code System: INTEL-BERMUDA	174
A. Hasegawa (JAERI)	
12.6 Development of Intellectual Reactor Design System (IRDS) ...	175
T. Kugo, et al. (JAERI)	
12.7 Development of Vectorized Monte Carlo Codes and Their Applications	176
T. Mori, et al. (JAERI)	
13. Closing Address	177
Y. Ishiguro (Japanese Committee on Reactor Physics)	
Appendix List of Participants	179

1. 開会の辞

佐野川好母（原子力コード研究委員会委員長）

原子力コード研究委員会と炉物理研究委員会の共催による「原子力におけるソフトウェア開発」研究会は、今年で第7回目を迎えることになりました。今回も多数の方のご参加を頂き、この研究会も関係者の努力によって、皆様方のご意見も取り入れながら、だんだんと定着してきたことは大変喜ばしいことあります。

コンピュータの最近の発展はめざましいものがあります。それも、従来のコンピュータ利用による単なる解析とは異なって、科学計算という新しい科学方法論が生み出されて、進展を見せております。それは、「未来の事象を仮定し、計算モデルにより既知の原理に到達する」というものであります。

計算科学における利用は、すなわち、対象・表象を要素に還元して計算機による「分析」を行い、還元済みの要素を計算機によって全体、あるいは部分を「統合」し、全体のシステムを理解することにより、新しい理論を構築するものであり、例えば、統計物理における多自由度系の非線形問題や、感覚・知覚の統合・学習での人工知能などに利用されています。これは、計算機の新しい方向として、今後も進展を続けるものと期待されます。

しかし、こうした計算科学には、現在の1000倍程度の計算機のスピード、すなわち1秒間に1兆回程度の加減乗算(Tera FLOPS)が要求されますので、コンピュータのパワーアップは、これまで以上に重要な課題となっています。

今回の研究会では、こうした背景も踏まえて、

- ① 計算機シミュレーションの新展開
- ② 超並列並列計算機とその応用
- ③ 原子炉における3次元核熱計算のシミュレーション

をメインテーマに選び、更に「原子力分野における専用計算機への期待」というテーマでパネル討論会を企画しました。

また、今回初めての試みとして、ポスターセッションも予定しています。そこでは、

- ① 3次元炉心ヒートアップ及びFP放出、遮蔽、ベクトル化モンテカルロコード、知的原子力設計システムの開発
- ② 乱流数値、物理シミュレーション
- ③ イオンビーム引き出しシミュレーションにおける並列化手法

などが予定されています。これらも含めて、活発な意見交換が行われることを期待したいと思います。

原研のことになりますが、現在、東海研究所で計算科学研究グループの組織化が検討されており、今回の研究会の成果がこの組織化にも生かされることが期待されています。

なお、今回も(財)原子力データセンターのご好意により、計算科学における超並列計算機を駆使したスーパーコンピューティングの最前線について、ボストン大学教授 クラウディオ・レビ(Claudio Rebbi)博士に講演して頂くことになりました。多数ご聴講下さるようお願い致します。

最後に、2日間にわたるこの研究会が充実した実り多いものであることを祈りまして、私のご挨拶といたします。

2. 原子蒸気流の数値シミュレーション

鷲木 英雄 (原研)

各種の蒸着過程、原子分光学、原子法レーザー同位体分離過程等では、原子蒸気流の詳細な解析が要求されている。ここでは、円形領域から噴出する対称原子蒸気ジェットに直接シミュレーションモンテ・カルロ法を適用して2次元の数値シミュレーションを行なった結果を報告する。大量蒸発を行なう場合は、連続流に近い領域から希薄気体領域に至るまで幅広い領域の流れが対象となる。特に、直接シミュレーションモンテ・カルロ法は蒸発面近傍の流れの詳細な解析に有効である。しかし、連続流に近い流れに対して直接シミュレーションモンテ・カルロ法を用いる場合、粒子数、サンプリング数の計算結果に与える影響は明確にされていない。そこで、2次元キャビティ流れに対し、衝突過程でBird法と修正南部法の2種類のアルゴリズムを用いてシミュレーションを行ない、その結果の比較検討を行なった。この結果をもとにして、種々の連続流に近い流れに直接シミュレーションモンテ・カルロ法を適用した結果を紹介する。

Direct Numerical Simulation of Atomic Vapor Flow

Hideo Kaburaki (JAERI)

Detailed numerical analyses of atomic vapor flows have been called for in vaporization processes of vapor deposition, atomic spectroscopy, and atomic vapor laser isotope separation process. Direct simulation Monte Carlo (DSMC) method has been applied to the two-dimensional symmetric evaporating spherical jet. In the case of strong evaporation, the flow near the surface is near-continuum, while it becomes rarefied as it expands into vacuum space. In particular, the DSMC method is suited for the analysis of evaporating flow near the surface. However, in applying the DSMC method to near-continuum flow, effects of the number of quasi-particles and samplings on simulation results have not been made clear. In this paper, detailed comparisons have been made on the two-dimensional cavity flow with a wide range of simulation particles and samplings. Two different algorithms in the collision process, the Bird method and the modified Nambu method, are studied. On the basis of these results, some of the simulation results in the near-continuum region are presented.

Keywords (atomic vapor flow, DSMC method, spherical jet, cavity flow)

1. はじめに

近年のスーパーコンピュータの発達に伴い、気体の流れを模擬粒子を用いて計算する粒子法による数値シミュレーションがさかんに行なわれるようになった。粒子法シミュレーションの代表的なものとして分子動力学法 (Molecular Dynamics method : 以下 MD 法と呼ぶ) と直接シミュレーションモンテ・カルロ法 (Direct Simulation Monte Carlo method : 以下 DSMC 法と呼ぶ) がある。これら粒子法は、非常に多くの原子からなる実際の流れを、約 10^3 個以上の模擬粒子で代表させてシミュレーションする方法である。

MD 法では初期状態のみ確率的に粒子を配置するが、その後は仮定したポテンシャルに従って、運動方程式により決定論的に位置と速度が決定される。一方、DSMC 法では、粒子を確率的に衝突を起こさせて運動量の交換を行ない、変更された速度を用いて粒子の移動を行なう方法である。

これら粒子法では、ボルツマン方程式もしくは連続体領域において流れを記述するナビエ・ストークス方程式を差分法等により離散化して数値計算する方法に比較して、境界条件等でより単純な仮定を設定する事できる等、流れ場の詳細な情報をより多く引出す事ができる。また、差分法等により偏微分方程式を離散化する方法に比較して、粒子法は数値解析手法による不安定性を生じない利点がある。しかし、粒子を扱う手法では統計的ゆらぎが生じるためサンプリングにより巨視的量を求める過程が重要になる。

Meiburg⁽¹⁾は、MD 法と DSMC 法を用いて、スペースシャトルの大気圏突入時を想定して、45° に傾いた平板下の流れを計算した。その結果、MD 法では、平板下に渦が発生したが、同じ条件下で DSMC 法で計算した結果渦の発生は見られなかった。Meiburg は、この結果に基づき DSMC 法で渦の発生が見られないのは、角運動量の保存が破られているためであると推測した。これに対し Bird⁽²⁾ は、Meiburg の計算は DSMC 法の基本的な前提条件が満たされていないためであると反論した。更に、MD 法は希釈気体 (dilute gas) のシミュレーションには不適切であると結論した。

ここでは、DSMC 法を用いて真空中への蒸発原子噴流の 2 次元数値シミュレーションを行なった。これは、各種の蒸着過程、原子分光学、原子法レーザー同位体分離過程等で原子蒸気流の性質を調べるために重要である。このシミュレーションでは、蒸発面近傍から真空中へ原子蒸気が急激に広がり各場所における平均自由行程が大きく変化する。そのため、上記に述べた DSMC 法の基本的前提条件を厳密に満たすためには DSMC 法についての計算条件を厳密に見直す必要がある。そこで、第 2 章では、DSMC 法の計算手法、アルゴリズムを簡単に述べた後、2 次元蒸発原子噴流の正確な計算に必要な自動衝突セルの生成法について述べる。また、巨視的量を導く際、2 次元シミュレーションで使用する模擬粒子の数とサンプリング数は大きな影響を与えるため、ここでは、2 次元キャビティ流れを用いてこれらの関係を明らかにした。また、この結果は、非定常流れを計算する場合に役に立つ。第 3 章では、第 2 章での知見を基に、円形領域から噴出する対称原子蒸気噴流の結果を述べる。また、特に、蒸発面近傍のクヌーセ

ン層における原子蒸気の反射について考察を行なった。第4章では、クヌーセン数が低い連続体に近い流れについての数値計算例について述べる。

2. 直接シミュレーションモンテ・カルロ (DSMC) 法

2.1 計算手法、アルゴリズム

DSMC法は、非常に多くの原子から構成される現実の流れを、 10^3 程度もしくはそれ以上の模擬粒子で代表させて、それらを確率的に衝突させたり境界と相互作用させてシミュレーションする方法である⁽³⁾。この数値計算手法は、ボルツマン方程式に基づいていたため、ボルツマン方程式の基本的仮定（二体衝突、瞬時の衝突、分子混沌の仮定）を満たす必要がある。そのためこの方法は、ボルツマン方程式と同様、希釈気体(dilute gas)に対して適用される。更に、DSMC法の計算では、次の様な近似を行なっている。第一に、計算では有限の模擬粒子を用いている。実際には、コンピュータの制約から実際の流れと同じ数の粒子を用いる事ができないため、本質的なゆらぎ以外の統計的なゆらぎが存在する。そのため、巨視的量を求めるには、サンプリングをする必要がある。第二に、計算領域を有限のセルに分割する必要がある。セルに分割するのは、粒子のサンプリングと粒子間衝突を行なわせるためである。セルの大きさは、気体の平均自由行程程度に小さい方が望ましい。第三に、平均自由時間程度に非常に小さい時間間隔 Δt_m では、原子の運動と衝突が分離できることである。

以上の基本的仮定に基づいて、DSMC法のアルゴリズムは次の様になる⁽³⁾。まず、初期状態を設定するため、計算領域全体に速度と気体の温度に対応したゆらぎを持たせた粒子を確率的にばらまく。次に、時間間隔 Δt_m に対応して粒子を移動する。この時、境界との相互作用を考慮する。粒子の移動後、衝突を起こさせるため、粒子をセルに割当てる。そこで、時間間隔 Δt_m に対応して、衝突セルの中の粒子を衝突させ運動量の交換を行なう。ここで、衝突セル中の粒子の位置は、セルの中心点で代表させるとする。剛体球近似では、衝突の散乱角は等方的なのでランダムに選ぶ。ここで、非定常解析の時は巨視的量を導くためのサンプリングを行ない、時間を進める。定常解析では、十分時間が経過した後、サンプリングを行なう。

衝突過程では、タイムカウンタ法による2つの方法を用いた。第一の方法は、Bird法⁽³⁾である。この方法では、衝突セルの中からランダムに衝突ペアを選び出す。最初に、2粒子間の最大相対速度 g_{max} を仮定しておく。この g_{max} は、衝突の度に新しい最大相対速度に更新していく。この値は、数回の繰返しで収束する。選んだ衝突ペアの相対速度 g_{ij} と g_{max} の比が、一様乱数と比較して小さければこのペアは衝突するとする。この時、次のタイムカウンタ τ を計算する。

$$\tau = 2 / N n \sigma g_{ij} \quad (1)$$

ここで、 N は衝突セル中の模擬粒子の数、 n は気体の数密度、 σ は気体の散乱断面積である。相対速度の比が一様乱数より大きければこのペアは衝突しない。この過程

を繰返し、タイムカウンタの総和 $\Sigma \tau$ が Δt_m より大きくなった所で衝突過程を終了する。

第二の方法は、より厳密な意味でボルツマン方程式と対応させた修正南部法⁽⁴⁾である。この方法では、衝突セル中の全ての粒子を扱う。粒子 i が粒子 j と衝突する確率は、

$$P_{ij} = n \sigma g_{ij} \Delta t_m / N \quad (2)$$

である。粒子 i が衝突セル中の他の粒子と衝突する確率は、

$$P_i = \sum_{j=1}^N P_{ij} \quad (3)$$

である。Fig. 1に示すように、[0, 1]の線分を N 個の等しい領域に分け、 P_{ij} を割当てる。一様乱数を発生させ [0, 1] の上に置く。この乱数が、 $[(j-1)/N, j/N]$ の区間に落ちた場合、粒子 i の衝突相手は粒子 j である。また、乱数がこの区間の右側にある場合、粒子は衝突するとし、左側の場合は衝突しないものとする。衝突が起こる場合は、運動量保存則に従い衝突後の速度を計算し、粒子 i のみの速度を変更する。これを衝突セル内のすべての粒子について行なう。

2.2 自動衝突セルの生成

DSMC法では、粒子間の衝突を計算をする場合衝突セル中の粒子の位置は考慮されないため、衝突セルはできるだけ小さくとる（場の平均自由行程程度もしくはそれ以下）必要がある。一方、統計平均をとるサンプリングセルの大きさは場の性質が著しく変化しない程度にとる必要がある。Bird⁽²⁾は、この点を考慮して計算を進めるため、サンプリングセルの中に衝突サブセルの概念を導入した。この概念の導入により、DSMC法の基本的な仮定を破らずに計算を進める事が可能になった。しかし、例えば蒸発原子噴流のように、場中の平均自由行程が連続体に近い流れから希薄気体領域まで激しく変化する流れをシミュレーションする場合、計算領域中の場所によって衝突セルが大きく変化する。このため、固定の衝突サブセル用いた場合でも、流れの変化に伴い衝突セルの大きさに十分注意する必要がある。固定セルでは、特に非定常の計算において、DSMC法の基本仮定を局所的に厳密に満たすことが困難になる。そこで、Kaburakiら⁽⁵⁾は、計算を進めるに従い場の量を局所的に評価してサンプリングセルの中に自動的に衝突セルを生成する方法を提案した。通常、サンプリングセルは流れ場の勾配を十分とらえればよいため、平均自由行程程度より大きくとれる。そこで、噴流の計算の場合には、おおよその平均自由行程を見積っておき、最も流れ場の変化が大きい所にあわせて、平均自由行程程度もしくはそれ以上の大きさのサンプリングセルを生成する。ここでは、場の量から平均自由行程を評価してサンプリングセルの

中に衝突サブセルを各ステップ毎に生成しながら、計算を進める。一方、平均自由行程が大きい希薄気体領域では、衝突セルよりサンプリングセルの方が大きくなる場合がある。この場合、粒子間の衝突はサンプリングセルの中で行なう。ただし、ここでは衝突セルすなわちサンプリングセルの大きさを時間間隔 Δt_m 音速との積より小さくとっておく必要がある。

2.3 2次元キャビティ流れにおける粒子数、サンプリング数の影響

DSMC法では、計算機の制約より、非常に数多くの原子から構成される現実の流れをそれより少ない模擬粒子で置き換えてシミュレーションを行なう必要がある。そのため、瞬時平均(instantaneous averages)を行なった計算結果には、統計的なゆらぎが入っている。一般に3次元の場合では、巨視的平均量のゆらぎはサンプルサイズの平方根の逆数に比例しており、サンプルサイズを 10^3 以上とらなければ統計的ゆらぎが小さくならない事が知られている⁽³⁾。ここで扱うような2次元の流れに対しては、3次元の流れを任意に薄く切り取った断面と考えれば、現実の流れの原子数と模擬粒子数を対応させるように断面の厚さを調節できる。しかし、現実の流れにおける原子数に比較して模擬粒子の数は少ないため、時間平均またはアンサンブル平均により巨視的量を求める必要がある。時間平均は、定常の流れの場合に用いられる。また、非定常流れの場合は、巨視的には同じ初期条件から出発して計算を繰返すアンサンブル平均が用いられる。また、周期的境界条件を用いた場合の様に一つの方向が一様であると考えられる時は、空間的な平均をとることも考えられる。

現在の計算機では、場全体に投入できる模擬粒子の数は 10^6 程度が限界であると思われる。2次元の流れにおいては 100×100 のセルを用いると、セル当たりの粒子数は約100個程度となる。セルの大きさを平均自由行程程度と考えれば、上記の値は気体の標準状態と比較してもまだかなりの開きがある。巨視的量を求める場合、統計的ゆらぎに寄与するのは、セル当たりの粒子数とサンプリング数と考えられる。また、粒子数とサンプリング数の積である全サンプルサイズも一つの指標になると考えられる。しかしながら、2次元の流れでは、これらの値の計算結果に与える影響は詳細には調べられていない。特に、非定常の計算をするとき、これらの情報が必要となる。

そこで、ここでは2次元のキャビティ流れを例にとり粒子数、サンプリング数の数値計算結果に与える影響を調べた。計算は次の条件下で行なった。正方形の計算領域は $64 (=8 \times 8)$ 個のサンプリングセルに分割する。キャビティ内の気体は地上から100km上空の空気に相当する。初期設定状態の圧力と温度はそれぞれ $0.002[\text{Pa}]$ 、 $283[\text{K}]$ である。クヌーセン数(Kn)は 0.042 、上壁の移動速度は初期状態の気体の最確速度の0.6倍に設定する。境界条件としては、上部の移動する壁は拡散反射、他の壁は鏡面反射とした。衝突サブセルは、サンプリングセルの中で生成するがだいたい 4×4 から 2×2 程度になるよう設定してある。計算では、大部分 3×3 の衝突サブセルが生成されていた。時間の1ステップは、平均自由時間の0.1倍にとった。この流れでは計算を始めたらすぐ定常に達するが、色々なケースを比較するため、まづ5000時間ステップ計算を進め、次の12,000ステップでサンプリングを行なった。この条件下での2次元キ

ヤビティ流れでは、実験または厳密解は存在しないため、粒子数（セル当たり2000個）とサンプリング数（1000回）が最大の場合の結果を基準ケースとした。DSMC法の衝突過程では、Bird法と修正南部法の両方法について比較検討を行なった。Fig. 2に基準ケースの場合について行なったBird法と修正南部法による流線図の計算結果を示す。流線の等高線は、他の場合と比較するため同じレベルの値を描いている。ただし、流線は質量流量を基にして描いている。Fig. 3は衝突過程にBird法またFig. 4には修正南部法を用いた時、サンプリング数とセル当たりの粒子数を変化させた場合の流線図を示している。Fig. 3のBird法の場合では、セル当たりの粒子数が、100個と200個の間で大きな変化を示しているのが分かった。これは、一つの衝突サブセル当たりでは、約10個と20個に対応している。また、セル当たり200個以上の粒子が入っていれば、サンプリング数には大きくは依存していないことが分かる。サンプリング数が極端に少ないケースでも、粒子数を増加していくと結果が改善されているのが分かる。これより、Bird法ではセル当たりの粒子数が200個（衝突サブセル当たり20個）を越えると全般に良い結果が得られているのが分かる。また、全サンプリングサイズ（=粒子数×サンプリング数）を合せても同じ結果は得られない事が明らかになった。サンプリング数を上げるより、粒子数を増加する方が有効である。一方、Fig. 4の修正南部法を使用した場合では、サンプリング数、粒子数とも、多い時は非常に良い結果が得られる。しかし、粒子数、サンプリング数が減少するにつれて急激に結果が悪くなる傾向があることが分かった。これより、瞬間平均をいくつかとるアンサンブル平均をとる非定常解析の場合には、修正南部法は向かない事が分かった。

3. 原子蒸気噴流の数値シミュレーション

3.1 序論

非常に密度の高い原子蒸気流を発生させるには、通常るつぼにおかれた金属の表面に電子ビームを照射して真空中に噴出させることが考えられる。このような原子蒸気発生法は、非常に高い密度の原子蒸気を必要とする種々の蒸着過程、原子法レーザー同位体分離の蒸発過程等で用いられている。非常に強い蒸発の場合では、この流れは蒸発液面付近で連続体に近い領域になっており、真空中に膨張するにつれて遷移流から希薄流領域になる。また、このように非常に高い密度の原子蒸気を得るには、金属表面に入射する熱フラックスを増加させる必要がある。しかしながら、熱フラックスを増加すると蒸気フラックスが飽和する傾向が見られる。このように、真空中に噴出する蒸気の特性をとらえるためには、蒸発表面付近のクヌーセン層における蒸発原子の挙動を調べる必要がある。ここでは、クヌーセン層における蒸発原子を2次元DSMC法で数値シミュレーションすることにより、原子間の散乱により蒸発表面に戻る割合を評価すると共に真空中における蒸発特性について結果も述べる。

3.2 軸対称噴流と蒸発モデル

計算は軸対称性を考慮して2次元で行う。全体の噴流特性をとらえるための250x240mmのモデルおよび蒸発面近傍のクヌーセン層の性質を調べるために20x20mmのモデルの2つのケースについて行った。この2次元面は、軸対称噴流の一断面を表わしており、この面における面積要素は対称軸を中心とした3次元の環状の領域を代表していると考えられる。そのため対称軸からの距離によって面積要素内での模擬粒子の重みが変化することを考慮する必要がある。もしも、対称軸近くのセルと対称軸から遠く離れたセルに同じ数の模擬粒子が含まれていたならば、前者のセルの数密度の方が高くなる。各模擬粒子は3次元の速度成分を持っており、衝突および移動過程は3次元的に行なう。ただし、移動後は軸対称性を利用して2次元の計算領域に射影するものとする。模擬粒子は、対称軸上では鏡面反射を受け、他の境界を横切る時は抜取ることとした。蒸発面は、半径5mmの円形領域とし一定の表面温度 T_0 が保たれていると仮定する。噴流のクヌーセン数は、蒸発表面における平均自由行程と噴出領域の直径との比で定義する。サンプリングセルは、20x20mmのモデルでは20x20個、250x240mmのモデルでは25x24個用いた。衝突用のサブセルは、サンプリングセル内で場の量を自動的に評価しながら、平均自由行程程度またはそれ以下に設定して生成した。

蒸発表面のモデルは以下の通りである。速度分布からランダムに抽出された速度をもつ模擬粒子を蒸発領域にランダムに配置する。y-方向の速度成分は、次の分布からサンプルした。

$$f(v) = (\beta / \pi)^{1/2} v \exp(-\beta^2 v^2) \quad (4)$$

ここで、 $\beta = (2RT_0)^{1/2}$ である。x-およびz-方向の速度成分は、マックスウェル分布よりサンプルする。このモデルでは、液体表面温度と平衡状態に達している気体の、穴からyの正の方向に飛出してくる模擬粒子のみを蒸発分子としている。このモデルでは、粒子フラックス Φ は、

$$\Phi = n c / 4 \quad (5)$$

となる。ここでは、平均熱速度で ($= (8RT_0/\pi)^{1/2}$) である。式(5)は通常経験的な蒸発係数 α をかけて使用している。 $\alpha=1$ の場合は理想的な蒸発の場合で、蒸発係数 α はいったん蒸発した原子が蒸発面に戻った分を引き去って正味に蒸発する割合を表わしている。ここでの数値シミュレーションでは、数密度の正規化に式(5)を用いている。

衝突過程の計算には、20x20mmのモデルでBird法、250x240mmのモデルで修正南部法を用いた。また、剛体球近似を用いた。クヌーセン数は、 $Kn = 0.4, 0.04, 0.004$ の3ケースについて行なった。使用した模擬粒子数は250x240mmのケースで約 $7 \times 10^4 - 2 \times 10^5$ 個、20x20mmのケースで約 1.1×10^6 個である。計算時間は、VP-2600を使用してCPUが15時間から最大30時間程度である。計算は、場に全く粒子の無い状態から始めて、各点での速度、場中の模擬粒子数をモニターしながら定常に達したことを確認してサンプリングを開始した。

3.3 計算結果

Fig. 5, 6に $Kn=0.004, 0.4$ ($20 \times 20\text{mm}$ モデル) の場合の密度分布を示す。密度分布が等方的な $Kn=0.4$ の場合に比較して $Kn=0.004$ の場合は、対称軸方向に歪んでいるのが分る。Fig. 7, 8, 9は、 $250 \times 240\text{mm}$ モデルで対称軸から 15° 傾いた方向の平行、垂直及び全体温度の変化を示している。ここで温度は各方向の速度のゆらぎ（偏差）で定義する。クヌーセン数が大きい場合では、進行方向と垂直な成分は距離のほぼ比例して減少するが、平行成分は凍結されているのが分る。一方、クヌーセン数が減少するに従い、平行成分の凍結が徐々に遅れてくる様子が分る。Fig. 10, 11には、 $20 \times 20\text{mm}$ モデルで蒸発面からいったん真空中に飛出した原子が衝突により蒸発面に戻ってきた割合（蒸発した総粒子数で正規化している）を示している。この値を1から引いたものが、前に述べた蒸発係数 α に対応するものと考えられる。 $Kn=0.4$ の場合では約1%程度蒸発面に戻ってくるが、 $Kn=0.04, 0.004$ と減少するにつれその割合が約8%，15%と増加するのが分る。特に、クヌーセン数が0.04から0.004に減少して蒸発が激しくなるにつれ、散乱原子の戻る割合が2倍程度しか増加していないのが分かる。また、横軸方向の分布では、蒸発面を少しづれると $Kn=0.004$ の場合では10%から0.1%に急激に減少するのが分る。

4. 連続体に近い流れの数値シミュレーション

4.1 粒子法による計算例

DSMC法は、一般に希釈気体 (dilute gas) に適用できる方法である。この方法は、気体が希薄になるに従い正確になって行くが、連続体に近い流れからナビエーストokes方程式が成立する連続体領域の気体の流れもセルを十分小さくとると計算することが可能である。現在の計算機の能力では次に示す粘性流を再現する程度であるが、数値解析的不安定性がでないと複雑な境界条件の計算に適用できるなどの利点があり、今後より高いレイノルズ数の計算の可能性も大きいと思われる。

4.2 粘性シェア流れ

粘性流をDSMC法で計算する場合、レイノルズ数の評価等には粘性率を見積る必要がある。ここでは、上下反対方向に一定速度で移動している粘性シェア流れを用いて、経験式から評価した粘性率との比較検討を行った。この流れは非定常流れであるため、巨視的量を評価するにはアンサンブル平均をとる必要がある。サンプリングセルは80（縦） \times 28（横）に設定した。衝突サブセルはサンプリングセル中平均2 \times 2個程度である。気体の温度、圧力は、2次元キャビティ流れと同様にとった。また、速度は、マッハ数 $Ma=0.7$ に設定した。境界条件として、上下が鏡面反射、前後が周期的条件を設定した。衝突サブセル中の粒子数が25, 50, 100, 200個の場合について計算を行った。速度は、中心の縦一列についてアンサンブル平均を取った。アンサンブル平均の収束

状況を見るため、サブセル当り25個の模擬粒子を用いた場合の結果をFig. 12に示す。また、Fig. 13にサブセル当り200個の場合について10回アンサンブル平均を取った結果を示す。図中の点線で示したのは、粘性流の場合の厳密解

$$\frac{u}{u_\infty} = \operatorname{erf}\left(\frac{y}{2\sqrt{\nu t}}\right) \quad (6)$$

である。特に、サブセル当り200個の模擬粒子数のケースでは理論解と良く一致しているのが分る。

4.3 ポアズイユ流れ

2次元ポアズイユ流れを考える。サンプリングセルとしては、ほぼ平均自由行程と同程度の大きさのものを用い、160(縦)×40(横)とした。ここでは、サンプリングセルと衝突サブセルはほぼ一致している。上下境界には拡散反射、前後境界には周期的境界条件を設定した。粒子数は全体で約 1.92×10^6 個であり、衝突サブセル当り約300個である。クヌーセン数は $Kn = 6.25 \times 10^{-3}$ 、マッハ数は初期状態で $Ma = 0.1$ に設定した。レイノルズ数は最終状態で $Re = 24.2$ であった。縦一列のセルについて100回時間方向にサンプリングした結果をFig. 14に示す。図中の実線は中心の二つのセルで平均した最大中心速度の値を用いてナビエーストークス方程式の層流解を示している。この図より、DSMC法による数値計算結果はナビエーストークス方程式の解を良く再現していることが分った。

5. 謝辞

この研究を遂行するに当たり特にアルゴリズムの面で協力頂いた日本原子力研究所情報システムセンター横川三津夫氏またプログラミングの面で御助力頂いた燃料・材料工学部山本浩康氏に感謝の意を表します。

参考文献

- (1) E. Meiburg, Phys. Fluids 29, 3107 (1986).
- (2) G. A. Bird, Phys. Fluids 30, 364 (1987).
- (3) G. A. Bird, Molecular Gas Dynamics, Oxford (1976).
- (4) H. Babovsky, Math. Meth. Appl. Sci. 8, 223 (1986).
- (5) H. Kaburaki et al., Proc. of the 1st Int. Conf. on Supercomputing in Nuclear Applications, pp. 51 (1990).

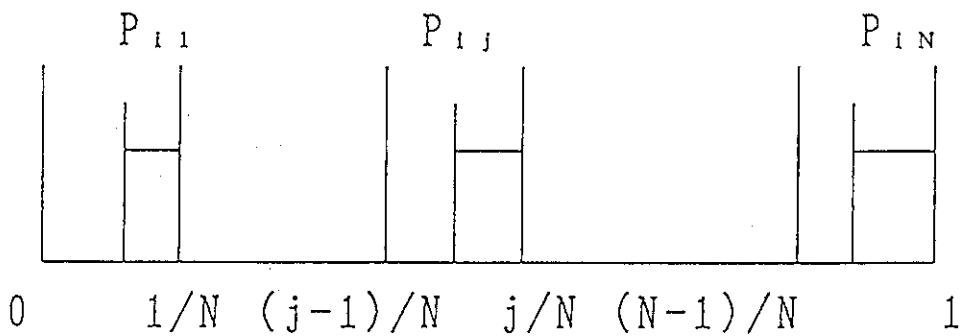


Fig.1 Distribution of the probabilities $P_{i,j}$ over the unit interval $[0, 1]$ for the modified Nambu method

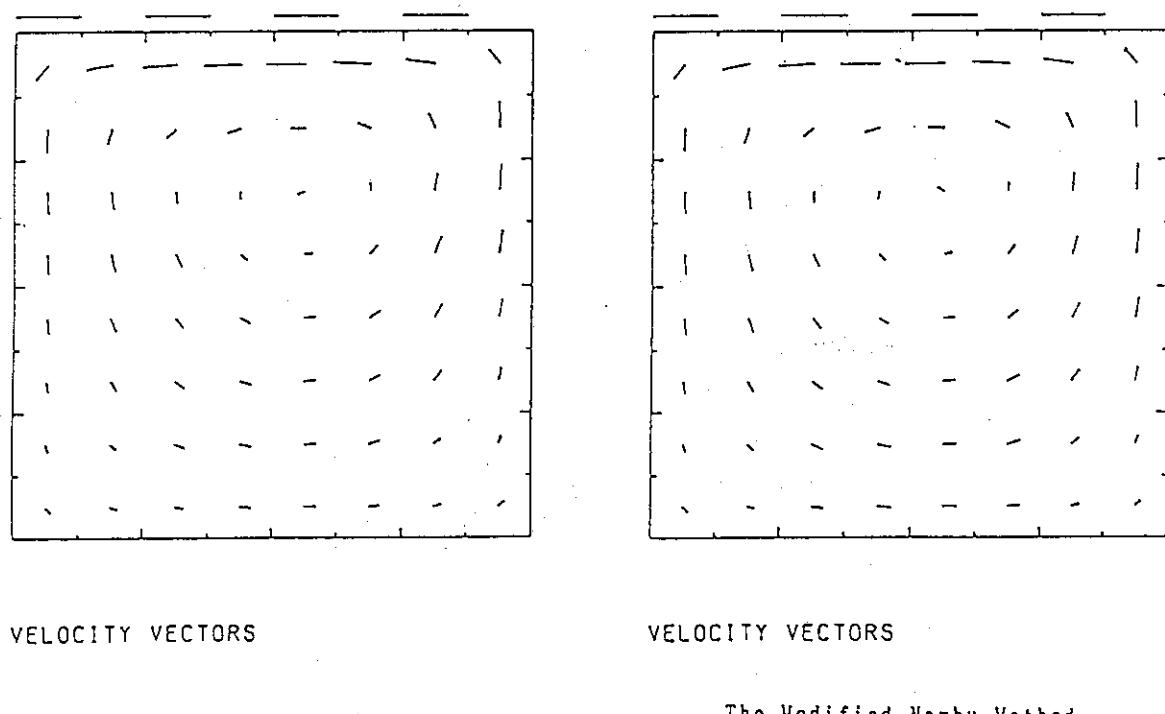


Fig.2 Streamlines for the standard cavity flow using the Bird and the modified Nambu method

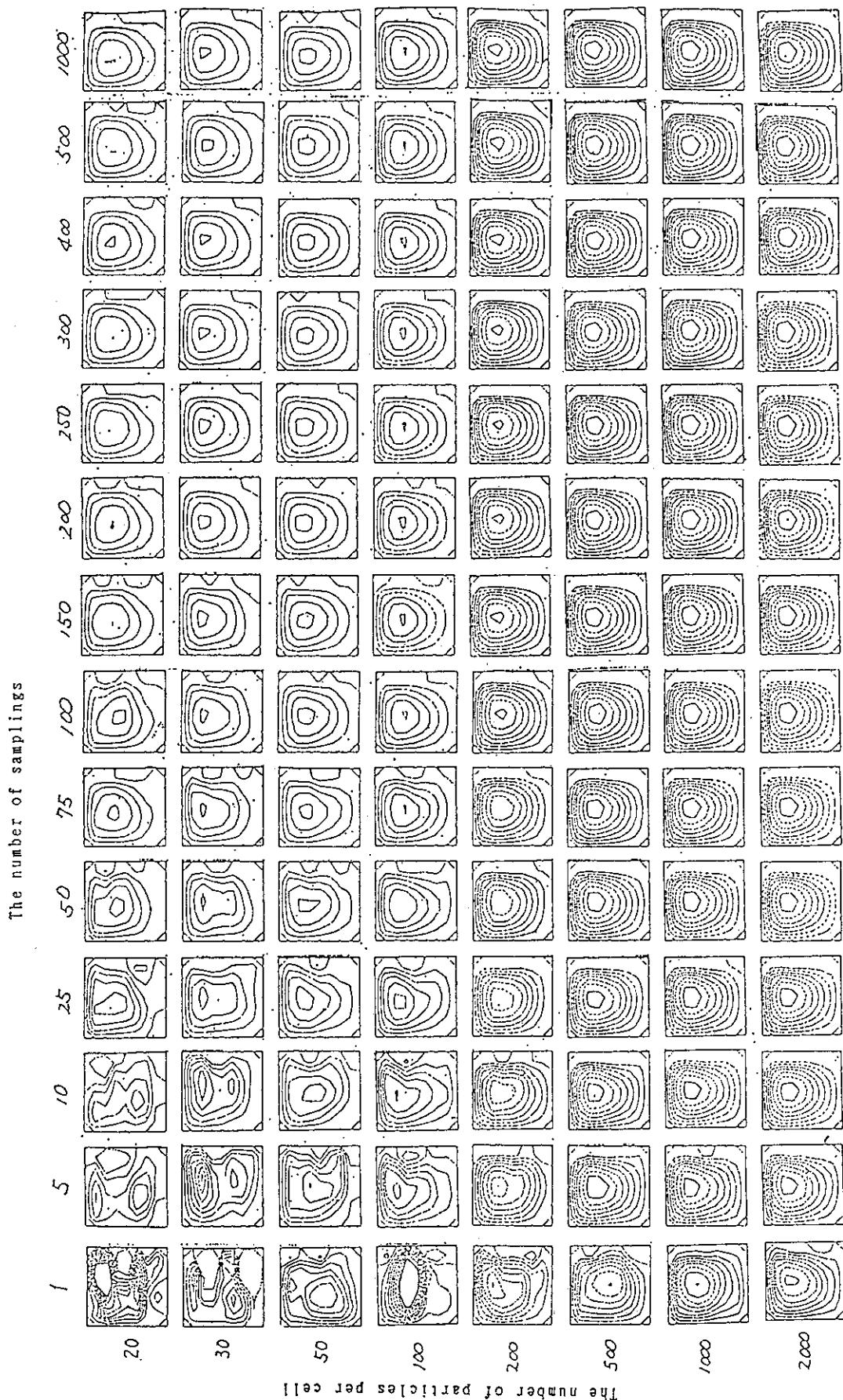


Fig. 3 The effects of sampling and particles (Bird method)

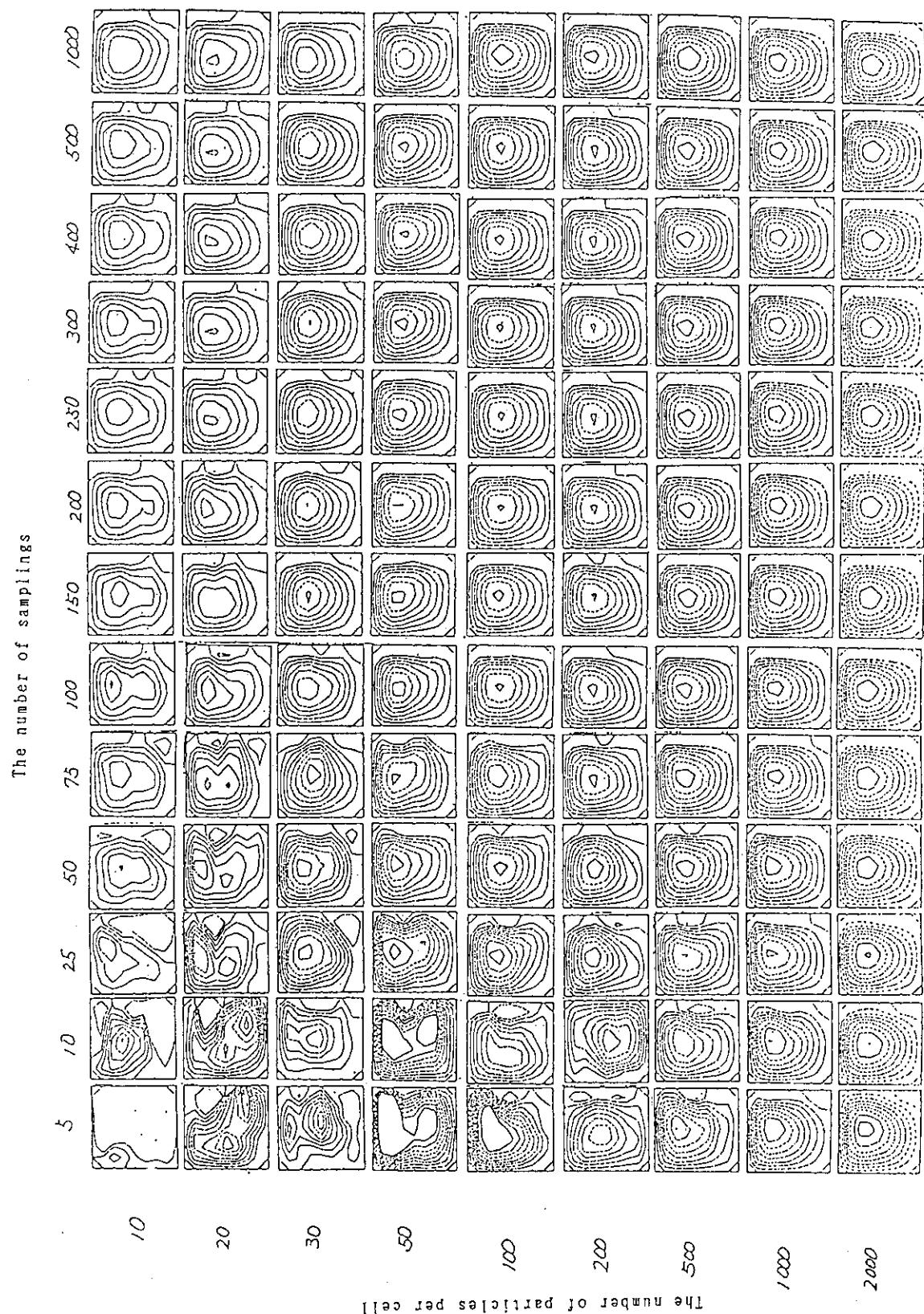


Fig. 4 The effects of sampling and particles (Modified Nambu method)

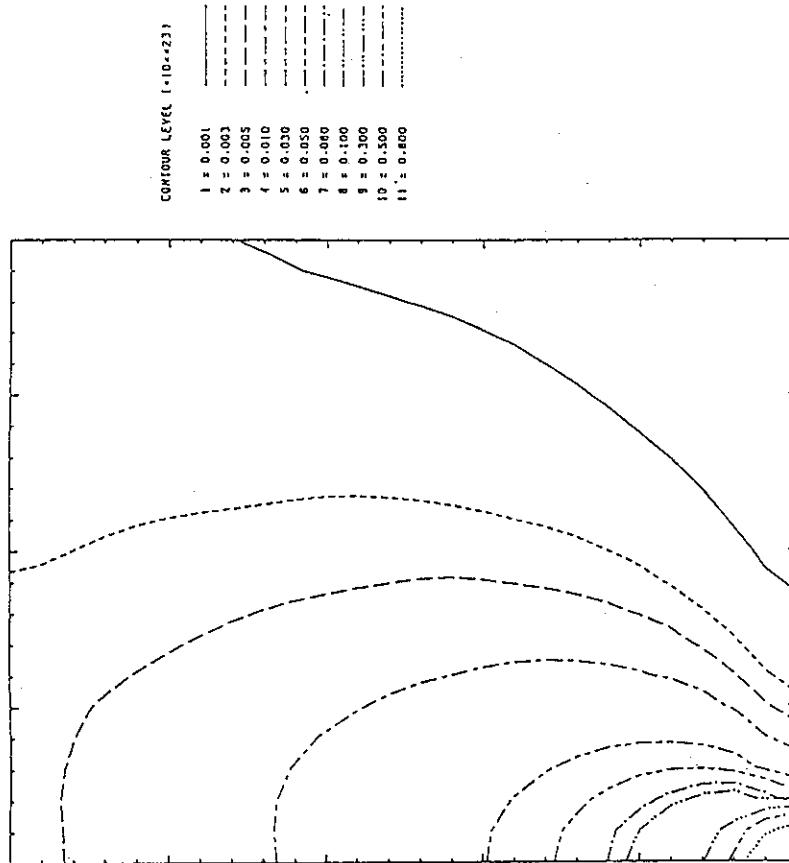


Fig. 6 Density distribution for $\text{Kn}=0.004$

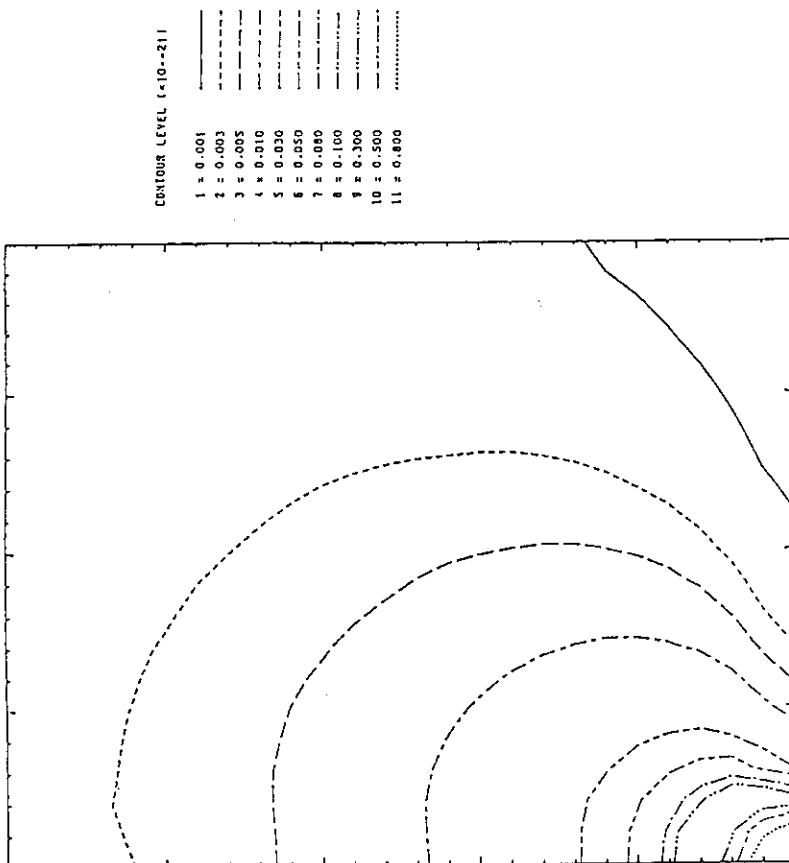
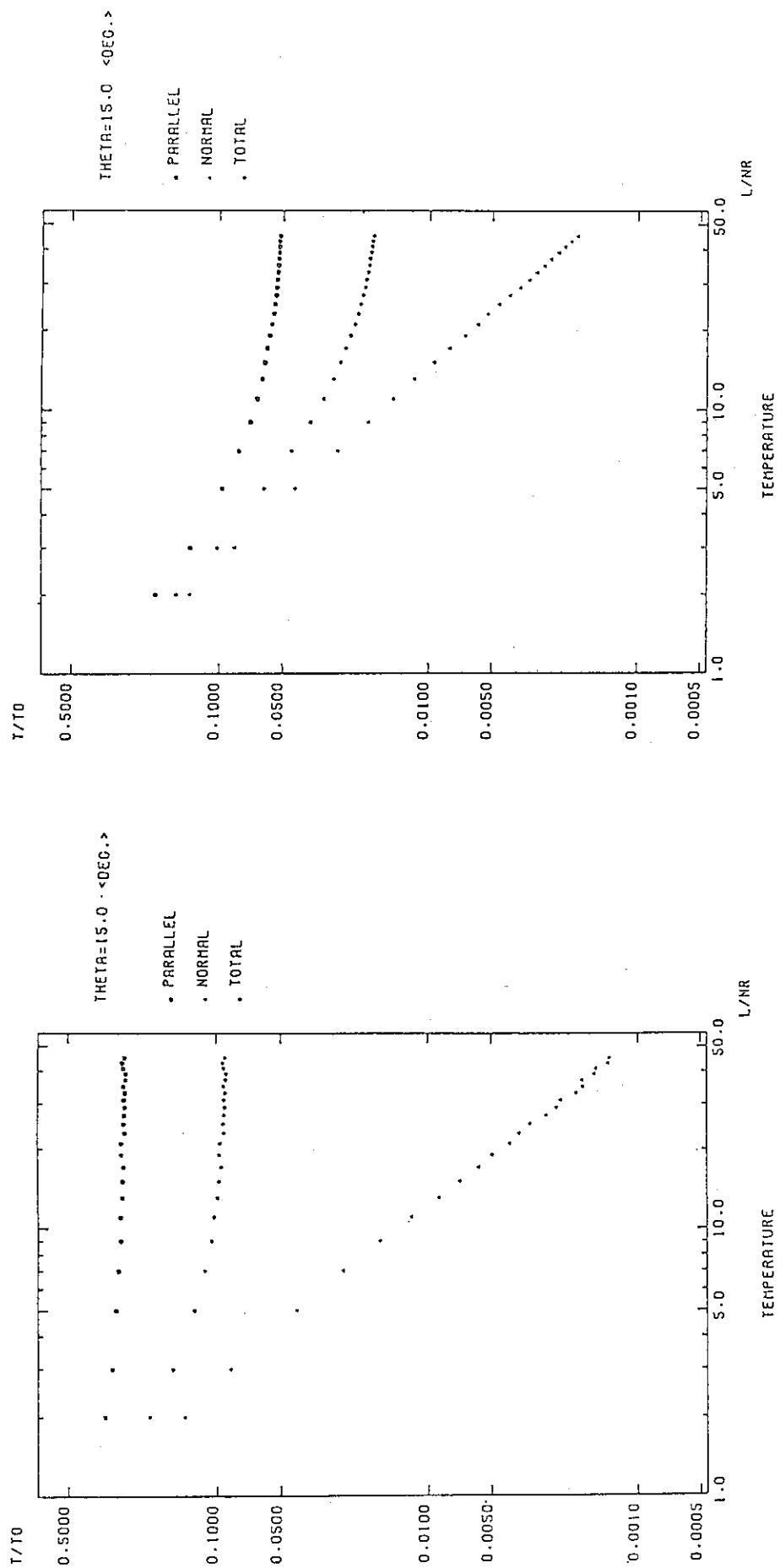


Fig. 5 Density distribution for $\text{Kn}=0.4$

Fig. 7 Temperature distribution for $\text{Kn}=0.4$ Fig. 8 Temperature distribution for $\text{Kn}=0.04$

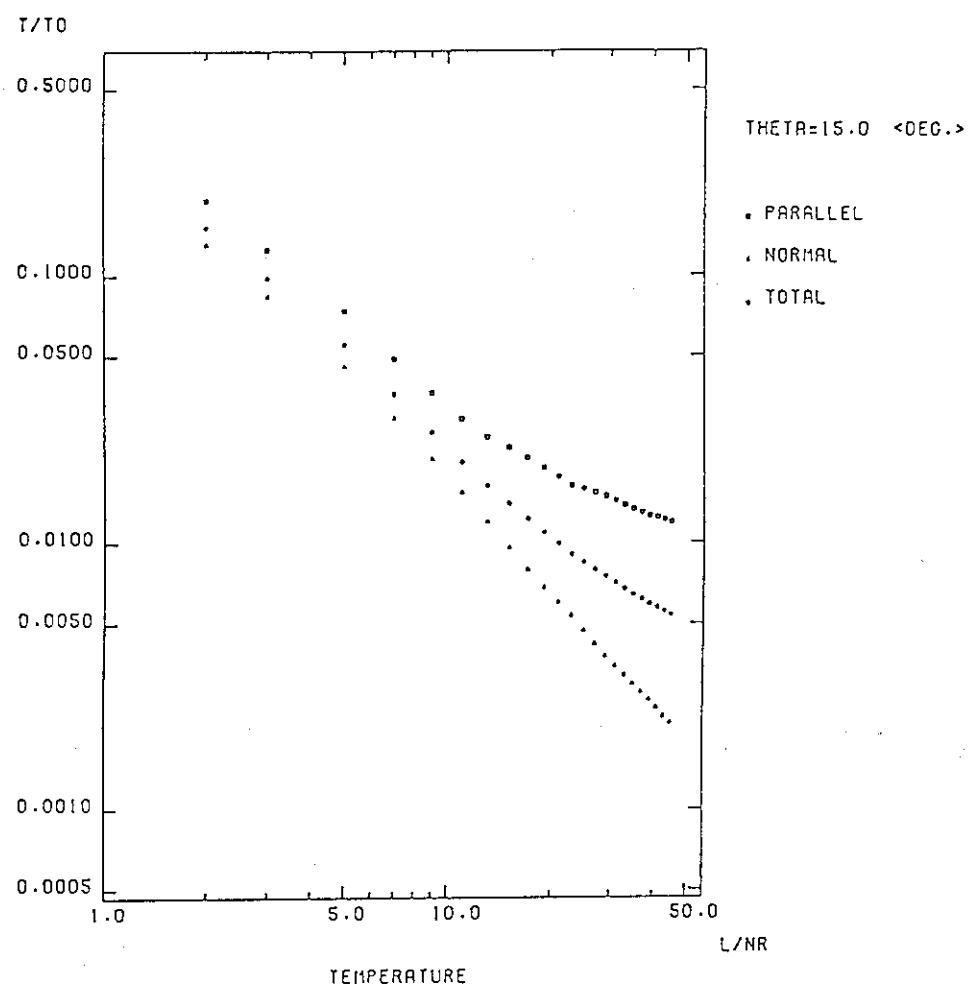


Fig. 9 Temperature distribution for $\text{Kn}=0.004$

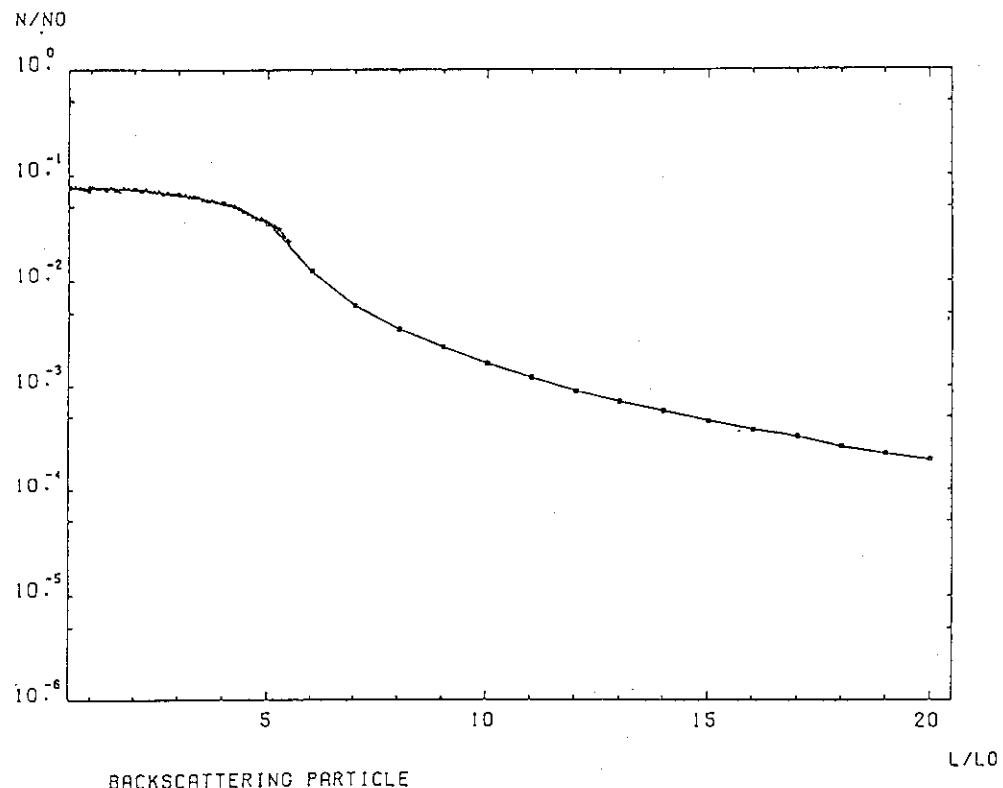


Fig.10 The surface return flux as a function of the distance from the symmetry axis ($Kn=0.04$)

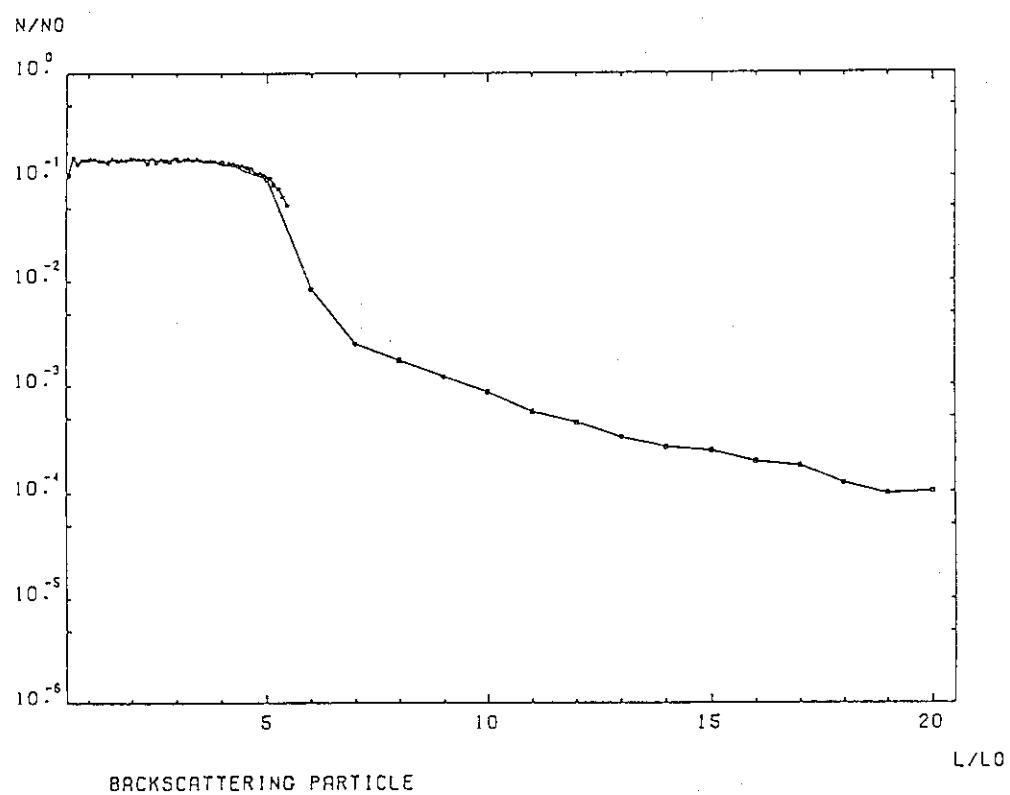
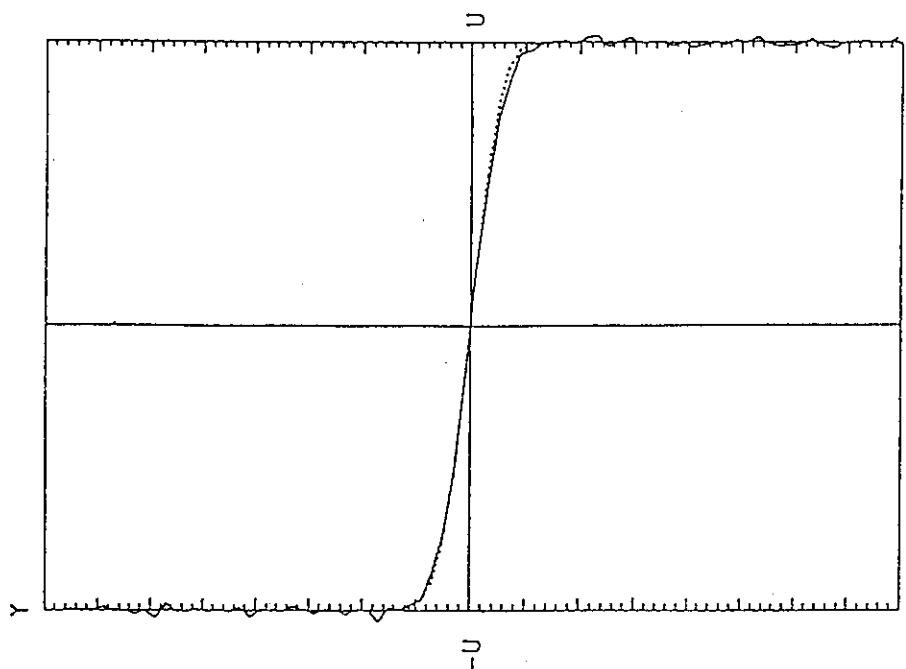


Fig.11 The surface return flux as a function of the distance from the symmetry axis ($Kn=0.004$)



VELOCITY DISTRIBUTION

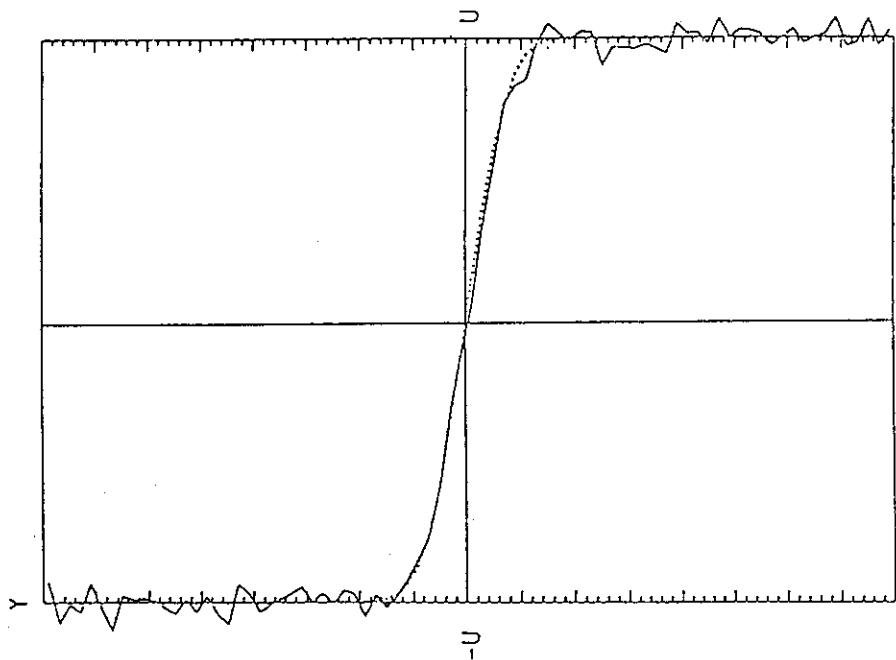


Fig. 12 Ensemble averaged result for the shear flow
(25 particles per subcell)

Fig. 13 Ensemble averaged result for the shear flow
(200 particles per subcell)

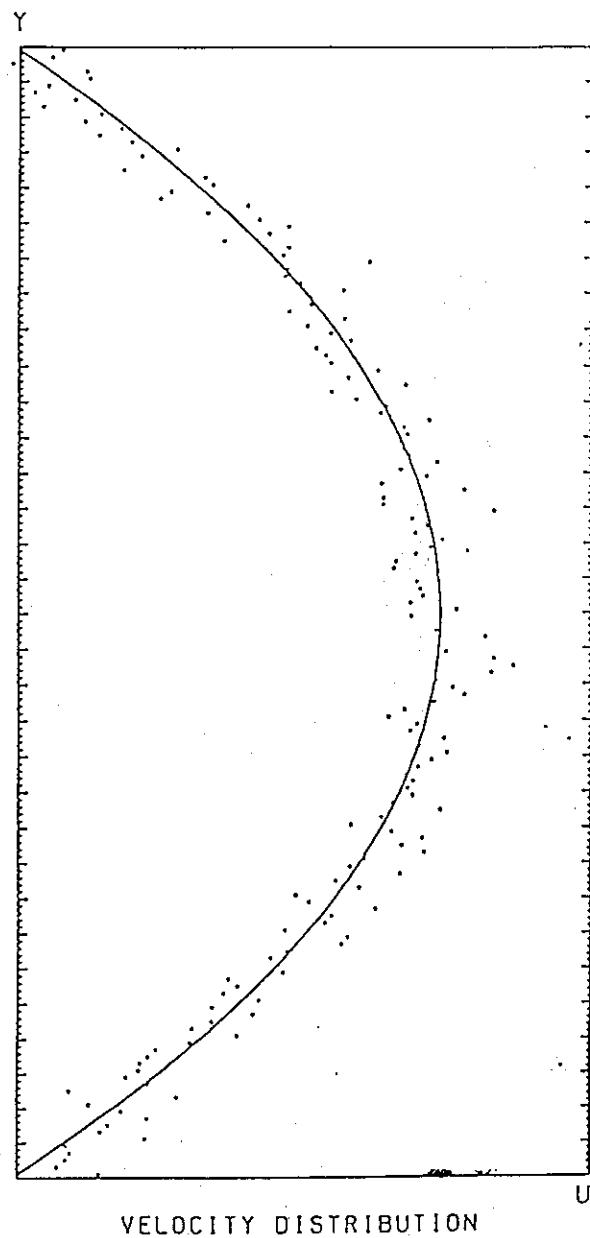


Fig.14 Time averaged result for the Poiseuille flow

3. 陽子リニアック設計の為のコンピューターコード

加藤 隆夫 (高エネルギー物理学研究所)

要旨：陽子リニアックを設計して、製作して、そして運転すると言う一連の流れの中で、その設計にコンピューターコードを利用する時の問題点を説明する。KEKにおける経験（40 MeV 陽子リニアックの設計と製作とチューニング及び 1 GeV 陽子リニアックの設計等）をふまえて、今まで使用した計算コードについて、目的により三種類に分類して、その内容と相互の関係を概説する。次に、最近の幾つかの仕事から、特に、スーパーコンピューターと三次元計算に関する以下の事例を紹介する。1) 三次元電磁場計算コード MAFIA のスーパーコンピューター用のチューニングについて、2) 加速管の電磁場計算へのMAFIA の応用例、3) 空間電荷効果を取り込んだビーム輸送計算コードの開発。

Computer Codes for Designing Proton Linear Accelerators

Takao Kato

(National Laboratory for High Energy Physics)

Summary: Computer codes for designing proton linear accelerators are discussed from the viewpoint of not only designing but also construction and operation of the linac. The codes are divided into three categories according to their purposes: 1) design code, 2) generation and simulation code, and 3) electric and magnetic fields calculation code. The role of each category is discussed on the basis of experience at KEK (the design of the 40-MeV proton linac and its construction and operation, and the design of the 1-GeV proton linac). We introduce our recent work relevant to three-dimensional calculation and supercomputer calculation: 1) tuning of MAFIA (three-dimensional electric and magnetic fields calculation code) for supercomputer, 2) examples of three-dimensional calculation of accelerating structures by MAFIA, 3) development of a beam transport code including space charge effects.

Keywords: Computer code, proton linear accelerator, design, three-dimensional calculation, supercomputer

1. はじめに

世界には、陽子リニアック設計用の数多くの優れたコードが存在しているが、新しいコードも続々と生まれている。新しいタイプのリニアックが発明されると、新しい設計コードが開発される。古いタイプであっても、新しくリニアックを建設する場合には、より最適なコードが新たに開発されるか、又は大幅な修正が加えられる。そして、コンピューター性能の目覚ましい向上により、以前は計算出来なかった事が簡単に計算出来るようになった結果、多くの新しいコードが書かれている。これらのコードは殆ど手に入れる事が出来る。コンピューターの互換性の欠如に起因する作業を終えてしまえば、好きなようにリニアックが設計できる。パソコンで簡単に設計ができてしまう。全く夢のような時代になってきた。さて、夢のようだと感じる裏には、昔は厳しかったという思いがある。一昔前は、良かれ悪しかれ自分で、細かい所まで非能率的にやらざるを得なかつたという事情があった。現在は恵まれ過ぎている点に、多少の怖さを感じる。シンクロスコープやネットワークアナライザがデジタル化と同時に自動化され、訳がわからない内に正しいとされる答を出してしまった時に感じた怖さである。

将来、コンピューターの性能と使い勝手が飛躍的に高まると予想される。これは誠に歓迎すべき事であって、その性能を生かすような新しいコードが開発されると期待される。ここでは現状を見渡しつつ、陽子リニアックを設計して、製作して、そして運転すると言う一連の流れの中で、その設計にコンピューターコードを使う時のお話をしたいと考える。従って、今まで使用した計算コードについて、KEKにおける経験をふまえて概説する。これは、世の中に存在するコードについてのレビュー的なレポートではなく、小さい陽子リニアックの設計と製作とチューニング（40 MeV）と、大きい陽子リニアックの設計（1 GeV）の経験を率直に反省する内容になっており、いささか本研究会の主旨から逸脱している点をお詫びしておきたい。

2. 陽子リニアックの諸事情とコンピューターコード

2.1 陽子は中々速くならない

陽子はその質量が電子の約 1800 倍である。この事が、電子リニアックに比べて、陽子リニアックの構造を複雑にしている。

一般に加速構造（加速管）の単位長さ L は、加速に使用する高周波の一周期の間に粒子が通過する距離になる。

$$L = v t = \frac{v c}{c f} = \beta \lambda$$

ここで、 v は粒子速度、 t は高周波周期、 f は周波数、 c は光速、 $\beta=v/c$ 、 $\lambda=c/f$ (波長) を表している。静止質量 W_0 、全エネルギー W と β との関係は

$$W = \frac{W_0}{\sqrt{1 - \beta^2}}$$

静止質量は、 $W_0 = 938.28 \text{ MeV}$ (陽子)、 $W_0 = 0.511 \text{ MeV}$ (電子) であり、軽い電子のスピードは、少しの加速によりすぐに光速に近くなり、それ以後はエネルギーは増えてもそのスピードは殆ど一定といえる。従って、加速に使用する加速管の形を、加速効率等の理由により変える必要が無い。これに対して、実用上の陽子リニアックのエネルギー領域では、陽子のスピードは大きく変化して、一定にはならない。(3 MeV では $\beta=0.08$, 1 GeV で $\beta=0.88$) この場合、あるエネルギー範囲で最適な加速管構造は、その他のエネルギー領域では、加速効率の減少とか、製造上の難しさの増大等の理由により、使用しにくくなる。従って、1 GeV 程度まで加速する陽子リニアックでは、2 ~ 3 種類の異なった加速管を使うのが普通であり、このあたりに設計者の考え方方が強く反映される余地がある。計画中の大型ハドロン用 1 GeV 陽子リニアック⁽¹⁾は Fig.1 に示す構成になっている。

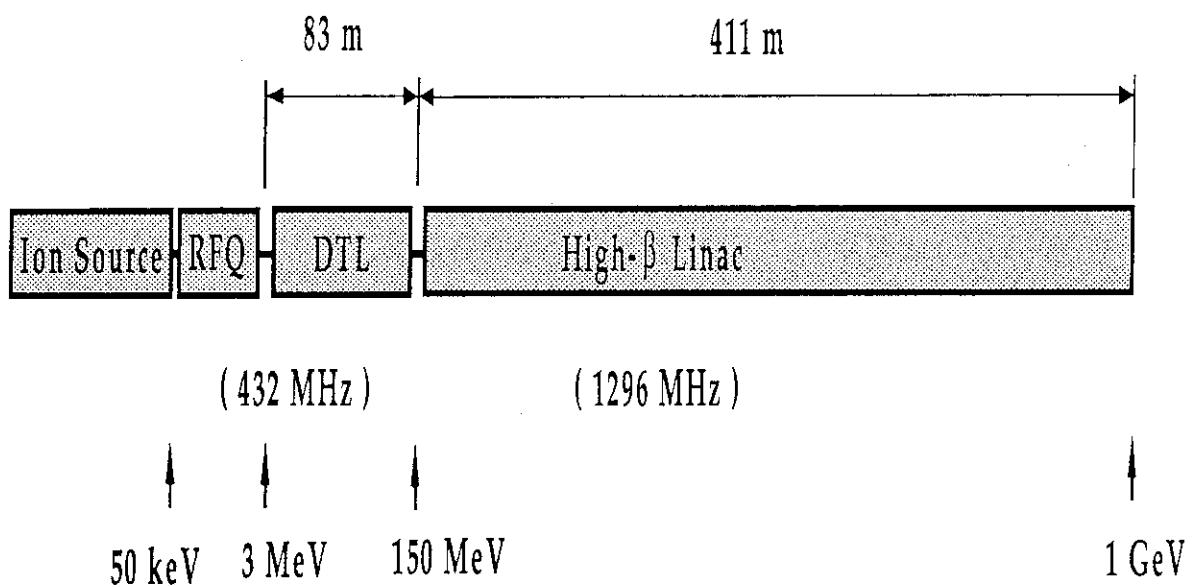


Fig. 1 The scheme of the 1-GeV proton linac.

イオン源に続いて、RFQ (radio frequency quadrupole)、DTL (drift tube linac)、そして CCL (coupled cavity linac) と全く性質の違う三種類の加速管で加速する方式である。

2.2 陽子リニアックの運転の難しさ

陽子リニアックの加速管では、加速に必要な電場の強さと位相が定められており、性質の良いビームを得る為の、それらの値の設定の許容範囲は以外と小さい。特に高エネルギーで大強度の陽子リニアックに対しては、高度の高周波技術により初めて対応可能な精度が要求される。一方、電子リニアックにおいては、極論を言えば、ある一部の加速管に電力を供給しなくとも、ビームは全体的には加速されるのである。即ち、ある一部分の故障を別の部分で補う事が可能である。ところが、陽子リニアックは、それぞれの部分が特定のエネルギーに対応している為に、ある部分の故障を他の部分で補う事が出来ない。これは運転する上では実に厳しい条件であり、高周波源の重要さを強調する所以である。

次に、重要でありながら、電子リニアックと陽子リニアックの人の重なりが少ない事もあって、以外と意識されないのは、加速途中の粒子の漏洩によるリニアックの放射化の問題である。特に、大強度高エネルギーの陽子リニアックでは、漏洩陽子による放射化のレベルが、実質的な運転時間を決める事が予想されている。高エネルギー領域においてビームロスが何故おこるのかは断定できないが、加速電場の強さと位相のエラーが関係している事が推定される。ビームロスは、エラーの複雑な加算的効果による為に、ある部分のチューニングが良いか悪いかの判定が難しくなってしまう。この部分の解明の為には、作られたリニアックの諸パラメーターが設計値に近く、運転の時のパラメーターがコンピューターシミュレーション用のパラメーターと良く一致しており、しかもシミュレーションにビームの振る舞いを正確に反映出来る事が要求される。

以上の理由により、陽子リニアックには精度の高いチューニングが必要であると言える。実際、設計と製作と運転方法とが三位一体となって、初めて実質的に精度の高いチューニングが達成されるのであるから、これらに使われるコンピューターコードの重要性は、非常に高い。

2.3 全体を考える

従って、陽子リニアックの設計では、高周波源とチューニングまで含めた全システムの設定が非常に重要となる。そのリニアックの目的と要求性能により、それに最適なリニアックが考えられる。我々は、一般に指標となる数値（電場、シャントインピーダンス等）だけを取り出して判断を下す傾向がある。便利で即座にできるから止むを得ない点もある。数字の裏に見え隠れする様々な事情（何を前提条件として、何を重点課題と考え、何をどの程度最適化しているか）は、夫々の場合により異なる事を踏まえて、個々のリニアックについて考察する事が重要と考えている。

例えば、電力が殆どコスト無しに供給されるとすれば、加速管設計におけるシャントインピーダンスの最適化の重要性は半減するであろう。同じく、ビームロ

ーディングが極めて大きいリニアックでも似た状況になるであろう。加速電流とエネルギーが小さいリニアックでは、シングルセルを並べて、夫々を固体高周波増幅器でドライブすれば魅力的なリニアックが出来るかもしれない。高エネルギー大強度のリニアックでビームロスによる放射化が問題となるとすれば、シャントインピーダンスを犠牲にしてもビーム穴を最大限大きくするだろう。しかし、低エネルギー小強度のリニアックでは、狙いをシャントインピーダンスの最適化に置くかもしれない。

全般の考察に際し重要なのは、使用できる技術レベルを的確に把握しておく事であろう。例えば、運転時のチューニングの難しさは、優れたモニターと優れた高周波コントロール系と全体チューニングの適正な理論があれば、ある程度解消される問題である。高周波関係の技術のレベルは、理論的限界と一般的な技術水準、そしてリニアックの大電力高周波装置の環境内での制約を踏まえて把握できる問題と言えよう。例えば、精度1%という数字は、静かな実験室とMWクラスの現場環境ではその意味する重みが違う。

注) *シャントインピーダンス Z は、加速管の加速効率を表す数値であり、平均加速電場を E 、加速管の長さを L 、励振高周波電力を P とすれば、

$$Z = \frac{E^2}{P/L}$$

と定義する。実効シャントインピーダンス ZT^2 をいう場合もある。

T は transit time factor と呼ばれ、加速管の構造で決まる定数である。

*加速管のユニットセルによる加速エネルギーは次式で与えられる。

$$\Delta W = E T L \cos \phi$$

ϕ は、陽子の加速高周波位相を表す。このエネルギー増加が陽子の速度を増加させ、この増加に見合うようにユニットセルの長さを増加させている。従って、あるユニットセルで得られるエネルギーの増加は予め

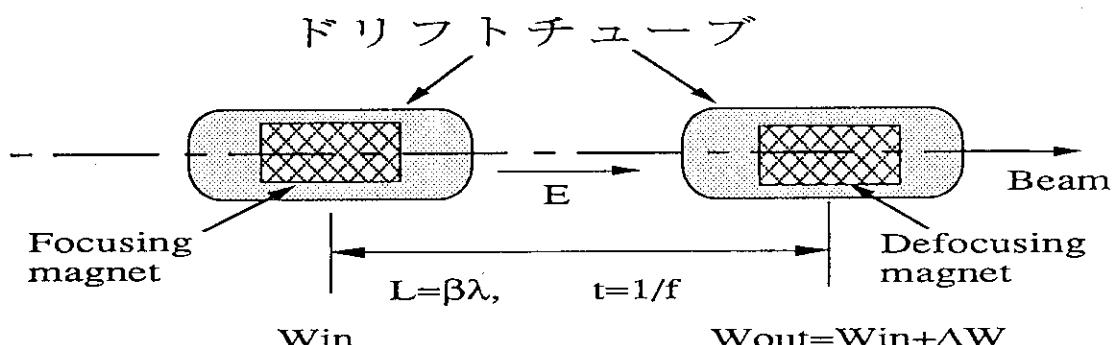


Fig. 2 Unit cell for a DTL accelerating structure.

決められた値でなくてはいけない。T と L は構造的に予め決まる値なので、加速電場 E と加速位相 φ の正しい値がチューニングで要求される。

*ビームローディングは、ビーム加速する時にエネルギーが消費される事を一般にいう。加速の反作用として、加速電場の強さと位相が変化してしまうので、対策が必要となる。

*エミッタンスはビームの広がりと言える。縦方向のエミッタンスは、エネルギーの広がりとビームの進行方向の長さに関係する。

*アクセプタンスは、その加速管が保持できる最大エミッタンスを言う。

2.4 便利だが怖いコンピュータコード

コンピュータコードを使う時、その便利さの故に、案外この辺の事情を看過してしまう事がある。例を二つあげよう。電磁場計算コードを使う時の目的は、高周波性能の良い加速空洞を設計する事にあるが、その目的が一人歩きを始めてしまう事がある。例えば、シャントインピーダンスが極めて優れている点が強調される加速空洞は、他の何らかの特性を犠牲にしている場合がある。アクセプタンスが小さくなっているとか、有り難くない他のモードが沢山あるといった具合である。実際に製作してみると、製作上の制約が極めて大きい場合もある。もう一つの例は、コードの入力データとして使うイオン源のエミッタンスの数値である。瞬間最大風速データと平均風速データの違い、そして、それ自体の値と加速器に応用する場合の違い（具体的には測定の位置、方法）など、加速器側では注意すべき点が多い。ここは、加速器の出発点なので特に重要である。

リニアック関係のコンピュータコードを使う時に注意すべき点をもう一点あげておこう。最近のコードの規模が大きい点を考えると止むを得ないと思うのだが、コードの中を具体的に知らずに計算する場合が多い事である。例えば、電磁場計算コードの内容を殆ど知らずに結果を利用している。勿論、そのコードを使い始める時は、予め解析的にわかっている問題を計算して、精度や適用範囲等のチェックを行なうが、そのコードが作られた時の目的から外れた事を計算しようとすると、計算エラーが起きたり、結果が物理的に間違っていたりする。これは、注意していないと見逃す事がある点が怖い。加速器業界で流布しているコードは、その生い立ちから考えてみても、汎用的な書き方がされていない場合がある。汎用的な書き方をする程の時間がおそらく無かったのであろう。汎用的に書き下すかわりに、開発時の要請を満たすように簡略化して書くのである。

リニアックの寸法を決めるコードでも事情は同じである。そのコードの開発時のリニアックの構造を計算しやすいように書かれている。自分が計算しようとしている事をそのコードが正しく計算しているのかは、コードを読まないと判断がつきにくい事が多い。それに、幾多の改良を経て使いやすくなってはいても、新しいリニアックを設計するには適用出来ない点や不便な箇所が出て来る。その場合には、

使いやすい形に修正するか、あるいは新しくコードを書かなければいけない事もある。又、空間電荷効果を考慮する時、そのクロスチェックの方法が無い場合など、結果の解釈が難しい場合がある。

これまでの議論の結論は、コンピューターコードは重要な要素技術であり、その目覚ましい発展は力強い限りであるが、これらを充分活用するには、リニアックの全体像（設計、製作、運転）まで含めた使用前後の諸検討が、益々重要になって来たという事である。

3. 陽子リニアック設計の為のコンピュータコード

陽子リニアック設計の為のコンピュータコードは、その目的により大きく三種類に分けられる。

第一は、ビームダイナミックスの基本的な関係を使って、ビームと加速管の最適なパラメーターを決める為のコードである（design code）。ここでは、リニアックにおけるビーム加速の諸性質を表す幾つかのパラメーターの最適な組み合わせが、要求仕様に従って考察される。

第二のコードは、第一の過程で決めたパラメーターを入力として使って、リニアック自体の基本的な寸法等のパラメーターを決める。上に述べた二種類のコードは、ビームシミュレーションを通じて、相互にチェックされて、全体の最適化が行なわれる。従って、第二のコードは、リニアックの基本寸法等を計算する部分（generation code）と、設定されたリニアックの中でのビームの振る舞いを計算する部分（simulation code）から構成されるのが、普通である。

第三は、リニアックを構成する機器（加速管や磁石）の設計を行なう為のコードである（電磁場計算コード）。ここでは、前項において設計された基本寸法を持つ加速空洞や磁石の諸特性を計算するが、一方では、ここで計算された高周波特性は、第二のコードの入力パラメーターとなっている場合があるので、第二と第三のコードの間でも何回かの反復計算が必要となる。この関係をFig. 3 に示す。以下に、これら三種類のコードの特徴をまとめてみよう。

3.1 第一種 (design code)

リニアックの基本的なパラメーターは次の諸量と考えられる。

まずビームに関しては、エネルギー W 、加速電流 I 、エミッタンス ϵ 、バンチの半径 a, b 、内部エネルギー U 。加速管に関しては、周波数 f 、加速電場の強さ E と位相 ϕ 、収束系の位相進み σ 、アクセプタンス A 、最大加速電流 I_{max} 、transit time factor T 、シャントインピーダンス Z 、最大表面電場 E_p 、ビーム電力 P_b 、励振電力 P_c 。

注) 上記の諸量は全てが独立変数では無い。また下線を引いてある諸

Design Code

Generation Code

Simulation Code

JAERI-M 92-054

f: 周波数
W: エネルギー
I: 電流
 ϵ_t, ϵ_L : エミッタrans
 σ_t, σ_L : 位相進み
 $A_p A_L$: アクセプタrans
E: 電場
B': 磁場勾配
Imax: 最大透過電流
Pb: ピーム電力
Pc: 励振電力
Zave: 平均シヤントン
インピーダンス

f
Ein,+ Win
 $\phi_{in,+}$
 $\sigma_t,+$
T,+ Z,+
a: ピーム穴半径
Input parameters
Parameters for j-th cell

$\epsilon_{t,in}$
 $\epsilon_{L,in}$
I
Nin: 粒子数
各種エラー

$\epsilon_{t,out}$
 $\epsilon_{L,out}$
I
Nout
各種エラー

t = transverse, L = longitudinal
in = injection value
j = value for j-th cell
+ = variation through linac

電磁場計算コード

a その他の形状

Ep: 最大表面電場

T: transit time factor

Z: シャントインピーダンス

L
f
W
a

Fig.3 Parameters of the codes for designing proton linac.

量は横方向 (transverse) と縦方向 (longitudinal) の二種類がある。現実のリニアックで最も重要なパラメーターの一つは、duty factor = pulse length x repetition rate である。この表舞台に登場しない量が、Pb とPcと共に、リニアック全体の性格を決めている。

ここに上げた諸量は、互いに幾つかの関係式で結ばれており、リニアックの中でどのように変化させるべきかをまず決めなければいけない。その時の経験的な注意がいくつかあるので紹介しよう。

A) あくまでビームだけで？

ビームの性質が重要としても、それをどこまで追及するかにより、設計が変わってくる。高周波源、冷却、放電限界、そしてコストなどと、どの程度まで妥協するか。例えば、コストを下げる為に、高周波源ユニットにあわせて電場を設定して、ユニットタンクの長さを決めて電力効率をあげる方法もある。更に、実験室のクリーンな条件のもとで見いだした放電限界に対するデータを、リニアックという悪条件下においてどこまで活用したらよいのか、加速途中のビーム損失をどの程度考慮すべきかなど解決すべき問題が多い。

B) ビームの何が重要か？

ビームの諸性質の中で重要なのは、最大加速電流の大きさか、エネルギーの広がりか、横方向エミッタスか。これらは、そのリニアックの要求仕様によりその優先順位が変わると同時に、全体の中での比重も変わる。

C) 提案されている関係式はどこまで適用すべきか？

例えば、エミッタスの増加に関する関係式がある。一体、この関係をどのエネルギー領域まで持続させればいいのか。リニアック全体にある関係式を適用しようとすれば、必要な加速電場が非現実的な大きさになる事がある。又、この関係式は入射ビームパラメーターに対する依存性が大きい。はたして入射ビームをどの程度まで制御できると予測しておけばいいのか。そして、製作上の精度限界に由来するエミッタスの増加の方がはるかに大きい場合の対策は等、ここにも問題が多い。

D) 細かな定義を？

例えば、単に"無酸素銅"という時に、数多くある無酸素銅の違いを全て捨て去ってしまい、問題点の所在を複雑にしてしまうような事例を指す。放電限界の何倍という時に、特に RFQあたりではある時期、数値だけが世界を駆け巡っていた感があった。どの様な条件のもとでどの程度の内容を誰がどうしたか？の

三点を明確にする事が肝要であろう。

E) 結論

以上の事から、リニアック全体を通して、明快に一つの理論から設計出来る現状では無いという結論に達する。一つの設計を簡単に実験検証出来ない事、いわゆるビームシミュレーションは進歩しているが、まだ完璧ではないと思われる事、そして最後に、リニアックは、ビームという判断基準により結果が試される事を考慮して、現実的な条件のもとに妥協しながら設計する事になる。その場合、大型の計画になればなるほど、コストの最小化が重要視される。

大強度陽子リニアックでは、大強度に由来する空間電荷効果を第一に処理しなければいけない。空間電荷効果は、低エネルギー領域でその効果が著しいから、RFQ と DTL からなる入射部分を考える。入射部分の重要性は昔も今も変わらないが、以前は、縦方向の優れた入射用加速器が無かった。仕方無しに横方向のマッチングだけを考えて、コッククロフトで 750 keV まで加速して、DTL の捕獲効率を上げる為にバンチャードを通過してから DTL へ入射させた。その場合のビームの縦方向のエミッタスを Fig. 4 に示す。近年 RFQ なる新型加速器が発明されて、これをコッククロフトとバンチャードの変わりに使うと、エミッタスは Fig. 5 のようになる。Fig. 4 では縦方向のエミッタスは、その定義と実体がかけ離れてしまう。従って、入射部において、縦方向と横方向のエミッタスの大きさがある関係にしよう (Equipartitioning) という議論⁽²⁾ がやりにくい。RFQ の発明により、初めて、縦方向のマッチングもうまくできるようになった。

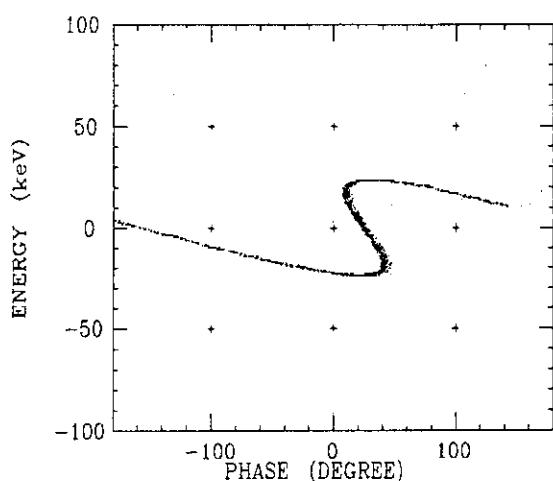


Fig. 4 Injected beam with a buncher.

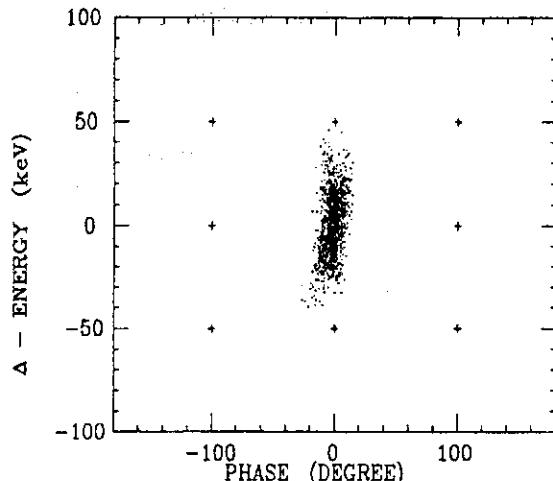


Fig. 5 Injected beam with an RFQ.

従って、design code に取り入れる主な関係は以下のような諸関係である（DTL の場合）。

*入射点のequipartitioning を考慮する。

横方向と縦方向の位相振動のエネルギーを等分配にして、しかもリニアックの縦と横の収束系にマッチングさせる。equipartitioning の基本式を示すと

$$\epsilon_t = \frac{\sigma_t a^2}{2\beta\lambda} \quad \epsilon_L = \frac{\sigma_L b^2}{2\beta\lambda} \quad \frac{\epsilon_L}{\epsilon_t} = \frac{\sigma_t}{\sigma_L} = \frac{b}{a}$$

位相進み σ とバンチの長さは空間電荷効果を通して関係しているので、簡単な見かけ以上に複雑である。この式は、波風が立ちにくくように滑らかに入射して、しかも横と縦の微小振動のエネルギーのバランスをとる様子を表している。

*equipartitioning を加速につれてどのように変化させるか決める。

縦横方向の位相進みを加速につれてどのように変化させるか決める。

ビームの内部エネルギーに由来するエミッタンスの増加を考慮する。

*エミッタンスとアクセプタンスの比をある値に設定する。

*電力の大きさを考慮する。

*縦横の最大透過電流を考える。

RFQ は、収束と加速を電場で行なう為に、縦横の収束力の変化は一元的にできて便利とも言えるし、自由度が少なくて不便とも言える。特に、イオン源からの DC ビームをうまくバンチさせる事が最重要課題なので、バンチング部分の最適化の方法が何種類か考えられている。DTL 以後のリニアックは、加速と横方向の収束は独立しているので自由度は高い。

以上を眞面目に考えると、リニアックへの入射時のパラメーターのセットと、それらの変化の仕方を導く事ができ、どの項目を優先させるかにより多種類のセットが考えられる。その中でどれが最善であるかは、第二種コードにより、リニアックを設定して、ビームシミュレーションを行なって、判断する。

3.2 第二種 (generation code, simulation code)

加速管の構造が異なる為に、陽子リニアックのすべてにわたって適用できるコードは存在しない。従って、加速管の構造毎に多くのコードが存在する。各研究所毎に異なったコードが存在する場合もあり、新しく加速器が建設される場合には、その加速器の特徴をより良く反映するコードが開発される状況である。しかしながら、世界的に流布している基本的なコードは存在しており、加速器用の

ソフトウェアの開発と整備に精力的なロスアラモス製 (LANL) のコードが多く、敬意を表せざるを得ない状況である。その中で代表的なコードは、RFQ用の PARMTEQ と DTL用の PARMILA⁽³⁾ である。

PARMTEQ と PARMILA ともそのGENERATION 部はすでにいくつかの実機製作によりその精度は確かめられているといえよう。SIMULATION 部も良くできており、殆どのシミュレーションには十分といえる。ただ、古い版の空間電荷効果を取り扱う部分は、軸対称なディスクモデルにより加速ギャップの中心位置で効果を取り入れており、その精度チェックが必要と言えた。

CCL (coupled cavity linac) 用のコードは、世界共通のものではなく、各研究所で独自のコードを開発しながら将来計画に備えている。KEKで開発されたコード (PROEND) は次のような特色をもっている。

1. $\pi/2$ モードの加速管を計算する。
2. 定速度型加速管と可変速度型加速管が比較できる。
3. 横方向の収束系を含んだ全系を自動的に計算する。
4. シミュレーション部は、実際のチューニングにおける測定可能量との対応を考慮している。

現実には、このコードをもとに製作されたリニアックは存在しないので、今後の検証が重要となる。

近年、エミッターンスの増大の機構の解明が進み、陽子リニアックの低エネルギー部の重要性が一段と認識されるに至っている。そこでは、異なった種類の加速管の間のマッチングとその間のビーム輸送が重要問題となる。イオン源とRFQ の間、そしてRFQ とDTL の間のビームの振る舞いを調べる為に、空間電荷効果を直接的に取り入れたビーム輸送用のコード (LEBT) を開発した。その特徴を以下に述べると、

1. 空間電荷効果を個々の粒子間で計算する。
2. ルンゲクッタ法により微分方程式を解く。
3. スーパーコンピューター用にコード化する。

問題の設定の仕方に異論があるかもしれないが、実際に走らせた結果は得るもの多かった。具体例は次章に述べる。

3.3 第三種（電磁場解析コード）

加速管の電磁場解析の重要性は益々増している。それは、最近、複雑な構造が増えてきた事と、要求される精度が増している為である。従来、加速管は軸対

称性を持つ構造のタイプが多かったので、軸対称な電磁場モードを計算するSUPERFISH⁽⁴⁾が多用された。計算精度等も実用上充分である。ところが、現実には非対称な構造を持つ製造物が沢山あって（高周波カップラー部、加速管の結合孔、DTL のポストカップラー、チューナーなど）、そこでは、trial and error で製作を行なっていた。3次元電磁場解析コードの登場がこの分野では非常に期待されていたのである。総合的な3次元電磁場解析コード MAFIA⁽⁵⁾が手軽に利用出来るようになって、この分野の省力化等の発展は非常なものである。しかしながら、現在のKEKにおける利用環境は、計算速度と利用出来る記憶容量の点が不充分であって、今後の環境整備が熱望されている。同時に、このような大型コードは、開発の始めから高速のコンピューター用に書かれるべきであって、その点、MAFIA には不満が残る。Table 1 にKEK でよく使われるコードをまとめた。

Table 1 Examples of codes for designing proton linac.

	第一種	第二種	第三種
RFQ	RFQUIK	PARMTEQ	MAFIA
	GENRFQ	<u>QKEK</u>	SUPERFISH
	<u>KEKRFO</u>		
DTL	<u>EQUI</u>	PARMILA	SUPERFISH
	<u>DELIFO</u>		MAFIA
CCL		<u>PROEND</u>	SUPERFISH
			MAFIA
Beam line		TRANSPORT	
		<u>LEBT</u>	
Magnet			PANDIRA JMAG

(下線のひいてあるのは KEK 製)

4. コード利用の具体例

KEK の中央計算機システムの中で行なってきた計算例を紹介する。まず、KEK の計算機環境を Table 2 に示す。

計算速度は標準並としても、問題は使い勝手にある。スカラーコンピューター (M680H) は、利用出来る記憶容量が最大で 75MB、計算時間が 2 時間である。スーパーコンピューター (S820) は、実用的に利用出来る記憶容量は 128MB、

Table 2 Speed of the computers at KEK.

	MIPS	MFLOPS	
HITAC M680H	32	59	B-system
次期HITAC	150	100	
Super HITAC S820		3000	C-system

計算時間は30分である。スカラーコンピューターは利用者が多いから、実質経過時間は計算時間の十倍以上になってしまう事がある。大きな三次元計算をする場合には、これらは障害となっているが、次第に解消される事を期待している。

4.1 MAFIA (old version) のチューニング

MAFIAは、Maxwell方程式を解く総合的なコードである。単に、電磁場解析をするだけでなく、電流（ビーム）を含む過渡状態まで計算出来るので、今後とも利用価値は高いと思われる。利用者が急増し、且つ計算時間のかかる大型のジョブが増えて來たので、それまでスカラーコンピューターにて走っていたMAFIAをスーパーコンピューターに乗せて走らせてみた。コードを書き換える前のベクトル化率は82.0 % である。筆者にとって、スーパーコンピューターを使う初めての経験であり、歩みはのろかったが、主としてサブルーティンを展開するなどの作業の結果、最終的には、ベクトル化率93 % 以上を得た。これ以上速くする為には、全般的な書き換えを必要とするので、作業を中止した。この結果、スカラーコンピューター上で82分（155000 メッシュ）かかった計算がスーパーコンピューター上では5分でできるようになった。この経緯をFig. 6 とFig. 7 に示す。

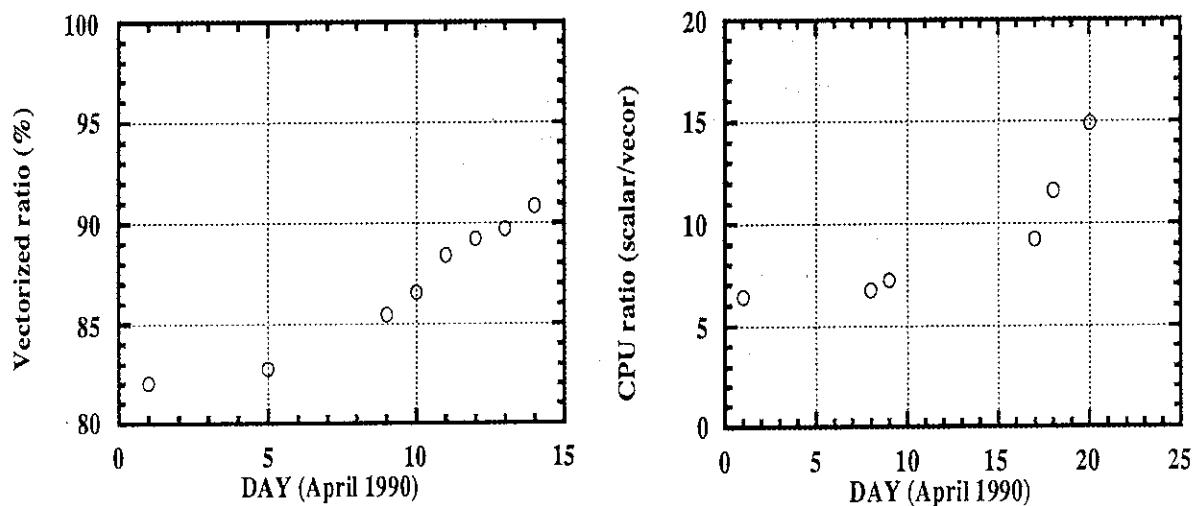


Fig. 6 Increase in the vectorized ratio by tuning. Fig. 7 Increase in the speed of calculation.

ここでの作業は良い経験となった。KEK のスーパーコンピューターは、ベクトル化率が 99 % 以上のコードである事を一般的に要求しているようなので、たかだか 93 % では、問題外という事である。早く良い使用環境が実現される事を望むと同時に、スーパー用のコードは、最初からそれなりに開発すべきであるという感想を持った。

4.2 RFQ の電磁場分布

RFQ (Fig. 8) は、低エネルギー領域のバンチングと横方向の収束と加速とを同時に行なうリニアックとして、期待されている。基本モードは収束力を産み出す TE21 であり、加速電極にうねりをつける事により加速電場成分を作っている。この加速成分をゆるやかに増大させる事により、イオン源から来る DC ビームをうまくバンチさせる事が出来る。軸対称でないこのモードは、長い間使われてきた電磁場計算コード (LALA, SUPERFISH) では殆ど取り扱えない。非常に tricky な計算法（長いアームをつけてドーナツ型の計算を行なう）を用いる事により、計算を行なっていた。三次元電磁場計算コードの登場により、RFQ は簡単に計算出来るようになった。しかしながら、必要なメッシュ数と計算時間を考えると、現在の KEK の大型計算機システムにおいて利用できるユーザー環境（利用可能なメモリー、実効的な計算速度）が二桁近く改善される事が望まれる。Fig. 9 - 11 に示したのは、MAFIA による RFQ の中心電極部の計算結果である。この計算は 165000 メッシュにて行ない、計算時間約 30 分である (M680 H)。

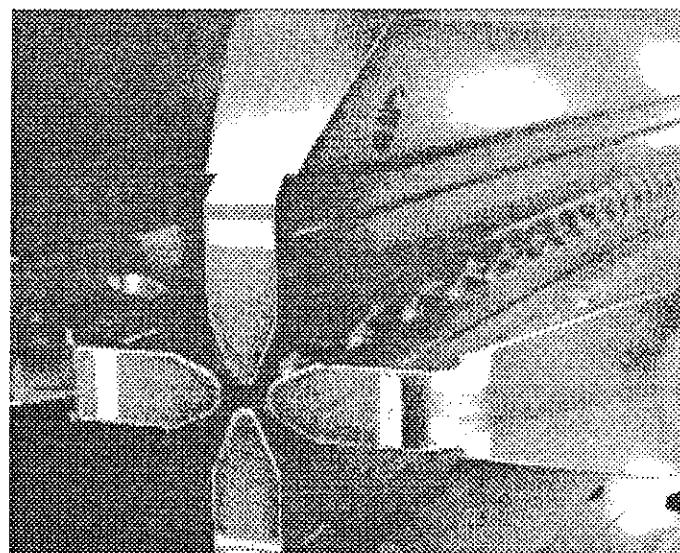


Fig. 8 Four vanes of an RFQ.

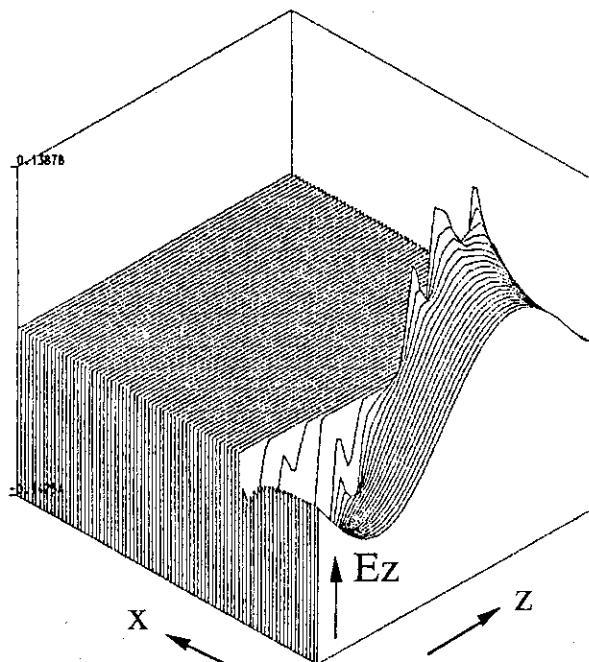
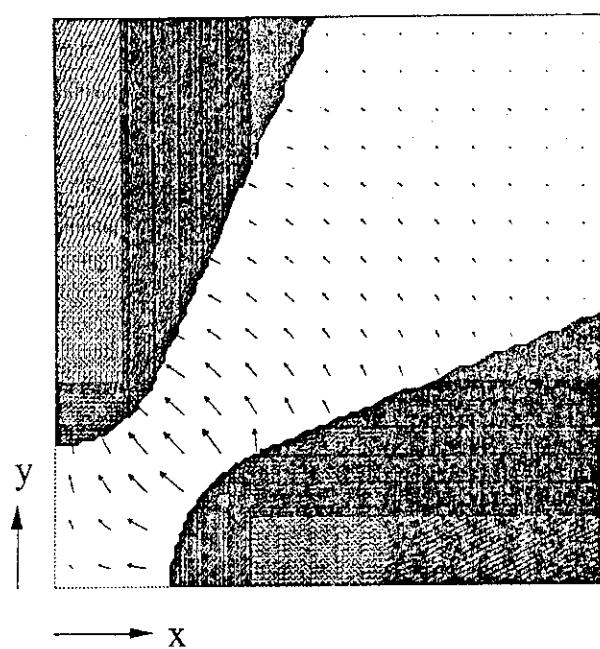


Fig. 9 Electric field distribution between two vanes. Fig. 10 Accelerating field distribution.

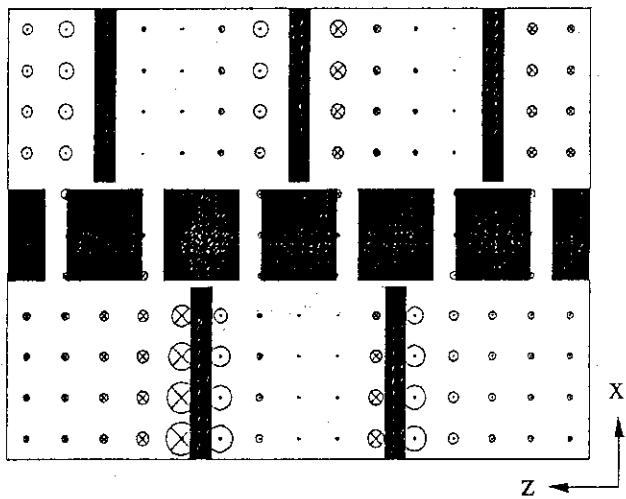
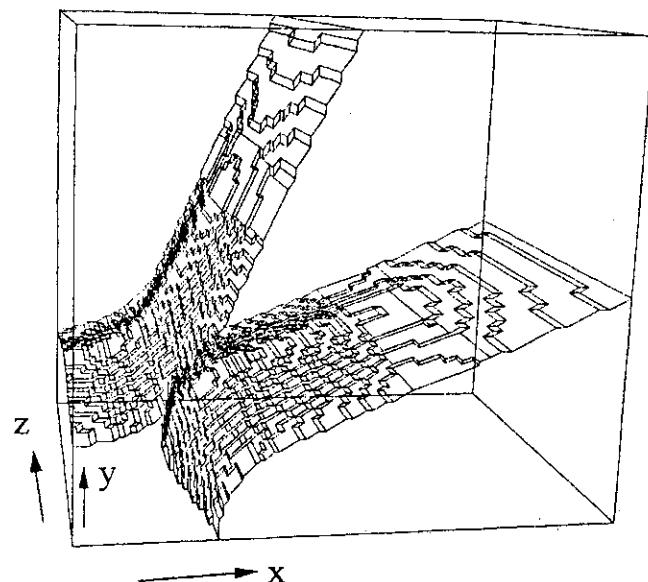


Fig. 11 Geometry of the vanes used for calculation. Fig. 12 Magnetic field distribution for post mode.

4.3 DTL の電磁場分布

DTL は、ドリフトチューブをぶら下げる為のシステム以外は軸対称なので、SUPERFISH により充分な計算が出来た。しかし、加速電場の安定化の為にポストカップラーを使うと、実に面倒な非対称系になる。加速管の直径が44cmと大きいのに対して、ポストカップラーの直径は1cmであり、チューニングの為の移動精度はmmの単位であるから、局所的なメッシュの大きさの選び方が難しくなる。精度を上げ

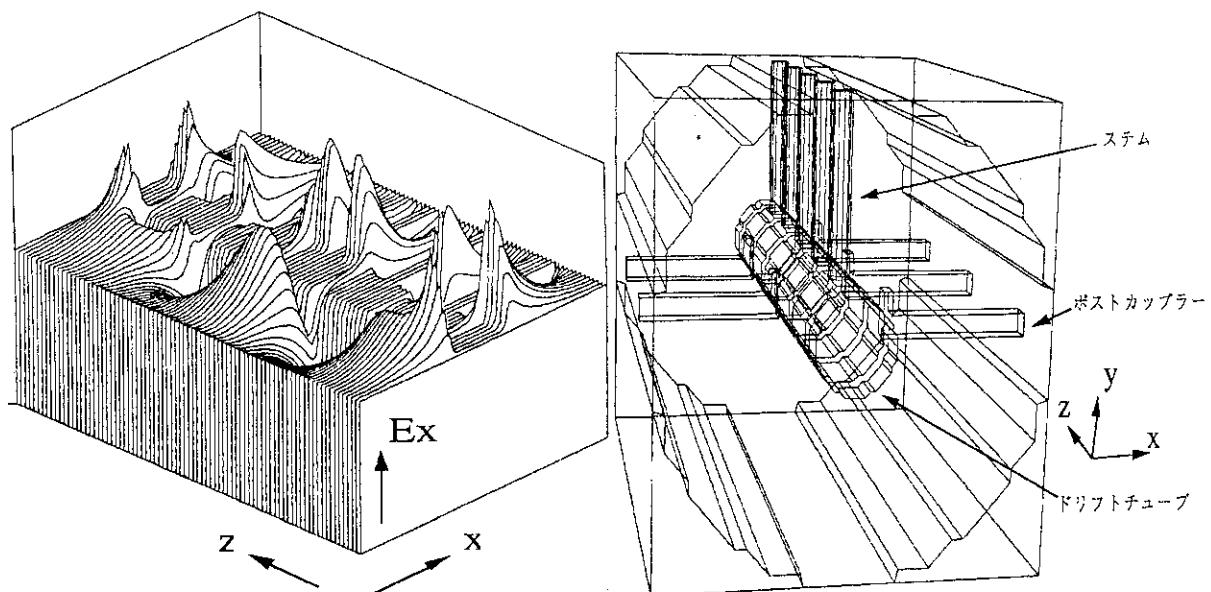


Fig. 13 Electric field distribution for post mode. Fig. 14 Geometry of DTL with post-couplers.

ようとするとメッシュが足りなくなり、メッシュサイズの比を大きく変化させると計算誤差が大きくなってしまう。従って全体のメッシュ数が100万程度では、精度の良い計算は出来ない悩みがある。Fig. 12 - 14に示したのは、ポストカップラー付きのDTLの計算結果である。精度は不充分ながらも、電磁場のモードがよくわかり、ポストモードの解析には非常に役にたつ。メッシュ数は405600、計算時間は113分である(M680H)。

4.4 LEBT (Low Energy Beam Transport) の計算例

電磁場中の粒子の運動方程式は

$$\frac{d\vec{p}}{dt} = q(\vec{E} + \vec{v} \times \vec{B}) \quad \vec{p} = m_0 \gamma \vec{v}$$

$$\vec{E}_i = \frac{q}{4\pi\epsilon_0} \sum_{j \neq i} \frac{1}{r_{ij}}$$

とかける。ここでは一般によく使われている記述法に従う。空間電荷効果は個々の粒子の間のクーロン力を計算して取り込む事にする。

この方程式を四次のルンゲクッタ法により倍精度を使って解く。ゼロからの出発なので、最初からスーパーコンピューター用に注意してコード化する。加速器用のビームラインを想定して次の様な構成とした。

1. 単位長さを $\beta\lambda$ とする。
2. 時間ステップは自由パラメーターとする。

3. 周期的又は自由な収束系を取り扱う。
4. 加速ギャップを取り入れる。
5. 空間電荷が無い時は、transfer matrix による厳密解と比較できる。
6. 多種類の入射粒子集団を用意し、外からの粒子も読みめるようにする。
7. 連続ビームとパンチビームを取り扱う。
8. 単位長さ毎に全粒子のデータをファイルに書き出す。
9. 必要なビームの性質を計算して、グラフ表示する。この一部はスカラーにて行なう。

Table 3 に、近年のコンピューターの速度の進歩を示した。

Table 3 Increase in speed of computers.

	1978	1990	YEAR
科学計算用	1	59	MFLOPS
ベクトル	10	3000	MFLOPS

1978年のコンピューターを使って、P-P (particle - particle) 計算を行なうと、例えば科学計算用では28時間かかり、ベクトルで行なうと3時間かかるから、P-P 計算是無理であると判断されていた。ところが、現代のコンピューターでは夫々28分と33秒なのである。P-P 計算を採用する価値はあると考えられる。Fig. 15 に最小限必要なビームライン長さ約70cmを計算するのに必要な時間を示す。粒子数5000以上の値は推定値である。ベクトル化率は99.6%以上あり、スカラーの30倍の速度が得られた。

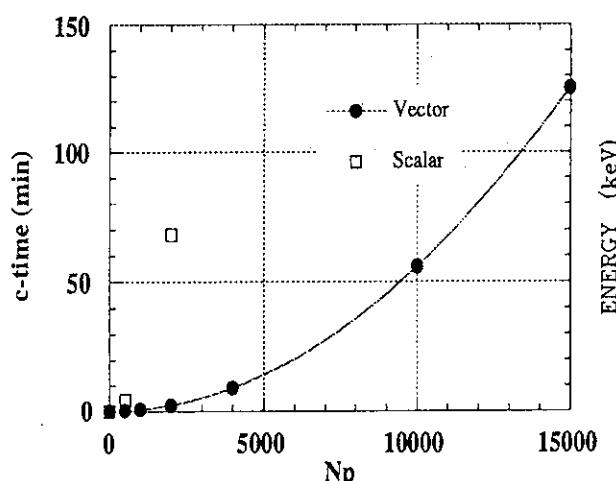


Fig. 15 Particle number dependence of CPU time for M680H and S820.

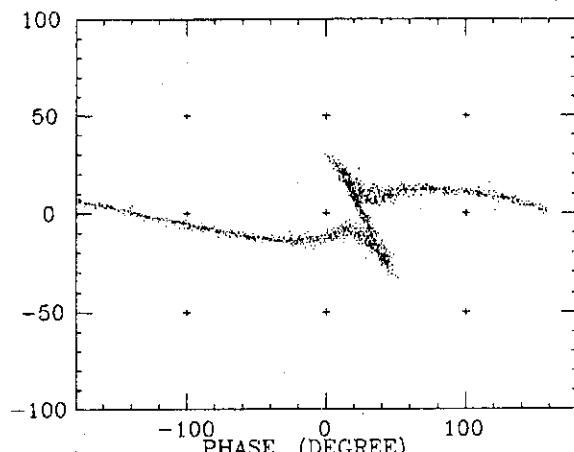


Fig. 16 A 100 mA injected beam with an RFQ.

このコードを使ってFig. 4に示した古い入射方式の空間電荷効果を計算してみる。エネルギー750 keVのビームはバンチャーランプにより急激にバンチングされるので、その時には大きな空間電荷効果がある。Fig. 16に電流が100 mAの時の縦方向エミッタンスを示す。Fig. 4と比べると空間電荷効果の作用がよくわかる。位相とエネルギーの広がりの増加、バンチャーランプの運転パラメーターの空間電荷効果による変化量などがこのシミュレーションから説明できた。

おわりに

今後への期待を膨らませてまとめたい。第一は、計算スピードの飛躍による効果である。10倍スピードが速くなると新しい事が期待出来る。今回は述べられなかったが、ビームを含む過渡状態の解析などは恩恵が大きい。第二は、次世代の知能的なコンピューターの進歩により、経験的な知識の集約を組み込んで、リニアックの設計が出来るかもしれないという期待である。そのようなコンピューターコードは、リニアックの設計の中で、単なる要素技術としての役割以上の働きをする事ができよう。

本発表の機会を与えて下さいました関係者の皆様にお礼申し上げます。

参考文献

- (1) Y. Yamazaki et. al., LA-11432-C, 80(1987), Los Alamos, KEK Preprint 87-159.
- (2) R. A. Jameson, 1981 Linear Accelerator Conf., 125 (1981).
- (3) Parmila Users and Reference Manual, LANL (1990).
- (4) K. Halbach and R. F. Holsinger, Particle Accelerators, 7, 213 (1976).
- (5) T. Weiland, Particle Accelerators, 15, 245 (1984).

4. Massively Parallel Supercomputing as the Centerpiece of a Diversified, Multidisciplinary Research Program

Prof. Claudio Rebbi(Boston Univ.)

**Physics Department and Center for Computational Science
Boston University**

Abstract

I am deeply honored to have been invited to attend and present a talk for the 7th meeting on Nuclear Code Development at the Tokai Research Establishment of the Japan Atomic Energy Research Institute. It is a great pleasure for me to have the opportunity of attending this very interesting meeting and to illustrate our work on massively parallel supercomputing.

At Boston University a few years ago we took the bold step of acquiring a massively parallel supercomputer, a Connection Machine CM-2, making it the central resource for a multidepartmental, multidisciplinary research effort. It was a bold step because at that time there was considerable skepticism about the range of problems to which this new technology could be applied. However, since that time, researchers at the University have shown that this new technology can indeed be applied to a vast range of computational problems.

In this talk I will analyze the viability of a massively parallel architecture for various types of computational research projects. I will focus on the main algorithmic aspects of wide classes of computations to discuss the challenges that they pose for a massively parallel implementation. The three major classes of computations will encompass:

- ◊ the solution of field equations, where the data structures and communication patterns are well organized and of local nature;
- ◊ molecular dynamics simulations, where one still has well organized data structures, but faces problems of global communication;
- ◊ optimization problems, with variable data structures and variable communication patterns.

For all of these classes I will give examples drawing on the wide range of massively parallel applications under investigation at Boston University.

Finally, I will discuss some very exciting possible future developments in massively parallel supercomputing.

1. Preamble

It is a great pleasure and a great honor for me to be given the opportunity of presenting this talk at the 7th Topical Meeting on Nuclear Code Development held at the Tokai Research Establishment of the Japan Atomic Energy Research Institute.

When I received the invitation, I thought that it would be a great challenge to give a talk on topics of advanced supercomputing to an audience of computational scientists of great expertise, such as the researchers attending this meeting. What could I say that, without being too technical and too specific to my research field, could nevertheless convey valuable information to the scientists in the audience? I decided that I could accomplish this task by reviewing our experience at Boston University in adapting a massively parallel supercomputer to a very diversified research environment. Massively parallel supercomputing is indeed still a rapidly developing field and several questions remain open on how suitable it may be for a wide range of computational problems. At Boston University we faced the problem of porting many different applications to the Connection Machine and our experience has to large extent become a case study in the development of massively parallel algorithms. Thus I hope that the work we have done in recent years at Boston University may help shed light on important questions of a more general nature on the applicability of the massively parallel paradigm to advanced computational problems. I hope that my talk may provide the audience with valuable information on these pressing matters. Preparing it has certainly helped me focus on broad issues of algorithm development and I wish to express again my gratitude for the invitation to present this talk. Let me make explicit my thanks to Mr. Masayuki Akimoto and Dr. Konomo Sanokawa of the Japan Atomic Energy Research Institute and to Dr. Masao Nozawa, of the Nuclear Energy Data Center, for their invitation and to Dr. Mitsuo Yokokawa of the Japan Atomic Energy Research Institute for invaluable assistance in the organization of my trip.

2. Introduction

Boston University has long recognized the importance of advanced scientific computing. When I joined the University in 1986 with a background of several years of research in computational methods for particle theory, I was very pleased to see that many of my newly acquired colleagues were quite active in a variety of computationally based research projects. The University's commitment to academic computing was apparent from the fact that it was one of the first academic institutions to provide its scientists with a vectorized supercomputer, an IBM 3090, and had installed a campus-wide advanced network and many departmental computers. This prompted a few of my colleagues¹ and me to propose to the University Administration a further expansion of our resources, through the acquisition of a more modern and powerful supercomputer, and the establishment of a Center for Computational Science, which would capitalize on the widespread interest in scientific computing to create a forum for the exchange of information and the promotion of advanced computational research. I am pleased to report that both suggestions have been met: in the Fall of 1988 Boston University purchased a Connection Machine CM-2, made by Thinking Machines Corporation, and in November of 1990, the administration formally inaugurated the Center for Computational Science.

One of the most important questions that we faced at that early stage was what supercomputer would best fulfill our major goal, namely to provide a resource, pioneering in nature, but at the same time capable of sustaining a wide, multidisciplinary and multidepartmental research effort. Needless to say we explored several options, including the most advanced vectorized supercomputers then available. We also considered supercomputers with massively parallel architectures, with particular attention to the Connection Machine with which some of our colleagues had already had research experience. It was immediately clear that massively parallel supercomputers would be very powerful for studying some special classes of problems. The crucial question was whether their range of applicability was sufficiently wide to justify making a machine with such architecture the centerpiece of the whole, University wide supercomputing research activity.

We examined this matter very carefully and during this period established the beginnings of a most productive interaction with scientists at Thinking Machines Corporation. We became convinced that it would be possible to port a sufficiently wide class of problems to a massively parallel architecture and that the potential gains in performance that the Connection Machine offered warranted the acquisition of this new technology. Since then many of our scientists have made enormous progress in applying the CM-2 to a very wide variety of computational research problems. I think that it is fair to say that the range of applicability of the Connection Machine has not only met, but and exceeded our expectations. The history of our research efforts has become to an appreciable extent a case study in how vastly and efficiently massive parallelism can be used in solving advanced computational problems. This is the topic that I will address in the balance of my talk.

¹I would like to mention their names in order to give credit to their pioneering efforts. They are: Richard Brower, David Coker, Robert Devaney, Dan Dill, Roscoe Giles, Stephen Grossberg, William Klein and John Porter.

3. The Connection Machine

In this talk I will focus on the Connection Machine CM-2. This is of course not the only massively parallel supercomputer commercially available. There are others with excellent performance which also allow the user to take great advantage of this novel architecture. But I know the details of the Connection Machine best. Moreover it is the supercomputer onto which we, at Boston University, have ported all the applications that I shall illustrate in my talk. Although I will concentrate on the Connection Machine, many of the things I will say are applicable to all massively parallel supercomputers. However, some of my remarks will be specific to the class of computers known as Single Instructions Multiple Data (SIMD) machines, others will be even more specifically directed to the CM-2.

The Connection Machine is a SIMD machine. Data are distributed through many processors. Operations on these data are performed by the individual processors in parallel, with each processor accessing the data that reside in its own memory. But the operations performed are the same, at any given step, throughout all the processors. Indeed, in the CM, the instructions themselves do not originate in the processors, but are broadcast from a front-end computer, where the actual program resides. This mode of operation – identical instructions performed on distributed data – constitutes the mode of operation characteristic of a SIMD machine. Multiple Instructions Multiple Data (MIMD) machines instead typically execute different instructions in their multiple processors. This mode of operation is characteristic of all supercomputers which possess multiple processors, but in rather limited number, (frequently, but not necessarily, vectorized machines, e.g., an 8 processor Cray Y-MP). These are parallel, but not massively parallel supercomputers. There are however massively parallel supercomputers with MIMD architecture as well.

As I mentioned above, the processors in the Connection Machine access and operate on memories directly associated with each processor. In contrast to this notion of local memory, one can think of situations where every processor can directly access data anywhere in memory. This is the typical mode of operation of supercomputers with limited parallelism. But given a very large number of processors one would run into severe problems of bandwidth as well as memory conflicts. Thus in massively parallel machines processors typically access only subsets of the data directly. "Directly," however, is the key word. Through a suitable communication network every element of data can be made available to every processor. The efficiency of such a communication process is of crucial importance for many applications.

An examination of the details of the architecture of the CM-2 reveals a rather hybrid structure. Indeed, the present Connection Machine, namely the CM-2, and the most recently released CM-200, have evolved by virtue of the addition of very powerful floating point capabilities, from a machine with simpler architecture, the CM-1 ([1]). This evolution has increased enormously the capabilities of the Connection Machine, but has resulted in a hardware architecture in which two different design concepts coexist.

The original architecture of the Connection Machine consists of a large number of single bit-serial processors, each one with its own memory. The total number of processors is a power of two, with $2^{16} = 65536 \equiv 64k$ as the maximum configuration. Today the memory of each processor can be as high as 1Mbit, resulting in a total memory of 8Gbytes in the maximal configuration. From the

point of view of the network, each processor can be thought of as the vertex of a hypercube, e.g. a 16 dimensional hypercube in the case of a 64k machine, capable of communicating directly with any neighboring processor in the hypercube. Communications between non-neighboring vertices must proceed in steps. However, because of the hypercube structure, every processor can communicate with any other processor in up to \log_2 (number of processors) steps. (In reality, the situation is even better than that. Since in the actual hardware, groups of 16 processors physically reside in a single VLSI chip, the real communication costs apply only to communication between processors located on different chips).

While this architecture has proven extremely efficient for communications and a variety of logic manipulations, the bit-serial nature of the individual processors does not allow for great efficiency in floating point operations. Thus, in a further development, fast floating point chips were added to the hardware, leading to the basic design of the CM-2. One floating point chip was added to each group of 32 bit-serial processors. These floating point chips can address the data in the processors in two ways. First, floating point numbers can reside entirely in the memory of the individual processors. Then, if, for example, all 64k processors are used for a calculation and a parallel variable, say A, is distributed through the machine, this means that A represents an array with 64k elements. In order to operate on this array, every floating point chip will operate in sequence on the 32 elements of A which reside in the memories of the 32 processors physically linked with the floating point chip. But secondly, it is also possible to store floating point data in a "shared" mode, whereby the 32 (64) bits of a single (double) precision number are distributed through the memories of the 32 processors linked to the floating point chip, which accesses them all simultaneously for a floating point operation. Then, clearly, if all 64k processors are used, as above, for a calculation and a parallel variable A is distributed through the machine, A is only a 2k element array.

The hybrid structure I was referring to consists in the coexistence of these two hardware elements, the bit-serial processors and the floating point chips. For the bit-serial processors, the granularity (i.e. the maximum subdivision of the data) of the machine is very fine. For the floating point chips, the natural granularity is 32 times coarser. This coexistence of two distinct hardware concepts is an accidental product of the actual evolution of the machine, not an essential feature of it, and I do not expect it to survive as new models of the Connection Machine are developed. It does not result in any penalty, from the point of view of the efficiency of the machine. On the contrary, for some calculations, the availability of two modes of data storage can be an advantage. We do not need to dwell any longer on such architectural details and even the programmer can ignore them to a large extent. The only crucial notion is the amount of granularity of the machine, which in a fully configured CM-2 (or CM-200) and for a shared storage of the data is 2k. Programming can be done at high level. Thinking Machines supports a parallel FORTRAN compiler (FORTRAN 90, with some extensions), a C compiler with parallel extensions and a parallel LISP language. Moreover one can access all parallel instructions directly through an extensive set of PARIS (parallel instruction set) subroutine calls, which can be incorporated into any of the higher level languages.

Another important aspect of the hardware is that the machine can be partitioned (up to fourfold in the present CM). This means that if an institution has, for instance, a CM with 32k processors (as we do at Boston University), it is possible to have all processors working simultaneously on a definite problem, or to have the machine divided into any combination of sub-cubes with 8k

or 16k processors and have them work independently under control of the same front-end or of multiple, independent front-ends. Moreover, any configuration with a definite number of physical processors, say 2^n , can be used to implement calculations with a larger number, $2^{n+1}, 2^{n+2}, \dots$ of virtual processors, if it becomes necessary to accommodate larger data structures. This means that larger hypercube configurations can be implemented (very efficiently, too) in software, replicating the data within the memories of the physical processors up to memory capacity.

The data in the machine can be visualized in real time either by means of a frame buffer, which permits the display of bit-mapped images of the memory on a high resolution monitor, or by sending the data through the network to some generic display terminal. High capacity Data Vaults permit the storage of large volumes of data on arrays of fixed disks which can be read and written in parallel. This makes it possible to download large portions of the memory of the CM, or the whole memory, in a short time and, if desired, to run the system in a time-shared multi-user mode. At Boston University we integrated our installation of the Connection Machine into a computational environment consisting, among other things, of several Silicon Graphics computers (serial processor machines with powerful special purpose, vectorized graphics hardware), a Data Vault and a Very High Speed network (these last two items were acquired through grants from the National Science Foundation).

4. Algorithms for Massively Parallel Computation

It is rather obvious that a massively parallel computer can only be useful for problems which involve a large number of variables that can be manipulated with some degree of independence. Then one can try to take advantage of the large number of processor by executing simultaneous operations on the arrays of data. A problem where only one or just a few data items can be manipulated at any one time is hardly suitable for a massively parallel architecture.

Thus, for instance, if our problem is the one of calculating the trajectory of a single classical particle in some potential field, where the only relevant variables are the particle's position and velocity and the calculation must generate iteratively a large number of subsequent values for such variables, nothing can be gained from a massively parallel computer over a single processor machine. But then, luckily for the potential applications of massively parallel computers, it is precisely those calculations that have large data sets that normally end up being the most time consuming and therefore those where an increase in computational efficiency can produce the largest payoff.

The application of a massively parallel computer to the problem of calculating a single particle's trajectory would be justified if one had to repeat the calculation a very large number of times in correspondence to a very large set of different initial data. Then the massively parallel computer turns from being the less suitable machine to being the most efficient tool for the problem under consideration. Indeed, every individual processor can then be assigned to the calculation of one of the independent trajectories and all can work together performing at each step identical instructions, namely those that serve to advance the trajectories to their next iterates.

Such an application is not so farfetched. We encounter it in the study of dynamical systems when we want to determine which domains in phase space correspond to stable, bounded orbits, which domains correspond to unstable, unbounded orbits, and where the motion is periodic rather than chaotic.

Similarly, one may want to study the properties of iterative mathematical mappings, such as those giving origin to Mandelbrot sets or Julia sets, with all their interesting fractal behavior. The operations to be performed are really elementary and the complexity of the calculation arises entirely from the fact that the mapping must be iterated over a very large number of points, trying to explore its behavior in the complex plane with as high a resolution as possible. At Boston University Professor Robert Devaney and collaborators have been studying these iterated mappings ([2], [3]). Not surprisingly they found the Connection Machine to be an ideal tool for calculating the fractal patterns determined by the iterated mappings. In some of their research they have been using movies which visualize how the patterns change in response to the variation of some crucial parameter in order to gain insight into the mathematical properties of the mappings. Since producing a single pattern, i.e. one of the frames in the movie, is *per se* a computationally intensive job, one can easily imagine the number crunching required to produce the entire movie. Professor Devaney and collaborators found that they could accomplish with an hour of CM time what it used to take them over a day of CPU-time on a more traditional, vectorized supercomputer.

Of course, these are rather special applications, and one can hardly expect a generic large scale computational problem to be so benign as to involve totally independent operations on a very large data set. In general, the results of the operations carried out within one processor will have to

be communicated to other processors in order to take into account the interdependencies among the data. This becomes then the key to the classification of algorithms that we will use in our discussion. We will broadly subdivide all computational problems in three groups, according to whether the required interprocessor communications are of local nature or span the whole processor set, and, in the latter case, according to whether the pattern of operations and communications remains regular or varies substantially in the course of the calculation.

4a. Problems of Field Modelling

There are very many problems where one needs to solve by computer a set of differential equations that describe the configuration or the evolution of some fields. The equations are discretized on a lattice and thus converted into difference equations that must be numerically solved. Such problems map extremely well onto the architecture of a massively parallel computer. The processors are put in close correspondence with the vertices of the lattice or, in case the number of processors is not large enough, with blocks of the lattice. Data elements that are neighboring in the physical problem are assigned either to the same processor or to processors which are neighbors in the hardware interconnections of the machine. The difference equations involve therefore communication among neighboring processors only. On the CM the transfer of data elements between contiguous vertices in the hypercube is extremely fast and indeed the problems that belong to this class are among those where one can get very high sustained performance without much of a programming effort. The required communication primitives can be implemented via PARIS calls or directly through high level language instructions. FORTRAN 90, for instance, has shift instructions which permit us to bring a copy of the neighboring elements in the lattice array onto the desired locations. Complex boundary geometries can be implemented by imbedding the system into a regular, rectangular geometry, using suitable masking operations to involve only those processors that represent interior points of the field.

The theoretical peak speed of the CM-2 is obtained by multiplying the floating point execution rate of the individual floating point units times their number. For the CM-2 at Boston University the floating point chips operate at approximately 7MHz (more recent models achieve 10MHz). In principle a floating point chip can execute an add and a multiply simultaneously, resulting in a theoretical peak speed of 14Mflops per chip. But let us leave aside this possibility, which, although realized in some specialized code, is not likely to be encountered in practice. Then the 1024 floating point chips of our 32k CM-2 have a theoretical peak speed of 7Gflops. In problems of field modelling a reasonably well written code, even in a high level language, will easily achieve a sustained performance ranging between 1Gflops and 2Gflops. In many applications one would have only one fourth of the machine available (an 8k partition), but could still achieve a sustained performance of several hundred Mflops.

In another class of problems, that describes quite different physical phenomena, but which is algorithmically very similar to the above, the evolution of the variables is stochastic rather than deterministic. One encounters this case, for example, in the statistical mechanics simulation of finite temperature systems. However the evolution of the variables is still determined, albeit in a stochastic manner, by the values taken by their neighboring variables, so, from the point of view of communications, the problem is similar to the above. In some cases the need to preserve statistical independence in the evolution of the system forces separate upgrades of the variables at even or odd

lattice sites (the so-called checkerboard or red-black subdivision of the lattice, well known also from vectorized implementations). One can then either use a masking operation to upgrade only the even or odd sites at any given step or, at the expense of a slightly more complex communication pattern, associate pairs of sites, one even and one odd, to the individual processors, so that all processors can remain busy at all stages of the upgrade. The best strategy will be determined by the balance of the communication versus operation costs in the specifics of the problem. However, an important notion in approaching massively parallel programming is that, because of the very large number of processors, operations have become a cheap commodity. One can then afford to have only part of the processors active at any one time and, provided this fraction is not way too small, the overall code will still perform at a satisfactorily high speed. In other words, with massive parallelism, the application scientist should free him/herself from the notion that, because computing is so expensive, every single unit of the supercomputer should be kept busy, if possible, at any one time. The massively parallel supercomputer should be thought of more like we think of our own brain: we have so many neurons that we can afford to keep a large fraction of them idle at any one time. Indeed, if we managed to put them all to work simultaneously, we would end up with a major headache!

The problems that I am most likely to encounter in my own field of research, computational particle theory, belong to this class. The applications of supercomputing to particle theory are fascinating and they are also extremely challenging from the point of view of the computational power required. Since they do represent my own field of interest, I would like to be allowed to spend a moment to illustrate them in just a little more detail.

Particles are described in terms of quantized fields. Their properties result from averages over the quantum mechanical fluctuations of such fields, very much like in statistical mechanics the values of physical observables are given by averages over the thermodynamical fluctuations of the system. In the theory of free fields, the quantum fluctuations are determined by a gaussian measure and therefore it is possible to calculate the quantum mechanical averages exactly by performing gaussian integrals. In the analytical approaches to the study of interacting fields, theorists try to isolate a free field component in the dynamics and then account for the interactions in a perturbative manner. This approach, based on perturbation theory, has been the main tool of quantum field theory since its historical origin and has produced remarkable successes. However, there are also vast classes of problems in particle physics where the perturbative approach does not work. Indeed, there are important cases where one can prove, on the basis of analytical arguments, that perturbation theory cannot be applied. An achievement of the last decade has been the realization that in many circumstances it is possible to calculate numerically the average over quantum fluctuations after the theory has been discretized over a suitable space-time lattice ([9], [10], [11]). The averages over the quantum fluctuations of the particle fields are performed by stochastic simulation techniques, very much like one proceeds with the simulations of thermodynamical systems, but the fields must be simulated over discretized space-time, i.e. over a 4-dimensional geometry. This implies that any realistic simulation involves lattices with a very large number of vertices and therefore requires the simulation of systems with an exceedingly high number of variables (lattices with $32^4 = 1M$ sites, over which the field configuration is specified in terms of 92M real numbers, have been studied).

Additional difficulties in the computational studies of non-perturbative particle phenomena derive from the need of accommodating variables which obey Fermi-Dirac, i.e. anticommuting, statistics.

This is a rather technical problem and let me say here only that what it implies, from the algorithmic point of view, is that at each step of the simulation process one must solve a very large system of linear equations. The matrix of coefficients is very sparse and one can use iterative procedures, such as relaxation or conjugate gradient algorithms. Indeed, one must use such techniques, since the number of equations and unknowns easily range in the millions. In the course of a complete calculation, systems of equations of this sort must be solved thousands of times, which, as one can easily appreciate, is computationally very demanding. Massive parallelism is ideally suited for these types of calculations and indeed it appears to be the only architecture capable of fulfilling the requirements for future progress in the field.

At Boston University we have implemented a variety of field modelling calculations on the Connection Machine, encompassing such applications as the simulation of high temperature Quantum Chromodynamics in particle theory, of the evolution and collision of cosmic strings and superconducting vortices, of radiation from antenna arrays, of a variety of models for nucleation, diffusion limited aggregation, dendritic growth, etc. (For instance, see [12], [13], [14], [15]) As discussed above, massive parallelism is very suitable for the implementation of all these problems.

One might wonder at this point whether there is any challenge at all, from an algorithmic point of view, in the application of massive parallelism to the problems presently under consideration. The answer is in the affirmative. However this is not because of straightforward programming considerations, but for subtler physical reasons. As the size of the physical system increases, the dynamics that governs the evolution of the field configuration in all of the above simulation processes, be they of stochastic or deterministic type, frequently undergoes critical slowing down. This stems from a competition between the natural rates of evolution of the long range and short range excitations of the physical system and implies that, if one wishes to simulate larger and larger systems, one must pay not only the price due to the sheer increase of the number of variables but also a sometimes very heavy additional cost due to the fact that the system evolves more slowly. Several techniques that permit us to overcome or reduce critical slowing down have been devised, such as Fourier acceleration, multigrid techniques and clustering algorithms, but this is still an active field of research. A general feature of all these techniques is that they provide a framework for isolating and projecting out of the system the long range modes responsible for critical slowing down. Since these are long range excitations, their determination involves problems of communications that extend beyond the nearest neighbors in the geometry of the system. These communication patterns can still be implemented, but not in as straightforward a manner as in the basic simulation of the evolution of the fields. Here lies the challenge for the application scientist: to devise the appropriate acceleration algorithms to overcome critical slowing down and to implement them in an efficient manner on the massively parallel supercomputer. In general this will involve some additional communication burdens, but the gain from the increased speed of the simulation will, in most cases, surpass by far any additional computational costs.

4b Molecular Dynamics Simulations

Next in computational complexity after the problems discussed in the previous subsection come those calculations which deal with a fixed, easily parallelizable data set, but where the communications span the entire machine. I have grouped such problems under the general heading of molecular dynamics simulations because of the importance of this class of simulations and because

the patterns of communication which I want to discuss are very likely to occur in molecular dynamics calculations. However the need for global communications with a fixed data set arises also in the solution of many other problems.

Indeed, the first example that I would like to discuss can, in some of its aspects, be considered a generalization of a problem of field modelling. At Boston University Professor Roscoe Giles and collaborators have been studying the properties of magnetic films, with particular attention to the way in which magnetic domains, domain walls, Bloch lines and points form and evolve under the action of external magnetic fields ([4], [5], [6], [7], [8]). The importance of such calculations for materials science and for computer technology can hardly be overemphasized. A good theoretical understanding of these phenomena is an obvious clue to the design of better substrata for the magnetic storage and retrieval of information.

The study of magnetic films proceeds through very large scale simulation of the evolution of microscopic variables representing the local magnetization of the medium. These variables are associated to the processors. Their evolution is determined by forces due to crystalline anisotropy which are computed locally, by exchange forces which involve nearest neighbor couplings and which are efficiently computed by regular grid interprocessor communications, and by long range magnetostatic forces. These long range magnetostatic components represent the novel feature with respect to the problems studied in the previous subsection. Indeed, the evolution of every site variable ends up being influenced, through the long range magnetostatic forces, by every other site variable.

A straightforward implementation of the effects of the magnetostatic interactions would involve communicating to each processor the value of the state variables at all other processor sites, a very costly, global communication procedure. However, in this specific problem one can exploit the linear nature of the equations for the magnetostatic interaction to solve them by means of a Fourier transform. A fast Fourier transform and its inverse can be performed very efficiently by taking advantage of the hypercube architecture of the machine. Thus we see here a case where a problem of global communication can be solved by resorting to an algorithm that makes fundamental use both of the mathematical properties of the equations and of the architectural structure of the machine.

In molecular dynamics simulations one implements via the computer the equations for the evolution of a very large number of interacting particles. Molecular dynamics simulations play an extremely important role in the study of a wide class of phenomena, such as melting and crystallization, glass formation, protein folding, turbulence, plasma, galactic dynamics, etc. The data set is very suitable for parallelization: typically one will assign individual particles or small sets with a fixed number of particles to the processors. The problem arises from the need of evaluating the combined force exerted on any definite particle by all the other particles in the system. (I am using the word "force" in a rather general sense, to denote all the terms in the equations of evolution for a definite variable which depend on the values taken by the other variables of the system. Often these terms will represent actual mechanical forces, but our discussion applies also to the cases where the interactions are of more general nature). We encounter then two possibilities: either the forces are of long range, in which case all terms must be computed, or they are of short range, so that we may limit the calculation to the forces stemming from the particles that are physical neighbors of the particle being evolved.

In the former case, the case of long range forces, unless we can take advantage of some special feature of the equations of motion in order to simplify the calculation of the forces as done for the study of magnetic materials, in order to evolve a particle we must necessarily collect information about all the other particles in the system. This could be done by repeated communications whereby the individual processors poll all of the other processors about the state of the particles they contain. Obviously the resulting pattern of communications would be of global nature and quite complex, and the communication routines would then end up being very time consuming.

With some ingenuity, however, we can make better use of the fact that the data structure is fixed and that the machine has a parallel architecture. Emulating a data flow paradigm, we can shift all the particle data by one unit through the ordered array of processors, and then by yet another unit, and so on, in such a manner that the data for every individual particle go, cyclically, through all the processors in the machine. As these variables move through the processors, the forces of interaction between the particle originally in the processor, whose data are maintained in the processor's memory, and the particle "passing by" are calculated and accumulated. At the end of a complete tour of the machine every particle has returned to the original processor and all the forces have been computed. This procedure cannot, of course, eliminate the fact that $O(N^2)$ operations must be done for N particles, but it takes full advantage of massive parallelism to perform all operations in $O(N)$ steps and makes use of fast nearest neighbor communication routines.

If the forces are of short range, the number of operations required for evolving N particles remains $O(N)$, since the calculation of the interaction can be limited to those particles which are physically neighboring. However, the proximity of any two particles is now determined dynamically by their evolution and is not fixed by the geometry, as in the case of the field modelling calculations. Thus linked lists assigning particles to a regular lattice of space cells must be maintained and updated. For the calculation of the forces, the particles are distributed among the parallel processors associated to the cells. For the integration of the equations of motion, the particle data are moved back to the processors associated to the individual particles. It is remarkable that the computational efficiency of the algorithm ends up being correlated with the physical state of the system. Indeed the average occupation number of the cells required for the calculation of the forces varies inversely with the magnitude of the density fluctuations. It is then possible to tune suitable computational parameters, on the basis of the available information about the physical properties of the system, in order to achieve better performance. This is one of many examples where insight into the physics of the problem under consideration can help achieve better computational efficiency.

The development of massively parallel algorithms for molecular dynamics simulations is an area of research actively pursued by scientists at Boston University, together with collaborators at Thinking Machines Corporation ([16]). They expect to be able to study the crystallization of a Lennard-Jones fluid with as many as 250,000 particles and to be able to investigate the onset of turbulence in a driven Lennard-Jones fluid.

4c. Problems of Optimization

As in the case of molecular dynamics simulations, I am using the subtitle "Problems of Optimization" in a rather general sense. The next and highest level of complexity in the application of massively parallel architectures is found when both the data structure and the pattern of commu-

nication are irregular and can change in the course of the computation. This is typically the case in problems of optimization, where one needs to explore a very large space of possible solutions in order to find an optimal one. However, such complexity can occur also in other problems, such as in detector simulation for the design and analysis of high energy physics experiments, where one can evolve large number of events in parallel, but meets serious problems of table look-ups and load balancing.

At Boston University a large group of researchers is studying the application of massive parallelism to the protein folding problem ([17], [18], [19]). Proteins are characterized by rather unique and stable structures that appear to emerge solely as a consequence of their sequence of amino acids. The protein folding problem consists in predicting the final three-dimensional structure assumed by a protein as it evolves from the original, unfolded linear chain. Since the geometrical structure of a protein is crucial to its biological functions, solving the protein folding problem is of enormous importance for the basic understanding of physiological processes as well as for its potential pharmaceutical applications to the design of drugs and vaccines.

The protein folding problem consists essentially in finding the configuration of the protein that minimizes the free energy of a particular chain of amino acids. But for the smallest sequences of amino acids, the space of configurations is so large that an exhaustive exploration constitutes a computationally impossible problem. One must therefore develop heuristics which take into account a substantial amount of information about energetically favorable substructures and configurational constraints. Algorithms based on dynamic programming and simulated annealing are being developed. A massively parallel architecture seems *per se* ideal for evolving the very large number of variables that characterize a configuration as one explores the free energy surface, but the implementation of a massively parallel calculation is made complex by the fact that the objects being modeled do not form a regular grid and that the interactions among them change as the algorithm progresses. One must therefore devise computational techniques that make optimal use of the interprocessor communication capabilities for updating linked lists and for rearranging the data structure as the computation progresses.

Similar algorithmic problems are encountered in problems as varied as multisensor data fusion and track reconstruction, speech recognition and identification of clusters. Many Boston University scientists are working on these problems, which emerge in different fields research but are unified by common algorithmic features.

Multisensor data fusion and track reconstruction is being investigated by Professor David Castañon and collaborators and by Professor Robert Hohlfeld and collaborators at the College of Engineering. The idea is to take advantage of massive parallelism to correlate a very large number of signals coming from different sensors. The information provided by each sensor must be related to the information given by other sensors so as to achieve an unambiguous identification of the signals that correspond to the same object. This is achieved by optimizing a maximum-likelihood function through a parallel Lagrangian relaxation algorithm. The solution of this problem has obvious, important applications to air traffic control, highway traffic monitoring, military surveillance, remote sensing, etc.

Of similar nature is the problem of reconstructing particle tracks within the detectors used in high

energy physics experiments, which is being studied by Professor Robert Wilson, of the Boston University Physics Department, and collaborators. In preliminary investigations they map the processors to the geometry of a simplified detector model and then use interprocessor communications to implement a filtering scheme which builds up and identifies the trajectories of the particles.

Another problem where one must solve global connectivity problems is the one of cluster identification. In the section on field modelling problems I have mentioned the fact that many algorithms suffer from critical slowing down in the region where the physics of the phenomena develop long correlation lengths, which is very often precisely the domain one wishes to study. In recent years it has been shown that the simulation of spin models and related systems in the neighborhood of critical points can be made much more efficient if, rather than updating the individual spins, one identifies clusters of spins and then changes the orientation of all the spins in the cluster in a single stroke. Cluster identification is of course useful also for many other computational problems. Building up a cluster would *a priori* seem to be a typical serial process, whereby one moves from spin to spin within a cluster, keeping a list of branches which still need to be moved forward. However, here as in many other problems which do not appear immediately suitable for a massively parallel architecture, one can achieve a parallel implementation by approaching the problem from the very beginning with a different computational perspective.

Richard Brower, Pablo Tamayo and Bryant York ([20]), working on the Connection Machine at Boston University, have developed an efficient massively parallel cluster identification algorithm which is based on the strategy of label propagation. At the beginning of the computation all the spins are assigned different numerical labels. Spins at neighboring sites are then compared and, if they are connected and therefore belong to the same cluster, the spin with the highest numerical label gets its label replaced with the one of the other spin. The procedure is then iterated. The lowest labels thus diffuse through the clusters until the lack of further label changes signals that complete cluster identification has been achieved. More recently, John Apostolakis, Paul Coddington and Enzo Marinari, working at Syracuse University ([21]) have been able to improve the algorithm incorporating into it multigrid features. In the course of the calculation the links between spins, which originally involve only nearest neighbors, are extended over longer distances, by adjoining shorter links into longer ones. Moreover, comparisons of spins are performed over multiple scales of distance, by taking advantage of the fact that communications between processors whose coordinates differ by powers of two is particularly fast. By sweeping over these multiple distance scales in a systematic manner, as is typical of multigrid algorithms, one achieves efficient cluster identification even in the cases where the clusters display a fractal structure and the basic algorithm undergoes critical slowing down.

The final example that I would like to discuss is from the work of Professor Robert Wilson and collaborators on particle shower simulation. The simulation of the process by which a primary particle, entering into a detector, produces a stream, or cascade, of secondary particles and of the propagation of such showers is of fundamental importance for detector design and other applications. There exist several well established serial codes that implement such simulations on conventional computers. The various particles that compose a shower are evolved individually, maintaining a list of produced particles that need to be evolved in turn, until all branches are exhausted. The procedure is time consuming and, since one typically must simulate a very large number of events, any speed up from a massively parallel implementation would be particularly welcome.

So long as all the particles evolving in the shower are of the same nature, they can all be propagated in parallel. Indeed, since, as I said above, one must typically simulate several showers, one can take advantage of massive parallelism to propagate simultaneously particles from different events. The problem is that different types of particles can coexist in a shower. Moreover, the number of particles is not fixed, but varies over a big range in the course of the cascade. This second problem is solved by maintaining in the local memory of the processors a stack of particles that need to be evolved. The first problem, the fact that different types of particles must be evolved, which of course requires different sets of operations, is particularly serious with a SIMD architecture. Wilson and collaborators have solved both problems by maintaining in the local memories of the processors a stack with the information on the particles that must be evolved. At any given step, all of particles of a definite type which are highest (among particles of that type) in the stack get evolved. One continues that to another type of particle, and so on, in a cyclic manner, continually drawing particles from the stacks and feeding to the stacks newly produced particles, until the propagation of all the showers is completed.

Clearly the procedure will work in an efficient manner if there are enough particles of a definite type, at each step, to keep a large fraction of the processors busy. If the stacks are uniformly depleted, one can start new showers. The problem arises if the distribution of particles in the stacks is very uneven, so that some processors contain large number of particles to evolve while other are idle. This situation can be remedied by systematic load balancing operations. Either at fixed times or when suitable checks in the program monitor a load unbalance, particles are redistributed among the processors so as to reestablish a more uniform loading of the stacks. This produces a dramatic increase in the efficiency of the algorithm, but of course requires global interprocessor communications.

Another potential problem in the calculation is that the evolution of the showers requires frequent table look-ups for obtaining the necessary information on cross sections and on the geometry of the detector. With sufficient local memory, one would want to replicate the information in the tables through all the processors, in order to speed the look-up procedure. In practice, for a realistic simulation this is not possible. One faces again the need of good global interprocessor communications. At the end, the efficiency of the code will depend on achieving a good compromise between how much information is replicated locally and how much is accessed through interprocessor communications. From this point of view, lower granularity is an advantage (i.e., a lesser number of processors with more local memory attached to each processor), although the number of processors should not be too low, in order not to forfeit the gains derived from massive parallelism.

In conclusion, the problem studied by Wilson and collaborators represents one of those cases where a massively parallel architecture, and in particular a SIMD machine, would have *a priori* appeared the least suitable for achieving good computational efficiency. Yet their experience shows that the massively parallel paradigm is indeed applicable also to this class of problems.

5. Conclusions and Outlook for the Future

I have illustrated in the previous section only a small part of the computational research that scientists from many different departments at Boston University have been doing on the Connection Machine. We consider our experience with massively parallel computing very positive. We have found that the range of possible applications met, indeed exceeded, our expectations. Of course, as I have explained above, for some problems the implementation of the code on our massively parallel machine has been more straightforward, for others more demanding, requiring in some cases a rethinking of the algorithms and, by necessity, a substantial rewriting of code.

In an academic environment, having to rewrite code in order to take advantage of a special architecture is not so much of a problem as it may be in other institutions, whose scientists may have to rely on large bodies of well tested software to carry out their work. In a university there are always students and young researchers who have the time and the eagerness to experiment with new things, and the opening of new avenues in computation owes much to their enthusiasm. As I say so, I realize that I may not be doing justice to those more senior researchers and faculty members, who also contribute with enthusiasm to the development of new computational methods: let me reconcile my former statement with the credit they deserve by saying that they are also young, in spirit. But the fact remains that, in order to take full advantage of massive parallelism, frequently one needs to rewrite large portions of the code. However, the programming languages are evolving naturally, irrespective of the changes in hardware, in a direction that makes massively parallel programming natural. The new version of Fortran incorporates many parallel constructs. The same standardization has not yet occurred with other languages, like C, but the work that the manufacturers of massively parallel machines have been doing to extend high level languages so as to incorporate massively parallel instructions and data structures will soon lead to a standardization in other top programming languages as well. Thus, we would expect that, whereas today most code is typically written in a scalar mode² and parallel code is the exception, a few years from now most high level software will be written with parallel constructs, so that scalar code will then be the exception. This is to say that the reformulation of software, that today may seem unduly costly in programmers' efforts, will happen anyway, so that, at least in considering new projects, one should approach the design of code with the parallel paradigm in mind.

How about hardware developments? What can we expect for the future? As several manufacturers prepare to enter the market with their new products, they guard of course their design secrets, so we cannot anticipate with certainty what lies ahead. However, we can ask ourselves how we would design the next machine, what would we want of it, on the basis of our experience as computational scientists working on a massively parallel architecture. Obviously we will want more memory and more CPU-power. We would try to obtain the latter not so much from progress in the speed of the individual components, which may not increase by more than a factor of two, but from having more processors instead. This is after all the basic idea of massive parallelism. It is much cheaper to build ten processors than to make a single one compute ten times faster (if that is possible at all). This progress is rather straightforward from a technological point of view and its pace will be determined mostly by economic considerations (the falling price of memory and processor chips).

²Even if code is written for a vectorized machine, the code constructs are still mostly scalar, with the vectorization of the inner loops done automatically by the compiler. Even in this case, one frequently has to redesign the calculational strategy and to rewrite portions of the code in order to let the compiler exploit vectorization.

More relevant are issues of architecture and communication. On the basis of some applications, we would argue that too fine a granularity is not always suitable. So it would be better to achieve 100Gflops, say, with 1000 rather than 10,000 processors, if possible. Some amount of independence in the code run by the processors would also be useful. Think for instance of the application to particle shower simulation. It would be advantageous to be able to accommodate branches to different sets of instructions as the particles are evolved. This would give more flexibility in adapting the code to a complex geometry and, indeed, would also make it possible to evolve simultaneously particles of different kinds, thus reducing the load balancing constraints. This seems to speak in favor of a MIMD (multiple instructions multiple data) machine. But there are clear advantages to a SIMD (single instructions multiple data) architecture as well. Dealing with simple, synchronized instructions is very important at the programming level. Thus, what we would really want is a kind of in-between, a machine that can be programmed as a SIMD computer, but that incorporates, for instance in the response to conditionals, MIMD features as well.

Enough communication capabilities are also crucial. The architecture of the network is not that important, provided it can sustain the amount and the speed of communications (i.e., very good communication between neighbors in 2-dimensional or 3-dimensional spatial geometries and good global communications as well) that we have found so useful in the CM-2.

My expectation is that these design goals will be generally met by the massively parallel computers of the new generation, which will reach the market within a few years of time. Different manufacturers will find different solutions, but they should adhere to these general constraints, if not with the very first new generation machines, with the refinements that will come soon after that. If not, the forces of the market will weed them out. Similarly, the availability of good software will be crucial, and I expect that the companies that will be ultimately successful will be those that will put as much effort, or even more effort, in the development of software (robust high level compilers, good communication software, excellent, well supported libraries), as in the advancement of the hardware itself. This, incidentally, has been the case for many years, not only with massively parallel computers. The time when somebody could come up with great hardware and let the customers develop the software to use it is most likely gone forever. From this point of view our experience with Thinking Machines Corporation has been very good, which augurs well for their future. But I must recognize here that our experience has been essentially limited to Thinking Machines Corporation, so I am not in the position of making any comment about the software development efforts of any other manufacturer. I would assume that all recognize the importance of this point.

How about the limits to massive parallelism? I do not see any! This is of course too bold a statement, inserted here for rhetorical purposes. But the amount of computational power that massively parallel architecture can muster is very large indeed. It may not be possible and not even desirable to put too many processors in a single space, but I see the possibility of distributing the components of a massively parallel computer over different, geographically remote locations, joined by a very powerful network that will allow them to communicate and calculate as a single unit. This would require solving problems of latency and going beyond the paradigm of fully synchronized, massively parallel computation, developing algorithms for parallel, but asynchronous calculations. But the advantages of such distributed, extremely powerful, massively parallel configuration would be very great.

In any event, the new avenues that massively parallel supercomputing has opened lead to very rewarding fields, and at Boston University we are quite proud and pleased to have taken early steps on them.

Acknowledgements

I wish to express my gratitude to Richard Brower, Roscoe Giles and many other Colleagues at Boston University, and to Ilona Lappo of the Center for Computational Science, for invaluable discussions and assistance in preparing the material for my talk. In addition I wish to thank Glenn Bresnahan, David Deutsch, Laura Giannitrapani, Roscoe Giles, Ilona Lappo, Richard Strilka, Pablo Tamayo, Robert Wilson, and my daughter Ariella for their great help in making the videotape used to illustrate my presentation.

References

- [1] W. Daniel Hillis, "The Connection Machine," *Scientific American* **256**, 108-115 (1987).
- [2] R.L.Devaney, *An Introduction to Chaotic Dynamical Systems*, Addison-Wesley Menlo Park, 1985.
- [3] R.L.Devaney, "Chaotic bursts in nonlinear dynamical systems", *Science* **235**, 342-345 (1987).
- [4] A. Melcuk, R. Giles, and H. Gould, "Molecular Dynamics Simulations of Liquids on the Connection Machine," *Computers in Physics*, 311-318, May/June (1991).
- [5] M. Mansuripur and R. Giles, "Simulation of Magnetization-Reversal Dynamics on the Connection Machine," *Computers in Physics* **4**, 291 (1990).
- [6] R. Giles and M. Mansuripur , "Possible Sources of Coercivity in Thin Films of Amorphous Rare Earth – Transition Metal Alloys," *Computers in Physics* **5**, 204 (1991).
- [7] R. Giles, P.R. Kotiuga, and F.B. Humphrey, "Three-Dimensional Micromagnetic Simulations on the Connection Machine", *J. Appl. Phys.* **67**, 5821 (1990).
- [8] G. Patterson, R.C. Giles, F.B. Humphrey, "Two Dimensional Non-Equilibrium Dynamics of Domain Wall Motion", *J. Appl. Phys.*, (1991).
- [9] K. Wilson, PRD10, p. 2445 (1974).
- [10] M. Creutz, L. Jacobs, C. Rebbi, *Phys. Rep.* **95**, p. 201 (1983).
- [11] C. Rebbi, ed., *Lattice Gauge Theories and Monte Carlo Simulations*, World Scientific (1983).
- [12] R. Brower, S. Huang, J. Potvin, C. Rebbi, "The Surface Tension of Nucleating Hadrons using the Free Energy of an Isolated Quark," to appear.
- [13] R. Brower, S. Huang, J. Potvin, C. Rebbi, J. Ross, "Numerical Study of Perfect Wetting in Quenched QCD," to appear.
- [14] R. G. Hohlfeld, S. M. Sparagna, and G. v. H. Sandri, "Direct Solution of the Wave Equation using Parallel Computational Techniques", 7th Annual Review of Progress in Applied Computational Electromagnetics, Naval Postgraduate School, Monterey, CA, 18-22 March 1991
- [15] E. Myers, C. Rebbi, R. Strilka, "A Study of the Interaction and Scattering of Vortices in the Abelian Higgs (or Ginzburg-Landau) Model," Boston University Preprint BUHEP-91-7 (1991).
- [16] P. Tamayo, J.P. Mesirov, and B.M. Beghosian, "Parallel Approaches to Short Range Molecular Dynamics Simulations," *in press*.
- [17] R.C. Brower, C. DeLisis, M. Silverman, G. Vasmatzis, "Large Protein Folding on a Lattice by Exhaustive Search and Simulated Annealing," work in progress.
- [18] J. Banks, R. Brower, "Dielectric Density Approach to Solvent Effects for Protein Simulations," work in progress.
- [19] T.F. Smith, C.M. Stultz, J.V. White, "Protein Predictive Modeling," to appear.
- [20] R. C. Brower, P. Tamayo and B. York, "Parallel Multigrid Algorithm for Percolation Clusters", *J. Stat. Phys.*, **63**, 73, (1991).
- [21] J. Apostolakis, P. Coddington and E. Marinari, Syracuse University preprint SCCS-103 (1991).

5. 並列計算機による遮蔽安全評価用モンテカルロコードMCACEの高速化

高野 誠*, 増川 史洋*, 内藤 俊孝*

川添 明美**, 奥田 基樹**

モンテカルロ法による遮蔽計算コード MCACE の並列計算機への適用性について検討するため、本コードを並列化した後、高並列計算実験機 AP-1000 上で実行し並列化による処理速度向上に関する測定を行った。最大 64 台のセルを使用し、基本的なプログラムフローを変えずに、比較的単純な方法で並列化を行ったが、最大 52 倍の処理速度の高速化が達成された。

高速化を妨げる主要因は、セル数が多い時に問題となる並列化の不可能な部分に要する計算時間、およびランダムウォークを行なう粒子数の少ない時に顕著となる、セルでの待ち時間であることが判明した。

Speedup of Monte Carlo Code MCACE for Shielding Analysis by Parallel Computer

Makoto TAKANO*, Fumihiro MASUKAWA*, Yoshitaka NAITO*

Akemi KAWAZOE**, and Motoi OKUDA**

In order to study the applicability of a parallel computer to Monte Carlo method, a Monte Carlo radiation shielding analysis code, MCACE, is examined and modified to execute on an experimental highly parallel computer, AP-1000. Despite the simple modifications without changing the original program flow, the MCACE code has achieved a speedup of 52 times on the AP-1000 equipped with 64 cell processors.

The main causes of deteriorating the performance are non-parallelizable part of the codes which affects more with the increment of cell processors, and an idle time of cell processors which becomes significant when the number of particles is small.

Keywords; Monte Carlo, Parallel Computer, Shielding Calculation, Criticality Calculation,

Speedup, Efficiency, AP-1000, KENO IV, MCACE

* 日本原子力研究所 東海研究所 (Tokai Research Est., Japan Atomic Energy Research Institute)

** 富士通（株）計算科学研究部 (Computational Science Research Laboratory, Fujitsu Limited)

1. 緒言

近年、計算機の発達により原子力の分野に於ても、モンテカルロ法に基づく解析計算が可能になってきた。特に、複雑な幾何形状を対象として取り扱う臨界計算や遮蔽計算にはこの方法は有効であると考えられている。現在使用されている第4世代の計算機の出現により、臨界計算をモンテカルロ法で行うことが実現可能となった⁽¹⁾。モンテカルロ計算における標準偏差は、試行回数の平方根に反比例するが、臨界計算においては積分量である固有値の標準偏差が問題となるのに対して、遮蔽計算においては着目する場所の線量率の標準偏差が問題となる。後者の標準偏差は、前者のそれよりも通常桁違いに大きな値となる。それゆえ、遮蔽計算においては臨界計算の場合に比べて多くの試行回数を要求される。現在の世界最高水準のスカラー計算機を使用しても、モンテカルロ法により十分満足できる遮蔽解析を行うことは容易ではない。また、ベクトル計算機用にモンテカルロ法のアルゴリズムをベクトル化する試みも行われているが、飛躍的な計算速度の改善は見られていない。一方、臨界計算や遮蔽計算で解いているボルツマン輸送方程式では、体系中に存在する中性子や光子の粒子に対し、相互干渉は無視できるものとして導出されている。通常、この近似は成立するため、ある粒子のランダムウォークは他の粒子のランダムウォークと互いに独立であり並列計算機向きであるといえる⁽²⁾。第5世代の計算機の開発の結果、Table 1⁽³⁾に示す様に実用的な並列計算機の開発が国内でも多く進められている。さらに、1991年から開始された第6世代の計算機の開発においても、超並列計算機が一つの目玉になっており、高性能の並列計算機が近い将来出現することが期待される。このような計算技術発展の展望にたって、モンテカルロ法による遮蔽計算コードの並列計算機への適応性について、富士通（株）で開発中の高並列計算実験機AP-1000^(4, 5)を使用して検討する。遮蔽計算コードとしては MORSE-CG⁽⁶⁾ コードを改良した MCACE コード⁽⁷⁾を対象とする。

2. 並列計算機

2. 1 高並列計算実験機 AP-1000^(4, 5)

並列計算機を利用するためのプログラムの基本的なアルゴリズムは計算機に対し本来独立であり一般性のあるものであると考えられる。しかしながら、並列計算機用コンパイラも未だ開発中であり、さらに各種並列計算機の標準化も今後の課題となっている現状においては、アプリケーションソフトウェアの並列化に際し、使用する並列計算機の機種を特定する必要がある。そのため、本報では最近

開発された富士通社製の分散メモリ型高並列計算実験機AP-1000を使用するものとした。

数値計算等の高速処理と並列ソフトウェア研究開発環境の実現を目的とした高並列計算実験機AP-1000は、最大1024台の汎用マイクロプロセッサ（セル）を使用可能なMIMD (Multi Instruction Stream Multi Data Stream)方式の並列計算機である。各セルは、25MHzで動作する32ビットRISC (Reduced Instruction Set Computer) を使用し、16MBytesのメモリを有する。各セルは単体で、15MIPS及び8.5MFlopsのピーク性能を有している。各セル間およびホストとセル間は、Fig. 1に示すように3種のネットワークで接続されており、高速かつ自由度の高いデータ通信が可能となっている。まず、ホストと全セルは階層構造状のコモンバスをリング状に接続した50MBytes／秒のデータ通信能力を有するプロードキャストネットワーク（Bネット）で接続されており、ホストからセルへの放送を基本にした効率の良い通信を可能としている。また、セルとセル間のデータ通信には、25MBytes／秒の能力を有する2次元トーラス状接続ネットワーク（Tネット）が使用されている。Tネットでは、各セルは隣接する4つのセルと格子状に接続されており、自動ルーティング機能により、隣接セルだけでなく遠隔のセルとも高速な通信が可能である。さらに、木構造をした同期およびステータス検知のための同期ネットワーク（Sネット）が使用されている。

AP-1000の基本ソフトウェアは、ホスト側とセル側の両方に搭載されている。各セルにはAP-1000独自のオペレーティングシステムが搭載され、ホスト側にはAP-1000ドライバと呼ばれるシステムがUNIX上で動作している。これらの基本ソフトウェア上で、並列計算を実現するためのシステムライブラリが呼び出され、ユーザプログラムが実行される。また、通常のスカラー計算機やベクトル計算機の場合と異なり、ホスト用とセル用に並列性を考慮して別々にユーザプログラムを作成する必要がある。並列性の記述の際、特に重要となるセルとホスト間およびセルとセル間の通信は、送信、受信および同期等の機能を持つAP-1000用サブルーチンライブラリを呼び出すことにより実現される。

2. 2 並列計算機とベクトル計算機

並列計算機の利用方法について、従来のベクトル計算機の利用方法との関連から概略検討してみる。ベクトル計算機では通常繰り返しループに着目したベクトル化を行い、ベクトル化演算処理を行っている。したがって、ベクトル計算機では繰り返しループ単位での高速化が行われており、これを、繰り返しループ内の演

算が並列に行われているものと解釈することも出来る。但し、ベクトル計算機ではCPUは通常一台のみであり、一種類の演算を異なる複数のデータに対し同時に実行することは出来るが、異なる演算を同時に行うこととは出来ない。一方、MIMD型並列計算機では、各セルが独自のメモリを有する単体の計算機と考えることが出来、ホストの制御下において各セルで独自の計算を行える。繰り返しループに対し並列計算機を適用すれば、繰り返しループ内に存在する条件や分岐を含めた総てのプログラムをベクトル処理したのと同等であり、ベクトル計算機より自由度の高い処理が可能となる。両計算機による繰り返しループ処理の様子をFig. 2に示す。

3. プログラムの並列化

3. 1 プログラムフローの解析

最適な並列化アルゴリズムを策定するため、MCACEコードのプログラムフローの解析を行った(Fig.3)。MCACEコードでは、初期に一回だけ行われる入力データ処理、形状モデルの幾何学的チェックや各種初期値の設定を行った後、世代ループと粒子ループの二重入れ子ループに入り、このループ中でランダムウォークの計算が繰り返し行われている。遮蔽問題では一般的にソース(線源)の分布は既知であり、遮蔽計算コードには入力データで線源分布を与え、この分布は通常、計算中に変化するものではない。

3. 2 実行時間の分析

先のプログラムフローの解析から、ランダムウォーク計算部分が両コード中で並列化可能な部分であることが分かった。それでは、実際に並列化を行った際に、どの程度の並列化効果が得られるかを概略検討するため、サンプル問題をスカラーカルcul機上で実行し計算時間の測定を行った。全CPU時間に対する、並列化可能なランダムウォーク計算部分に使用されたCPU時間の割合をTable 2に示す。同表から、ランダムウォーク計算部分が両コードで使用されるCPU時間の大部分を占めており、この部分の並列化により、かなりの処理速度向上が期待できるものと考えられる。ここで、ランダムウォーク計算部分の並列化処理が理想的に行われたと仮定し、セル台数(n)に反比例してランダムウォーク計算部分の処理時間が短縮($1/n$)されるとすれば、8世代、100粒子/世代のMCACEサンプル問題を全10セルの並列計算機で実行した時には、ランダムウォーク計算部分の処理に必要な計算時間が $1/10$ の約9.8%となり、並列化できない部分の処理時間2%を加えて全

体では11.8%に減少することとなる。これは、Table 2に示すように約8.5倍の速度向上に相当するさらに、この表から並列化の困難なランダムウォーク計算以外の部分（シリアル計算部分）で消費されるCPU時間が増加すると、並列化による速度向上が急激に悪化することが分かる。

ここで、スカラー計算機で実行時間の計測を行った時、並列化の困難なシリアル部処理に要するCPU時間が t_s 、全CPU時間が t であったとすると、セル台数が n 台の並列計算機により期待される理想的な速度向上率(f)は次式で示される。

$$f = \frac{t}{(t-t_s)/n + t_s} \quad (3.1)$$

ただし、並列処理部分の実行時間はセル台数 n に反比例する理想的な場合を想定している。(3.1)式の両辺を n で除して整理すれば、

$$\frac{f}{n} = \frac{1}{(n-1)t_s/t + 1} \quad (3.2)$$

となる。(3.2)式について f/n をパラメタとして t_s/t と n に関しプロットしたのがFig. 4である。この図から、例えばセル総数が1000台の並列計算機で、 $f/n=0.8$ つまり800倍の速度向上を達成するためには、 t_s/t は 2.5×10^{-4} 以下である必要があることを示している。つまり、シリアル処理に要するCPU時間が全体の0.025%以下である必要がある。このことは、モンテカルロ計算が大規模なもので莫大な粒子数のランダムウォークを必要とする時程、セル台数の多い並列計算機がその性能を十分に発揮できることを意味している。

3. 3 並列化アルゴリズムの策定

前節までの検討で、ランダムウォーク計算部分を並列処理すれば、かなりの並列化効率を期待できることが分かった。ここで、両コードがモンテカルロ法で解いているボルツマン輸送方程式は、体系中に存在する中性子や光子の粒子に対し、相互干渉は無視できるものとして導出されている。通常、この近似は成立するため、ある粒子のランダムウォークは他の粒子のランダムウォークと相互依存がないとして、この計算部分を並列処理の対象としても計算結果に不都合は生じない。したがって、最も単純な並列化アルゴリズムは、各セルで一個づつ粒子のランダ

ムウォーク計算を行い、全セルの計算終了を待ってホストはセルで行われたランダムウォーク計算結果を取りまとめてから、次の粒子を再び各セルに与えてランダムウォークの計算を再び行わせるという方法である。しかし、このアルゴリズムの欠点は、ランダムウォーク計算を他より早く終了したセルは、他のセルが総て計算を終了し、さらにホストが結果を処理した後でないと次の粒子の計算を開始できず、この間セルは待機（アイドル）を余儀なくされ並列化効率低下の原因となる点である。したがって、上記の問題を回避するためには、各セルで一回に計算を行う粒子数を極力多くして、ホストとのデータの授受回数を低減し、さらにランダム過程で発生する各セル間の終了時間のバラツキを粒子数を増加させることで最小化する必要がある。

ここで、現在では一般的となっているベクトル計算機を効率良く使用するためにはベクトル長を極力大きくする必要があったが、並列計算機を使ったモンテカルロ計算ではランダムウォークの計算対象となる粒子のセル当たり個数が多い程、並列化効率は向上するものと考えられる。つまり、各セルがホストとの通信をせずに独立して計算を行える最大の粒子数が並列計算機でのベクトル長に対応するものと考えられる。

単位セル内最大粒子数 P_{\max} は、両コードの入力データである世代数 (g)、粒子数 (p) およびセル数 (n) に依存し、

$$P_{\max} = g \cdot p / n \quad (3.3)$$

で示される。Fig. 5 にMCACEコードの並列化アルゴリズム概念図を示す。

3. 4 乱数発生部の並列化

モンテカルロ計算では乱数は重要な役割を果たしており、通常のスカラー計算機では乱数列から順番に一個づつ必要に応じてランダムウォークの計算に使用される。しかしながら、並列化されたモンテカルロコードでは並列演算を行っているセル群が同時に互いに異なる複数個の乱数を要求することになる。この時、セルの要求に応じ乱数をホストで発生させ一つづつ与えていたのでは、ホストが乱数発生で飽和てしまい、そのためセルのアイドル時間が増大し並列演算効率が悪化する可能性がある。したがって、乱数をホストではなく各セルで独立に発生させる方が良く、ここでは、合同乗算法に基づく乱数発生を次のようにして並列化を試みた。

まず、通常の合同乗算法では、 j 番目の乱数 x_j は次式で得られる、

$$x_j = \text{mod}(2^k, \lambda x_{j-1}) \quad (3.5)$$

ここで、 x_{j-1} , k および λ はそれぞれ $j-1$ 番目の乱数、使用中の計算機のビット数およびユーザーが入力した定数である。乱数を各セル内で独立に発生させるため (3.5) 式を変形し、

$$y_{ij} = \text{mod}(2^k, \lambda^n y_{i,j-1}) \quad (3.6)$$

ここで、 $y_{i,0} = \lambda^i x_0$ 、また y_{ij} は i 番目のセルに対する j 番目の乱数である。また、 n はセル台数である。(3.6) 式は基本的に (3.5) 式と同一であり、乗数項 (λ) の指数のみが異なっており、(3.5) 式と同一のアルゴリズムを用いて各セルで乱数を発生させることができる。また、(3.6) 式で発生させた各セルでの乱数は、(3.5) 式の乱数列をセル台数分 (n) スキップして得られたものと同一で、全セルで見れば、(3.5) 式の乱数列を再現するようになっている。ただし、ここで(3.6) 式の初期値として使われる (3.5) 式で生成された乱数列の最初の n 個は除く。また、(3.6) 式を使用するためには、合同乗算法計算部分のプログラムの変更を必要とはせず、 λ^n および $y_{i,0}$ をホストで初期値として一回求めてセルに与えればよいことになる。このような方法で、乱数を各セルで同時に発生することが出来る。

4. 並列性能評価と問題点（並列版 MCACE コードの処理速度測定）

並列化した MCACE コードに対し、セルを 64 台実装した AP-1000 上でサンプル問題を実行し処理時間の測定を行った。AP-1000 で使用するセル台数を、1、2、4、8、16、32 および 64 台とした場合のそれぞれに対し処理時間の測定を行い、セル台数による処理速度変化について検討した。サンプル問題としては、「 ^{60}Co ガンマ線源によるスカイシャイン問題^⑨」を使用し、ランダムウォークの試行を行う粒子数と処理効率についてさらに検討するため、バッチ数（世代数）と粒子数をそれぞれ 128 バッチ、200 粒子／バッチ（全 25600 粒子）のケースと 128 バッチ、400 粒子（全 51200 粒子）のケースについて検討した。

ここで、 n 台のセルで実行したときの処理速度倍率 (S_n) および並列効率 (U_n)

を次式で定義する、

$$S_n = T_1/T_n \quad (4.1)$$

$$U_n = (T_1/n)/T_n \quad (4.2)$$

ここで、 T_1 と T_n はそれぞれセル 1 台および n 台のときのジョブ処理時間である。測定結果を Table 3 と Table 4 に示す。また、速度倍率 (S_n) および並列効率 (U_n) のセル台数 (n) による変化を Fig. 6 に示す。

測定の結果、速度倍率 (S_n) はセル台数の増加に伴い増大し、200 粒子／バッチでは 64 セルの時 50.13 倍、400 粒子／バッチでは 52.53 倍にまで達している。ただし、並列効率 (U_n) を見れば明らかのように、セル台数の増加とともに効率が少しずつ悪化して行くのが分かる。この点は、定性的には先に第 3.2 節で述べたように、(3.2) 式の速度向上率 (f) により理解できる。つまり、セル台数が多くなってもシリアル計算部分に要する処理時間は変化せず一定であり、セル台数の増加とともに相対的に全処理時間に占める割合が大きくなるため、全体としての並列効率が悪化する訳である。400 粒子／バッチと 200 粒子／バッチの両ケースの差についても同様の議論が出来、全処理時間に対して 400 粒子／バッチのケースの方が総粒子数が多いため、全処理時間に対してシリアル計算部分の割合が相対的に小さくなり、200 粒子／バッチのケースより並列効率が良くなると考えられる。

ここで、200 粒子／バッチの場合について測定された速度倍率 (S_n) を (3.1) 式のシリアル部処理時間を考慮した理想的な速度向上率 (f) と比較してみると、Table 5 に示す様に、セル台数の増加と共に両者の差が増大している。これは、速度向上率 (f) の計算で用いた「並列処理部の実行時間がセル台数 (n) に反比例する」という仮定が成立していないことを意味している。この仮定が成立しない原因として以下の場合が考えられる。

(1) ランダムウォーク計算のバラツキ

各セルで行なわれているランダムウォーク計算時間は必ずしもすべて等しい訳ではなく、乱数を使用しているためバラツキがある。そのため、与えられた粒子数は同一であっても、セル毎に全ランダムウォークを終了する時間が異なり、他より早く計算を終了したセルでは全セル終了まで待機（アイドル）することになり、並列化効率を悪化させる原因となる。

(2) 並列化に伴い発生したセルでの処理

スカラー計算時には不要であった処理が並列化に伴い発生する。これは、ホストから初期値を受け取り、対応する変数へセットするための処理が含まれる。

また、ホストとセル間で行なわれるデータ通信に要する時間もスカラー計算ではなかったもので、並列計算特有のものである。しかしながら、この通信に要する時間は、AP-1000（64セル）上で測定した全処理時間の0.1%以下であり問題とはならない。さらに、(2)についても全処理時間の約0.3%であり特に問題となる程ではない。一方、(1)についてはセル数が増大し、セル当たりに割り当てる粒子数が少なくなるとその影響が増大する。例えば、64セルの場合には、与えられた全粒子のランダムウォーク計算に要する時間の最短と最長の差は5.71秒であり、Table 3に示した全処理時間に対し20%程度を占め、並列倍率(S_p)が理想的な速度向上率(f)より小さくなる主要因であることがわかる。これをある程度回避するためには、セル当たりの粒子数を増加させればよいが、完全には回避できないものである。この点についてさらに並列倍率を改善するためには、バッチ数および粒子数を入力で与えてモンテカルロ計算を行なうという現在のアルゴリズムを変更し、例えば、全セルのランダムウォーク計算終了を粒子数ではなく全セル同一の実行時間で規定すれば、セルでのアイドルは発生せず、効率はさらに向上するものと考えられる。この場合には、各セルでは与えられた実行時間中に可能な限り多くの粒子のランダムウォークを行なえばよく、セル台数に等しいバッチ数の計算が行なわれると考えればよい。したがって、各バッチ（各セル）での粒子数は一定とはならないが、最終的な計算結果に対しては、何ら不都合は生じない。

ここで、富士通社の超高速スカラー計算機M-780/20で400粒子／バッチのケースを実行すればCPU使用時間は538秒であり、64セルのAP-1000で実行したときの55、3秒に比べ約1／10の処理速度であることが分かる。

5. 結論

モンテカルロ法による遮蔽解析コードMCACEの並列化を行った後、高並列計算実験機AP-1000上で実行し、並列化による処理速度向上に関する測定を行った。最大64台のセルを使用し、基本的なプログラムフローを変えずに、比較的単純な方法で並列化を行ったが、MCACEコードでは最大52倍の処理速度の高速化

が達成された。このことから、超並列計算機を利用した、モンテカルロ法による高速遮蔽計算により、大規模な遮蔽問題を精度良く解析できる見通しが得られた。

高速化を妨げる主要因は、一定の粒子数に対するランダムウォーク処理に要する実行時間が各セルで異なり、このバラツキにより発生するセルでのアイドル時間であることがわかった。

並列版MCACE コードでは、ほぼ最適な並列化がなされているが更に高速化を達成するために、バッチ数と粒子数による繰り返しループを除去し、セル台数をバッチ数と考え、また各セルでは与えられた計算時間内で可能な限り多数の粒子のランダムウォークを行わせてセルのアイドル時間を極力排除すれば、更に一層の高速化が期待できる。

今後は、MCACE コードに関し、さらにセル台数の多い並列計算機での高速化の検討や、MCNP コード等の並列化についても検討を行う予定である。また、モンテカルロ法による炉心3次元炉心出力計算及び動特性計算も並列計算機による高速処理で実現可能な射程に入ったものと考えられる。一方、多数の並列計算機をネットワーク上で接続し超並列計算機として利用することも考えられ、今後の分野の展開が大いに期待される。

参考文献

- (1) 科学技術庁原子力安全局編：「臨界安全ハンドブック」、にっかん書房(1988).
- (2) Wood J., Al-Bahadili H., Khaddaj S.A., "A Comparison of Monte Carlo Photon Transport on Two Types of Parallel Computer", Ann. nucl. Energy, 18[3]155~166 (1991).
- (3) 小柳滋、田辺昇：超並列マシンの実現技術、情報処理、32[4]369 (1991).
- (4) Ishihata H., Horie T., Inano S., Shimizu T., Kato S., :"An Architecture of Highly Parallel Computer AP-1000", IEEE Pacific Rim Conference on Communications," Computers and Signal Processing, May 9-10 (1991).
- (5) Horie T., Ishihata H., Shimizu T., Kato S., Inano S., Ikesaka M., :"AP-1000 Architecture and Performance of LU Decomposition", 1991 International Conference on Parallel Processing, Aug. 12-16 (1991).
- (6) Straker E.A.,et al.: "The MORSE-Code with Combinatory Geometry", DNA-2860T(1972).
- (7) Yamano N., Minami K., Koyama K., Naito Y., :"RADHEAT-V4: A Code System

to Generate Multigroup Constants and ASnalyze Radiation Transport for
Shielding Safety Evaluation", JAERI-1316 (1987).

Table 1 Developments of parallel computers in Japan

Model	Organization	No. of processors
APP-2	NTT	65536
MAPLE	Fujitsu	65536
CAP	Mitsubishi Elec. Co.	4096
PIM/p	ICOT, Fujitsu	512
R256	NTT	256
ADENA	Matsushita Elec. Co.	256
Prodigy	Toshiba Co.	512
.....	Kyushu Univ.	128
QCDPAX	Tsukuba Univ.	480
AP-1000	Fujitsu	1024
SPA	Okayama Univ. of Sci.	256
SIGMA-1	Electrotechnical Lab.	256
EM 4	Electrotechnical Lab.	1000
EDDEN	Sanyo Elec. Co.	1000

Table 2 CPU costs of random walks

MCACE			
Generations (batches)	8	64	
Particles/generation	100	200	
CPU cost *1 [%]	98.0	99.8	
Ideal speedup factor *2			
(10 cells)	8.47	9.82	
(100 cells)	33.6	83.5	
(1000 cells)	47.7	333	

*1 CPU cost = (CPU time for random walk)/(total CPU time)

$$\text{*2 Ideal speedup factor} = \frac{100}{\frac{\text{CPU cost}}{n} + (100 - \text{CPU cost})}, \quad n : \text{number of cells}$$

Table 3 Measurement of CPU times (1)
(128 batches, 200 particles/batch)

unit : sec.(unless specified)

number of cells	Cell (average) execution	comm. & idle	Host execution	comm. & idle	Total processing time	Speedup factor [times]	Parallel efficiency [%]
1	1457.98	0.42	14.65	1445.89	1460.53	(1.00)	(100.0)
2	726.31	5.26	14.68	719.01	733.69	1.99	99.5
4	363.40	2.09	14.63	352.96	367.59	3.98	99.3
8	182.50	4.25	14.66	173.11	187.77	7.78	97.2
16	91.38	2.28	14.65	81.13	95.78	15.25	95.3
32	45.84	3.51	14.55	36.90	51.45	28.39	88.7
64	22.23	4.24	14.85	14.29	29.14	50.13	78.3

Table 4 Measurement of CPU times (2)
(128 batches, 400 particles/batch)

unit : sec.(unless specified)

number of cells	Cell (average) execution	comm. & idle	Host execution	comm. & idle	Total processing time	Speedup factor [times]	Parallel efficiency [%]
1	2901.45	0.12	14.91	2889.35	2904.26	(1.00)	(100.0)
2	1451.20	0.57	14.61	1439.26	1453.87	2.00	99.9
4	723.12	7.02	14.85	717.67	732.52	3.96	99.1
8	363.59	3.96	14.64	355.04	369.68	7.86	98.2
16	181.88	4.98	14.40	174.58	188.98	15.37	96.1
32	91.40	6.66	14.98	85.19	100.17	28.99	90.6
64	45.84	7.36	14.29	41.00	55.29	52.53	82.1

**Table 5 Measured speedup factor (S_n) and ideal speedup factor (f)
(200 particles/batch)**

No. of cells	Speedup factor	
	measured (S_n)	ideal (f)
1	(1.00)	(1.00)
2	1.99	2.00
4	3.98	3.99
8	7.78	7.93
16	15.25	15.71
32	28.39	30.82
64	50.13	59.40

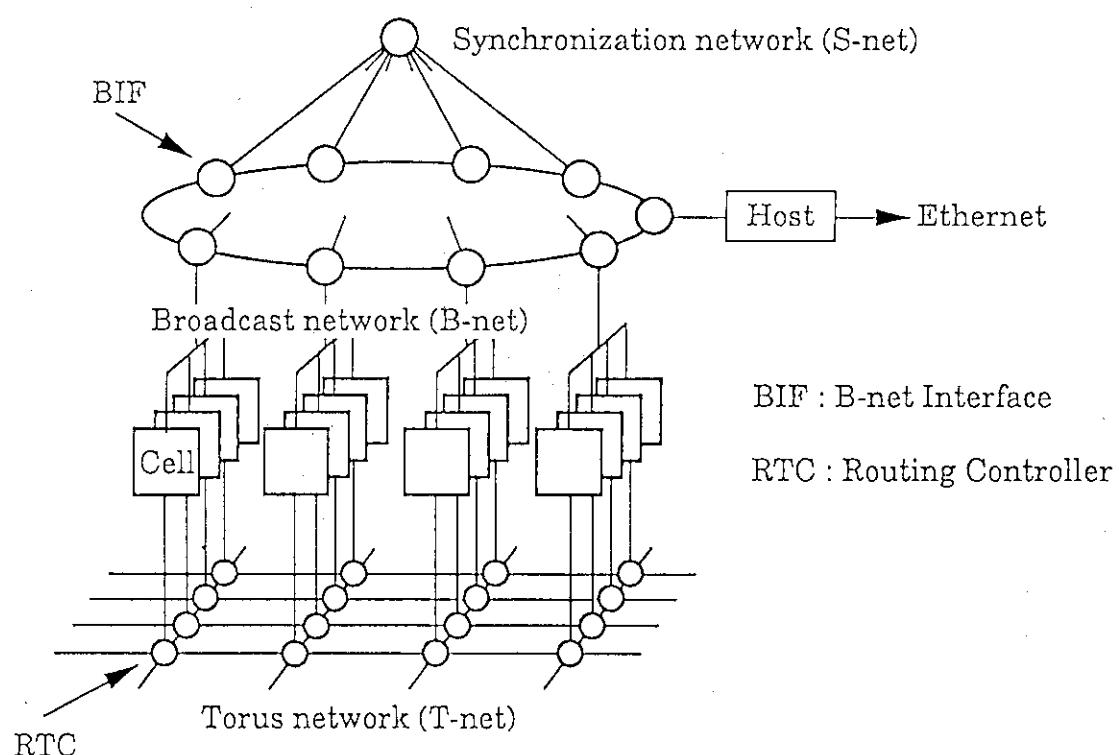


Fig.1 AP-1000 System Configuration

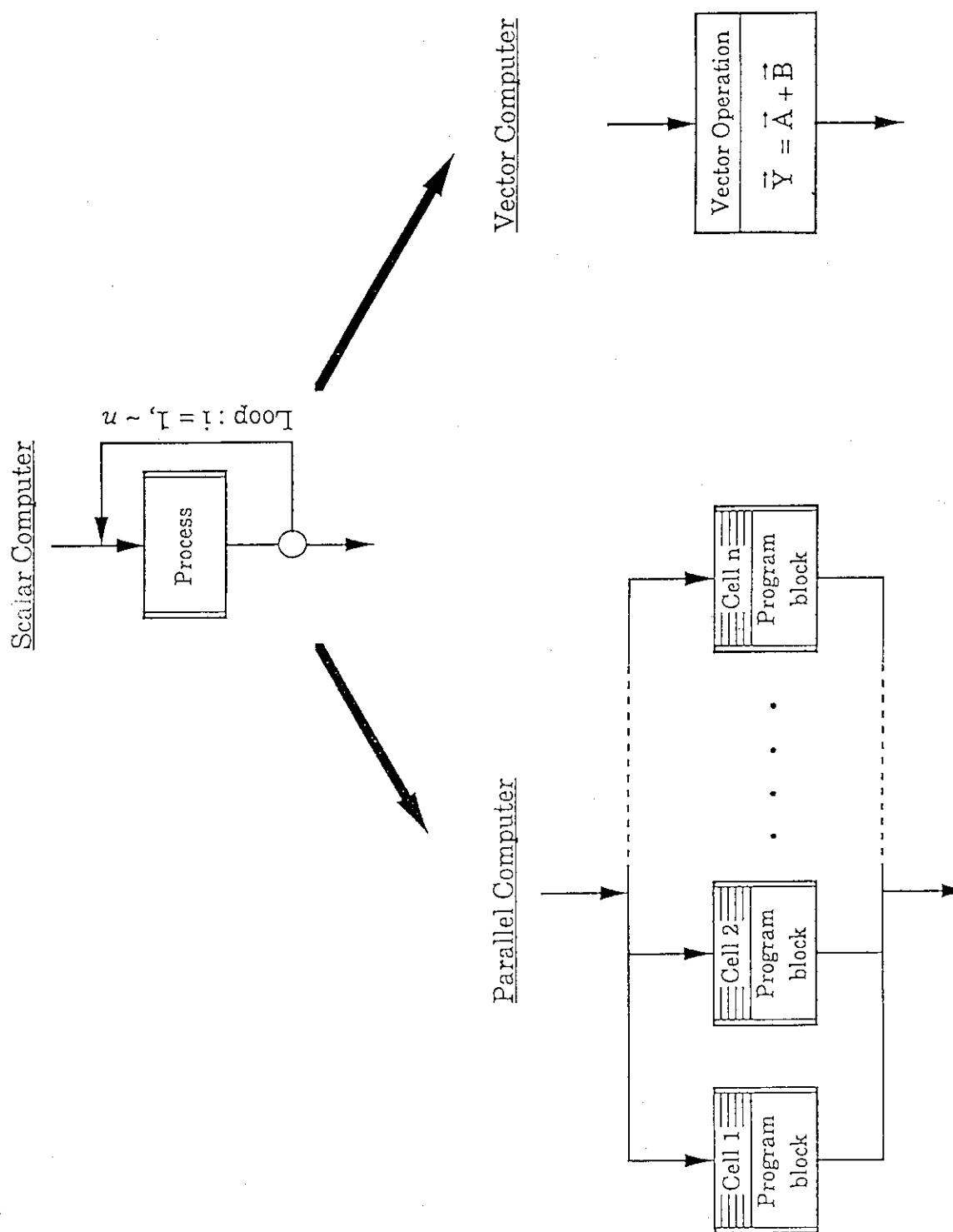


Fig.2 Parallelization and Vectorization

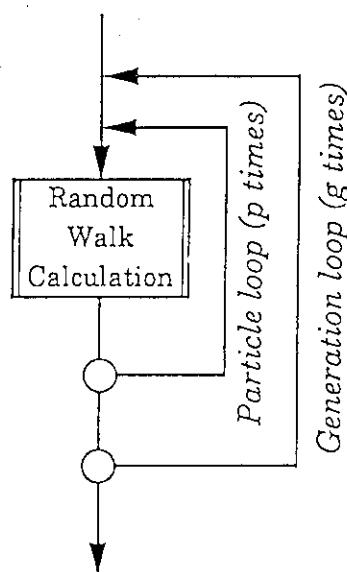
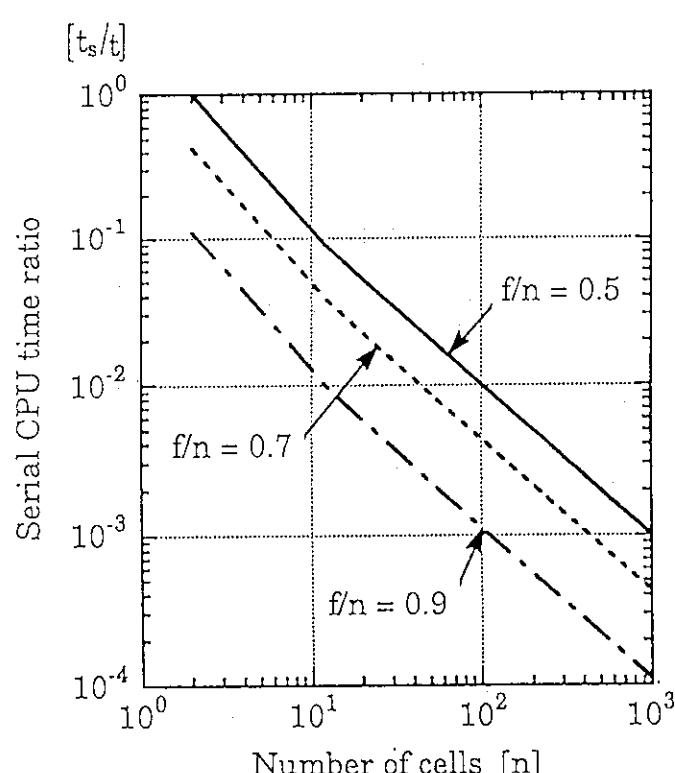
MCACE

Fig.3 Analysis of Program Flow

Fig.4 Deterioration of Parallel Efficiency caused by Non-parallelizable part of program (f/n =Index of Parallel Efficiency)

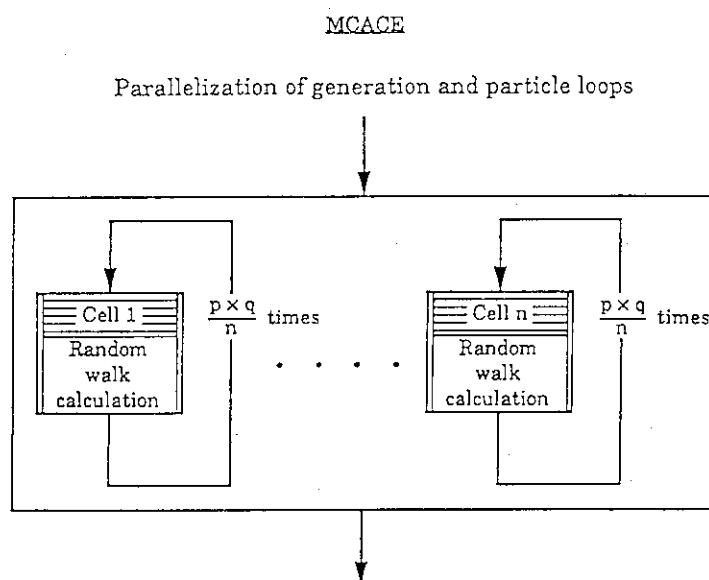
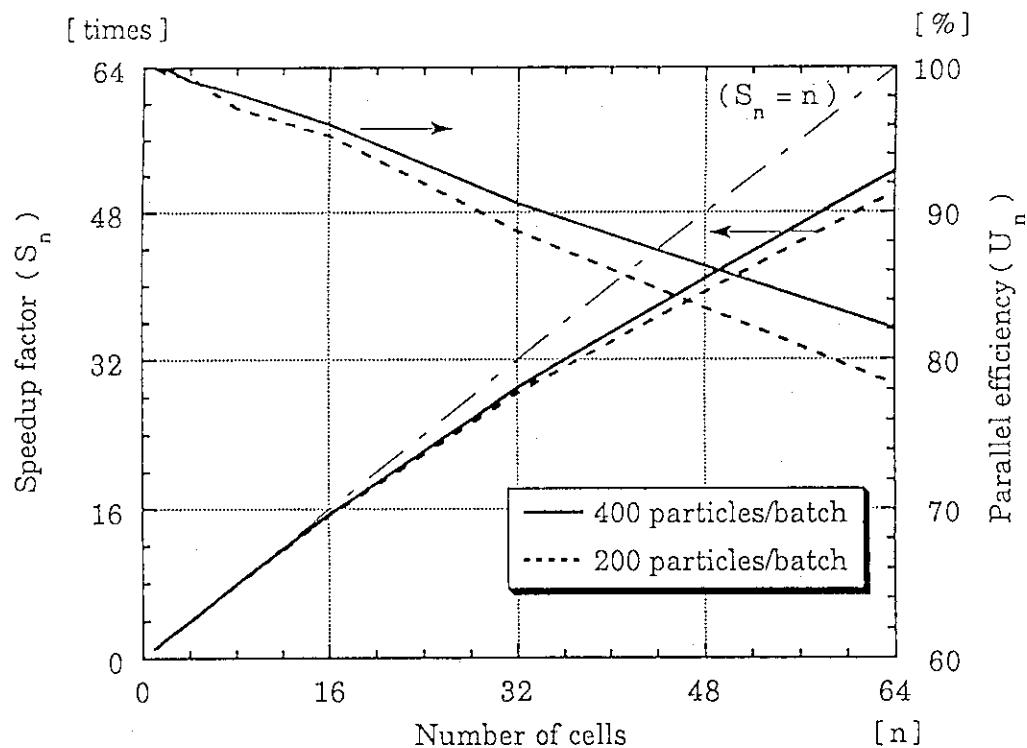


Fig.5 Algorithms for Parallelization

Fig.6 Speedup Factor (S_n) and Parallel Efficiency (U_n) with Number of Cells (n)

6. 知能ロボットにおけるセンサー融合による状況認識 —物理空間法について

Multi-Sensor Fusion Method for Intelligent Robots -Physical Space Method

日本原子力研究所 東海研究所 情報システムセンター

伊藤 伸泰, 藤井 実

Nobuyasu Ito and Minoru Fujii
Computing and Information Systems Center, Japan Atomic Energy Research Institute,
Tokai, Ibaraki-ken, Japan 319-11

Abstract

A new idea for multi-sensor fusion method named *Physical Space Method* (PSM) proposed by one of the present authors (N.I.) is introduced. This method prepares models of the world and the sensors and integrates the signals from the real sensors into the model configuration. A simple application for stereo-view problem is also explained.

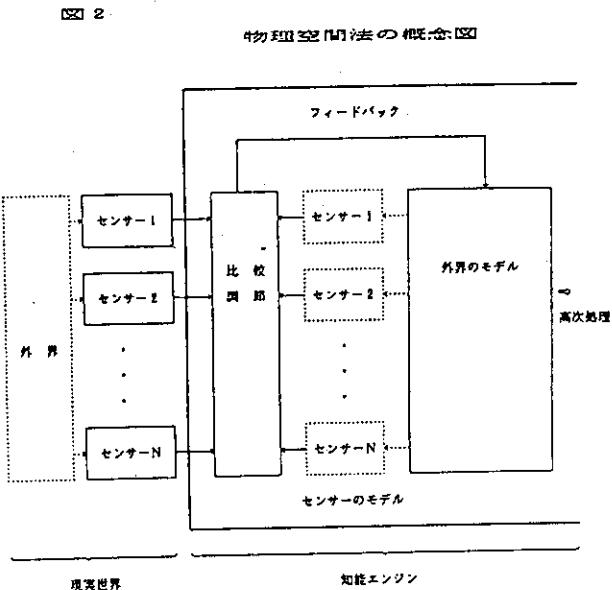
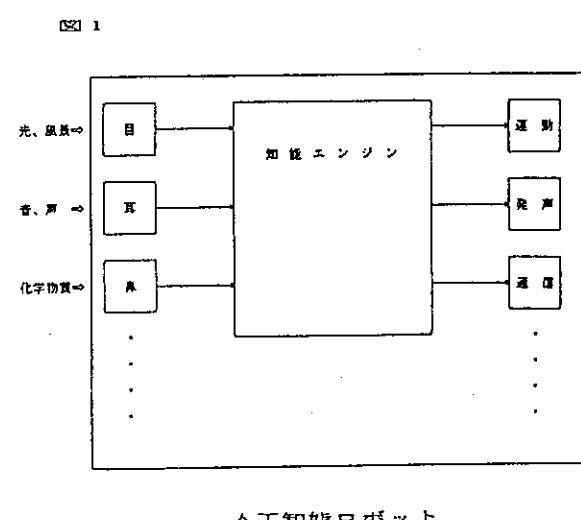
ロボットがどの様に行動するかは外界の状況によって決まる。自分のまわりに何があるか、自分にどの様な命令が与えられているかといったことを判断し行動に移すわけである。つまり、外界の状況判断が必要として十分なだけできる様にならなくては自律型のロボットは実現できないことになる。センサー群からの信号を分析することによって外界の状況を把握するのであるが、個々のセンサーからの信号は外界の対象の側面に限られていることが多い。このため、対象の様子を認識するためには、複数のセンサーからの信号を統合する必要がある。周囲の状況を把握する問題が最も典型的な例である。

本講演で紹介する「物理空間法」は、こういったセンサー融合を行うための新しい考え方である。図1に示した人工知能ロボットの概略を、もう少し詳しく見てみよう。特に、センサー群からの信号に注目する。先にも述べた様に、センサー群からの信号を1つの表現に融合する必要がある。この過程は、知能エンジンの行動決定機能などの高次機能をいかにして実現させるかとも関わっている。物理空間法では、センサー群からの信号に基づいて外界のコピーを知能エンジンの内部につくり、その後に高次の機能にそのコピーを渡すという方法を考えている。

物理空間法(Physical Space Method)でどのようにしてセンサーフュージョンを実現するかここではその考え方を説明しよう。詳細は文献1を参照のこと。まず図2のように外界のモデルおよびセンサーのモデルを用意する。ここで、外界のモデルを調節することにより外界のコピーをつくるのである。すなわち、外界のモデルはモデル変数を含んでおり、それら変数を変化させるとセンサーのモデルが出す信号も変化する。そして、センサーのモデルからの出力信号が現実のセンサーからの信号と同じになる様に外界のモデルの変数を調節するのである。そして、両者の出力が同じになった時、外界の様子が知能システムの中で把握できたと考える。この考え方を応用して、簡単な2眼立体視問題を解くモデルを作ることに成功した。その場合外界のモデルは色の空間的分布であり、比較調節には実センサーとモデルセンサーの差の2乗和をエネルギーとするシミュレーテッドアニーリングを使った。

Reference

- 1: N. Ito, A New Method for Multi-Sensor Fusion-Physical Space Method(PSM), submitted.



知能ロボットにおけるセンサー融合による状況認識

伊藤伸泰 藤井実

日本原子力研究所 東海研究所
情報システムセンター

1 知能ロボットとセンサー融合

自律的に動作するロボットを作ろうとする際に、いかにして外界の情報を得るかということが問題となる。ロボットは適切なセンサーをもち、センサーからの信号に基づいて自分の置かれている環境がどのようなものであるかを知覚する必要がある。そのためにどのようなセンサーをどのように使うかはロボットの作業の目的によってさまざまである。

動作環境が予め人為的に設定できる場合とできない場合とではロボットに要求される機能は大きく異なる。作業環境が十分整備されているところで明確に決められた作業を遂行するロボットは既に実用化されており、さらなる高度化も研究されている。このようなロボットを受動型ロボットと呼ぶことにする。例えば、工場内での製品の組み立てなどを行う産業用ロボットがあげられる。一方、任意の作業環境で高度な判断を伴う作業を行うことができるロボットは未だ研究段階である。このようなロボットを能動型ロボットと呼ぶことにする。本節では以下、このような能動型ロボットについて議論する。

受動型ロボットと能動型ロボットとの違いを明確にするために、原子炉などの複雑で高度なプラントの巡回点検をロボットに行わせることを考えてみよう。この場合、ロボットに要求される作業は、運転中の原子力プラントの中を巡回して異常の有無をチェックし、異常があれば適切な処置をとることである。このような作業を自動化するためには受動型ロボットで十分ではないかという考え方もある、現在のロボット技術に基づく限り、一理ある。チェック場所に必要なセンサーを配置し、リモートセンシングを行なえばよい。センサーをプラント全体に配備するのにコストが嵩むというのであれば、予め設置したガイドウェーに沿ってセンサーを移動させればよいであろう。しかしながら、想定していなかった事態が生じた場合には、こういった方法では適切に対処できないことが多いと考えられる。ある場所の放射線線量が得られればより的確に状況を把握できると考えられるのにその場所にモニターがなかつたり、ある場所の温度に異常が見られるがそれは実際に異常か否か判断がつきにくい場合などである。想定されていた事態であっても、プラントオペレーターが咄嗟に事態を把握仕切れず、自分の理解の範囲での的確な処置をするために、想定していない場所のデータが欲しいということもある。

このような際に、指示された場所にいって指示された作業を行って戻ってくる能動型ロボットがあれば必要なデータを自在に揃えることができる。また、運転に支障のない作業であれば、プラントを停止せずに行わせることができよう。事故などの際には、現場にいってみるまでどのような環境かわからないこともある。リモートコントロールのロボットを使ってもよいが、複雑な作業をこなせる熟練した作業員は数が限られ、しかも一人が操作できるのは一台だけである。このような際には、指示された作業内容を状況に応じて臨機応変に実行する複数の能動型ロボットが有効であろう。

さらに、日常の巡回点検でも、必要に応じてロボットが自分で点検すべき場所、内容を考えて行動すれば、的確な保守ができる。

上述のような能動型ロボットを実際に作ろうとすれば、電子部品の耐放射線対策はどうするか、放射性物質を拡散させる恐れはないかなどさまざまな問題がある。しかしながら、現在、このような

ロボットの実現を考えたときに最も問題となるのは、制御技術である。原子炉内という過酷な環境を考える以前に、オフィス内や街頭といった通常の環境でもこのようなロボットを作ることは難しいのが現状である。制御技術以外の部分で問題となるものに、ロボットの手足の構成があるが、本文では制御部分、とくに、センサーから外界の状況を把握するために必要なセンサー融合について扱い、センサー融合の新しい手法である物理空間法[1]を紹介する。

日本原子力研究所情報システムセンターでは、原子力知能化システム技術の研究として人間動作シミュレーションプログラム（略称 HASP）[2]と題するプロジェクトのもと、能動型ロボット実現に向けた研究を行っているが、物理空間法も HASP の 1 要素として提唱されたものである。以下では能動型ロボットのみを考え、単にロボットと言った場合は全て能動型を指すものとする。

2 物理空間法

ロボットの構成は、図 1 のようなものとなるであろう。このロボットを実現するには現段階ではさまざまな困難があるため、この構成をもう少し細分化して各要素を研究する必要がある。センサー融合もそういった課題のひとつで、センサー群からの信号を統合し外界の状態を再構成するところに相当する。

センサー融合の手法にはさまざまなものがあるが、物理空間法 (physical space method) [1] はロボットが外部環境を認識するのに適した方法として我々が研究を進めているものの 1 つである。

物理空間法の基本的な考え方は、ロボット内部に外部環境のモデルを予め用意しておきモデルのパラメータをセンサーからの情報と矛盾がないように調整して外界のレプリカをつくり、そしてできた世界のレプリカを使って高次の処理を行おうとするものである（図 2 参照）。物理空間法で主に扱うのは、外部環境のモデルの構成方法と外部環境のレプリカをつくる方法である。

ロボットの場合、各センサーから受け取る信号は外界の物理的な様子についての一面的な情報である。例えば、視覚情報を得るために CCD カメラを搭載していたとしよう。外界にある物体により反射された光や光源からの光はカメラの光学系を通って CCD チップ上に結像し、各ピクセルで電気信号に変換されてロボットに送られる。大雑把にいって、3 次元に分布している光をカメラのレンズで切りその面を通る光のうちカメラの向いている方向から来るものの分布だけを信号に変えていくわけである。つまり、ある 1 つのピクセルからの信号の意味するものは、その時のカメラの位置・向きおよび CCD 上のそのピクセル位置によって決まるある方向からの光の強度である。

センサーからのこのような信号に対してまず行うべき第 1 段階の処理は、外界の物理的状態を知ることである。しかるのちに、これはなんであるかといった意味論的な認識を第 2 段階として行うのが妥当であろう。具体的にカメラの例で説明しよう。CCD のピクセルからの信号は面状に並んでいるわけで、各ピクセルからの信号を 2 次元的に配置したものを画像と見なすことが第 1 段階である。そして、そこに写っているものが何なのかを認識するのが第 2 段階である。

現在あるセンサー技術では、1 つのセンサーだけでは外界の様子は正確には把握できない場合が多い。画像上に白い線が細長く写っていたら配管であるとする知識に基づいて外界を認識するとしよう。そこで、画像を白いピクセルからスタートして連続する白い領域を抜き出し、その領域の長径と短径との比率がある値よりも大きかったので配管として認識したとする。しかしながら見たものは実はカメラの直前の白い鉛筆だったかもしれない。そこで物体までの距離を測定する必要があるであろう。そのために 2 台のカメラ、または異なる位置からの画像を使って立体視により距離を求めるとする。その際、複数のセンサーの信号から意味論的あるいは認識論的な情報を直接得ようとするこことは困難であることが多い。そこでまず、2 つの映像から距離の情報をもった 3 次元的な映像を再構成してその 3 次元映像に対して物体の認識を行おうというのが物理空間法によるセンサー融合の考え方

図 1: ロボットの構成の概念図。

実線部分がロボットである。作業環境 (read world) 内の対象 (objects) からの情報はセンサー群で捉えられ、行動決定装置 (IE) におくられる。IE はその情報を基にして次の行動を決定し、外界に対する行動を制御する装置 (controllers) に指示を出す。

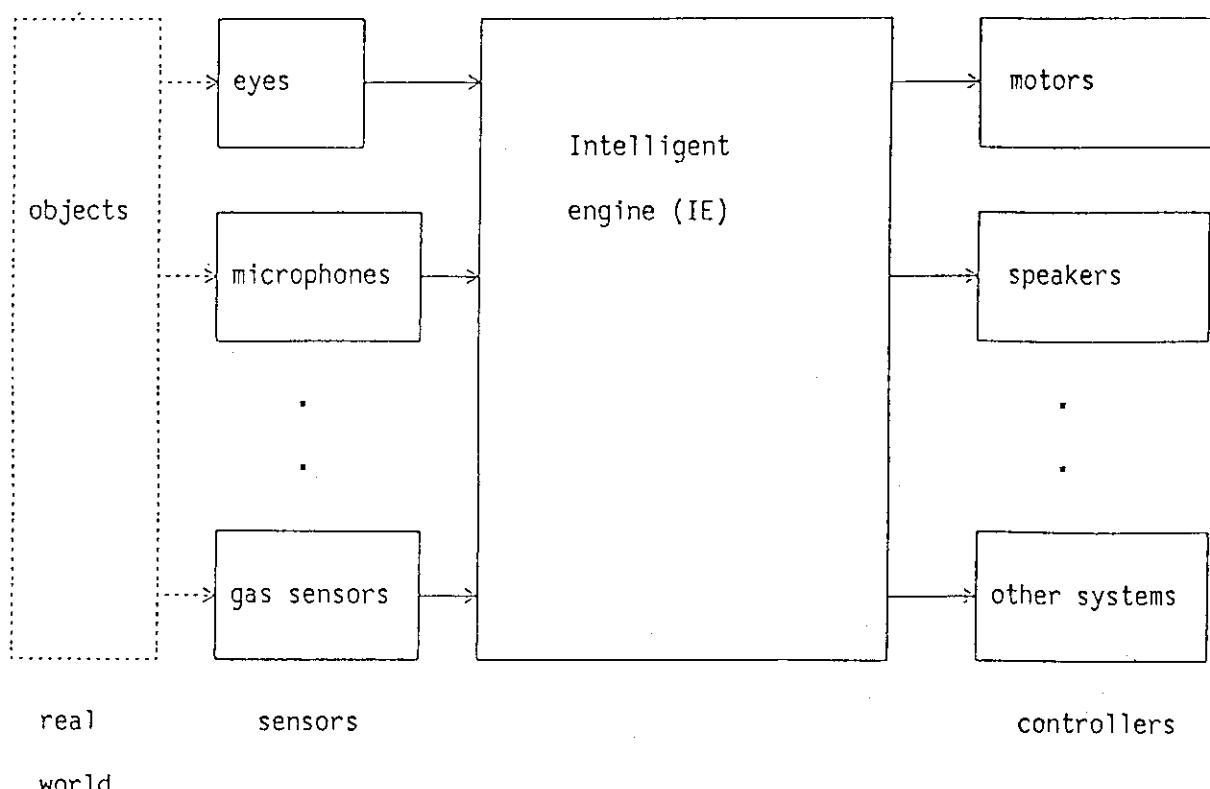
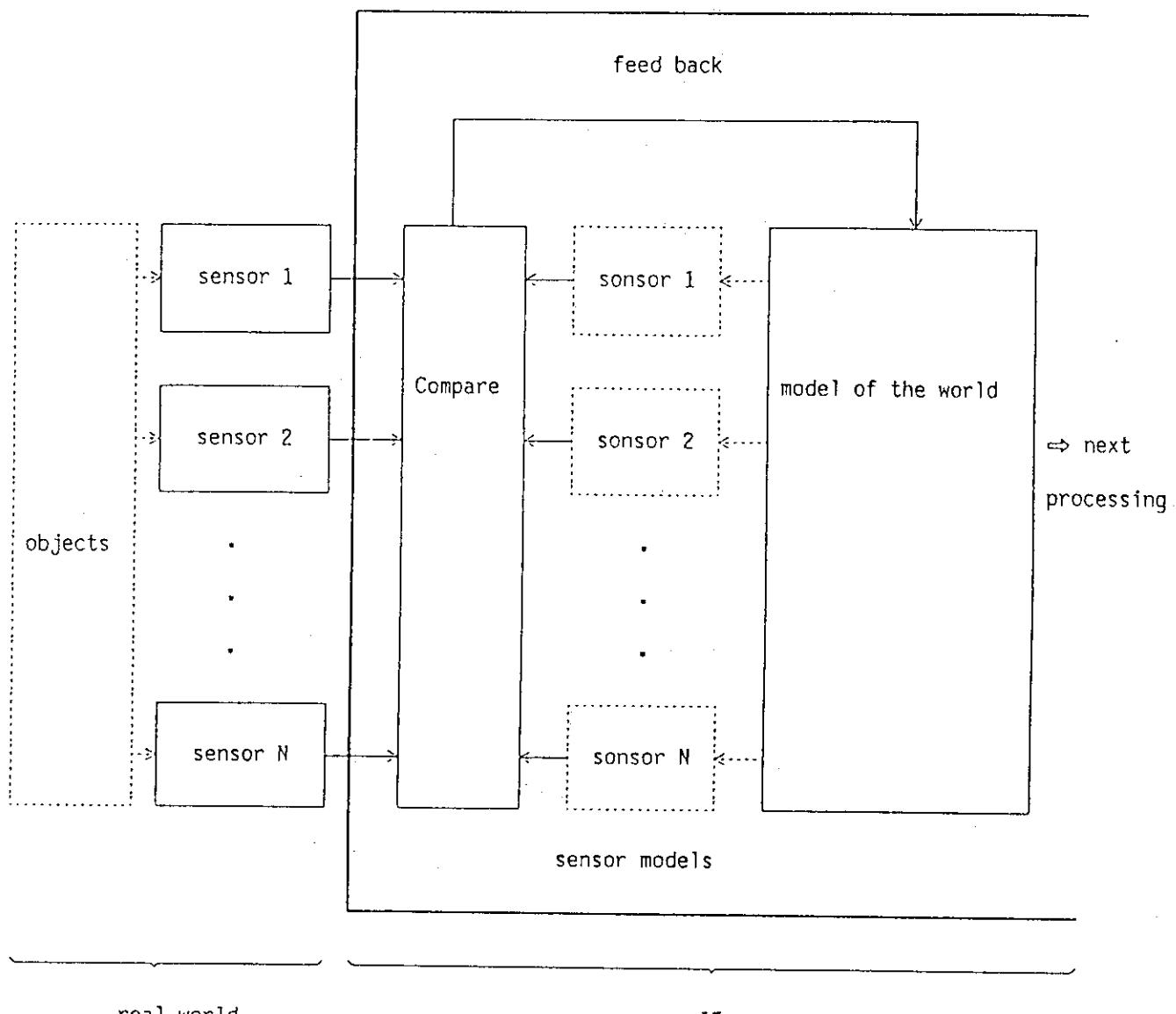


図2: 物理空間法の考え方を示す概念図。

黒枠で囲われた右側が行動決定装置部分である。図1の、センサー群からの情報を行動決定装置が受けとて最初の処理を施す部分にあたる。外界のモデル（model of the world）をセンサーのモデル（sensor models）で見たときの出力信号が実際のセンサーからの信号と一致するように外界のモデルを調整しようという考え方が物理空間法である。

multi-sensor fusion system following PSM idea



方である。先の例で続けると、細長い白い物がカメラから 20cm のところにあるので、もう少し退いて見たら鉛筆だったとか、カメラから 5m の所にあり床から天井に抜けているので配管に違いない、とかいった具合である。

つまり、大雑把にいって、外界の様子をセンサー融合によって再現するステップとその融合結果をつかって次の行動を決定するステップとにわけてロボットを実現しようという考え方が物理空間法によるセンサー融合の特徴である。もちろん、2番目のステップでの行動決定の際に、センサー融合部分のモデルの変更など1番目のステップへのフィードバックがかかるなどを否定するわけではない。

このような外界の状況をロボット内部のモデルに再現するという意味でのセンサー融合を、物理空間法はどのようにして実現するのか説明しよう。物理空間法では、まず、センサー群のセンスしている外界を物理学の客観的記述を使って実在論的に捉え、外界のモデルを物理法則に従って用意する。このモデルを世界モデルと呼ぶことにする。世界モデルは数多くの自由度を含んでいて、各自由度の状態を様々に変えることにより、ロボットの見るであろう世界の状態をほとんど全て記述できるものとしておく。例えば、カメラの例であれば3次元空間での色の分布などである。3次元空間を格子で近似し、各格子点に色と輝度の自由度を置いておけばよからう。放射線であれば線源の種類・位置と放射線の吸収物の分布などである。さらに、センサーの特性も物理的に記述してモデル化する。これをセンサーモデルと呼ぶ。カメラならば例えば幾何光学的なモデル化により、与えられた光線分布をピクセルからの信号に変換する物としてその動作を記述する。さて、センサーモデルに世界モデルをセンスさせておく。そして世界モデルの自由度を、センサーモデルからの信号が実際の信号を近似するように調製する。センサー（現実の世界をみている）とセンサーモデル（世界モデルを見ている）とが同じ信号を出すような世界モデルの状態を、センサー情報を融合して得られた結果と見なすのである。

次節で物理空間法の考え方に基づいた両眼（多眼）立体視問題を解くモデルの一例を紹介する。

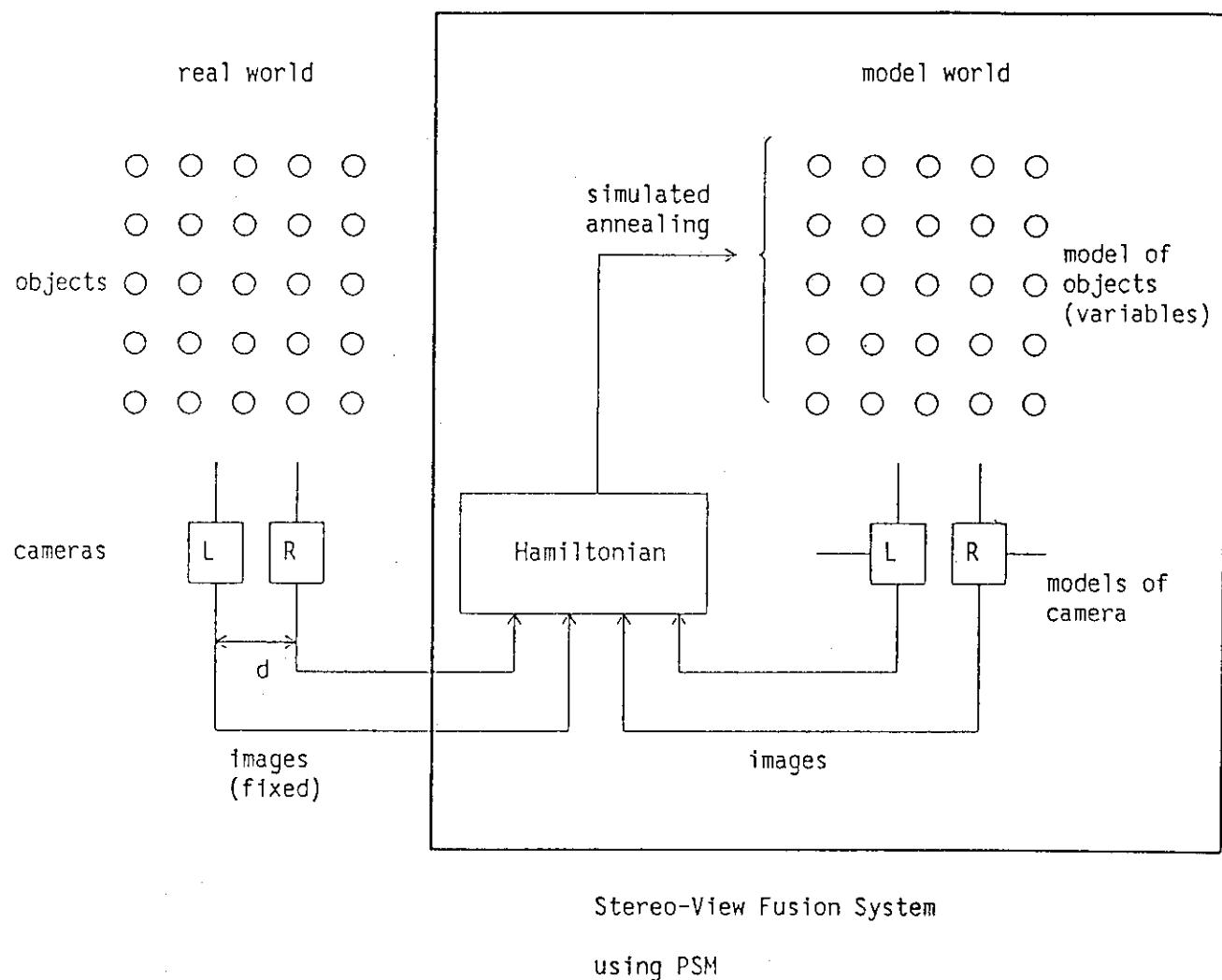
3 物理空間法の両眼立体視問題への応用

前節で多少触れたが、物理空間法の最初の応用例は両眼（多眼）立体視問題である。これは異なる位置からの映像を使い、同じ（と思われる）物体の視差から見ているものまでの距離を測定する問題である。両眼立体視法は物体までの距離を測定する方法として考えられているさまざまな手法のひとつである[3]。2ヶ所以上からの画像あったときに、同じ物がどこに写っているかを同定できれば、各画像での視差およびカメラの位置・特性からその物体までの距離を知ることができる。この方法によって距離を知ろうとする方法が両眼（多眼）立体視法である。この方法の長所は、純粹に視覚的に距離を測定できることである。しかしながら、どの像がどの像と対応しているのかを知ることは一般的には困難な問題であり、汎用的に使える立体視問題の解決法はまだない。この問題は多センサー融合の重要な例である。本節では、物理空間法の考え方によって多眼立体視を解く一つの方法を紹介しよう。

3.1 モデル

問題を簡単化して2次元の世界を考え、この世界を1次元のスクリーンをもつカメラで見るとする。モデルの詳細は以下に説明する。また、全体の模式図を図3に示す。

図 3: 物理空間法の考え方に基づいて複数のカメラからの映像を融合して距離情報を再現するモデル。



ロボットの作業している世界

- 物体は大きさ 5×5 の 2 次元正方格子上にのみ並んでいて、物体の広がりは無視できるものとする。
- この世界の座標の原点を正方格子の左下の格子点を $(0,0)$ とし、格子の軸に平行に x, y 座標をとり、格子間隔を長さの単位とする。すると物体の存在しうる格子点は

$$\Lambda = \{(i, j); i, j = 0, 1, 2, 3, 4\}$$

である。

- 物体の色は RGB 各 1 ビットで表現されるとする。
- 物体表面は他の物体からの光を反射しないとする。
- 物体・格子間隔は光の波長に比べて十分大きいとし、光線は直線的に進むとする。

2 台のカメラの動作

- カメラは 2 台あり、 $(1.5, -1)$ (カメラ L), および $(2.5, -1)$ (カメラ R) にあって、ともに y 方向正の向きを向いているとする。
- カメラの向いている方向を中心に左右 90 度、全体で 180 度 (半平面) からの光を受けるとする。
- ピクセルの数は 100 とし、各ピクセルは視野を等角に見るものとする。すなわち、各ピクセルは 1.8 度角を見る。
- 各ピクセルは色 RGB を各 1 ビットで感じて出力するものとする。
- 各ピクセルは自分の見ている範囲内で最も近い物体の色を出力するものとする。
- カメラ L および R のピクセルからの出力を $\tau_i^{l,D}$ とする。この τ は世界にある物体の配置によって決まる 0 または 1 の定数である。ただし、 i はピクセルの番号を表わし、 $1, 2, 3, \dots, 100$ とする。 l は色を表わし、R, G, B をとる。 D は左右どちらのカメラかを表わし、L, R をとする。

ロボットが受け取る信号は $\tau_i^{l,D}$ である。いま問題となることは、この信号からいかにして距離の情報を含んだ 2 次元空間での物体の配置を再現するかということである。

以下はこの問題を物理空間法によって解く最も単純な方法である。まず、外界のモデルとして、実際の外界と同じものを用意する。

世界のモデル

- 大きさ 5×5 の 2 次元正方格子の格子点 (i, j) 上にその場所の色を表わす変数 $\sigma_{(i,j)}^l$ ($l = R, G, B$) が並んでいるとする (座標は外界の格子と同じ様にとる)。
- 変数 $\sigma_{(i,j)}^l$ が与えられたら、その格子点にその変数の値の色の物質があるとみなして、上述の世界と同じ様に光が分布するとする。ただし、 $\sigma_{(i,j)}^R = 0, \sigma_{(i,j)}^G = 0, \sigma_{(i,j)}^B = 0$ のときにはその格子点 (i, j) には物体がないとみなす。

2台のカメラのモデル

- カメラのモデルは外界のモデルに対して上述のカメラと同様に動作するものとする。
- ある外界のモデルの状態 $\{\sigma_{(i,j)}^l\}$ をカメラのモデルで見たときのピクセルの出力を

$$\{\mu_i^{l,D}; i = 1, 2, 3, \dots, 100; l = R, G, B; D = L, R\}$$

とする。すなわち、これらは変数 $\sigma_{(i,j)}^l$ の関数である。

世界のモデルの状態を決定する方法

カメラのモデルで世界のモデルをみたときに得られる映像 $\mu_i^{l,D}$ が実際のカメラからの映像 $\tau_i^{l,D}$ と一致するように、状態 $\sigma_{(i,j)}^l$ を決めるには色々な方法がありうる。動的計画法、ニューロモデルなどさまざまな方法で決定することができるであろう。ここでは最も簡単に、シミュレーテッドアニーリングを使うことにする。

コスト関数（エネルギー関数） H を、最も簡単な形として実際の映像とモデルの映像との差の2乗和の形に取ることにする。すなわち、

$$H(\{\sigma_{(i,j)}^l\}) = \sum_{i,l,D} (\mu_i^{l,D} - \tau_i^{l,D})^2$$

とする。ここで、 $\mu_i^{l,D}$ は2台のカメラのモデルを通して $\sigma_{(i,j)}^l$ の関数である。世界モデルの状態全体の空間 $\Omega = \{\{\sigma_{(i,j)}^l\}; \sigma_{(i,j)}^l = 0, 1\}$ 上で逆温度 β のカノニカル分布

$$P(\omega) = \frac{\exp(-\beta H(\omega))}{\sum_{\omega \in \Omega} \exp(-\beta H(\omega))}$$

を考える。そして、各温度でこの分布関数に収束するようなモンテカルロシミュレーションを行い、だんだん温度を下げていくと十分低い温度（すなわち、十分大きい β ）であらわれる状態のなかにエネルギーが最小のものがでているであろうと期待される。そこで、十分低温であらわれる状態のうち、最も小さいエネルギーを与える状態 $\{\sigma_{(i,j)}^l\}$ を2台のカメラの情報を融合して得られたものとして採用することにする。

このシミュレーションのモンテカルロダイナミクスとしては、変数 $\sigma_{(i,j)}^l$ を熱浴法で逐次更新していく方法を使うこととする[4]。

3.2 融合例

このモデルで図4に示すような物体の配置を与えた結果を図5に示す。シミュレーテッドアニーリングでは、全ての $\sigma_{(i,j)}^D = 0$ の状態からはじめて、温度を $\beta = 0.2, 0.4, 0.8, 1.6, 3.2$ と下げてゆき、各温度でモンテカルロスイープを1000回行った。そして、 $\beta = 3.2$ のステップに現れた配位のうち、エネルギーが最小のものを融合結果として採用した。

この図からわかるように、確かに2つの映像から距離が復元されている。すなわち、簡単な配置の場合ではあるが、このモデルで両眼立体視が正しくできることが示された。

3.3 この方法の特徴

両眼立体視のこのモデルを任意の位置にある任意の数のカメラに拡張することは容易である。また、色の分解能をもっと現実的に拡張することや、移動眼に拡張することも容易である。

現在、この方法を実用的な両眼立体視をおこなうものに拡張することを試みている。

図4: 世界の物体の配置として与えたパターン。物体の色が $\sigma_{(i,j)}^R$, $\sigma_{(i,j)}^G$, $\sigma_{(i,j)}^B$ であることを、整数 $\sigma_{(i,j)}^R + 2 * \sigma_{(i,j)}^G + 4 * \sigma_{(i,j)}^B$ であらわした。

0	4	4	4	0
1	1	0	1	0
0	0	0	4	4
0	4	0	0	0
0	0	0	2	0

図5: 図5の世界にたいして映像を融合して得られた世界モデルの配置。図5と同様の方法で色を0から7までの整数で示した。(b)で*印の場所は0から7まで任意の値が入ったものがえられた。これはこの図で破線で囲んだ2つの格子点が今仮定したカメラでは区別が出来ないことから生じた現象である。

0	4	4	4	0	0	4	*	4	0
1	1	0	1	0	1	1	4	1	0
0	0	0	4	4	0	0	0	4	4
0	4	0	0	0	0	4	0	0	0
0	0	0	2	0	0	0	0	2	0

(a)

(b)

4 おわりに

1つ1つのセンサーからの信号だけでは得られない対象についての情報を、複数のセンサーからの信号を融合することにより得ることは知能化システムでは重要である。しかしながら、なんの知識も仮定せずに融合させるには組合論的な計算時間が必要となり、現実的でないことが多い。一方、あらかじめ強力な先入観をもって融合すれば計算の手間は省けるが結果が曖昧な、あるいは必ずしも正しくないものとなりがちである。本文で論じた能動型ロボットの場合、外界の様子は物理的なモデルによって十分良く記述されるであろう。このため、この物理的な世界像を知識として融合を行えば、正しい情報が得られるであろう、として提唱されたのが物理空間法である。前節にあげた両眼立体視問題の例で、図5 (b) で視差の限界が自然と浮き彫りとなったことからもわかるように、物理空間法によるセンサー融合は融合に使われたセンサー情報の限界をも明らかとする。必要とする情報が検出された限界のさきにありそうであれば、その情報を確実に得られる場所にロボット自ら移動する制御を行えば良い。すなわち、物理空間法はロボットが作業環境について情報を得る際に必要となるセンサー融合を実現する有効な考え方といえるであろう。この考え方は、さまざまな種類のセンサーを融合する一般的な方法論であり、今後、センサー融合を研究する際に有用なものであろうと考えられる。

謝辞

この研究は日本原子力研究所情報システムセンターの FACOM M780 を使ってなされた。計算機の運用に携わっておられる方々に感謝する。

参考文献

- [1] N. Ito, in preparation.
- [2] K. Asai, J. Uenaka, S. Kambayashi, K. Higuchi, E. Kume, T. Otani, M. Fujii and M. Fujisaki, Proc. of the First Intern. Conf. on Supercomputing in Nuclear Appl. (SNA '90) (NEDAC, 1990) p. 611.
- [3] U. R. Dhond and J. K. Aggarwal, IEEE Trans, Systems, Man and Cybernetics 19 No. 6 (1989) p. 1489.
- [4] N. Ito, *Monte Carlo Study of the Ising Model* (Doctor Thesis, University of Tokyo, 1990).

7. 固体物性の計算物理

宮下 精二（京大・教養）

他の分野同様、物性の分野でも計算機は研究遂行の上で重要な手段となってきた。この分野の主な研究対象は、相転移をはじめとする強い相互作用をもつ多体系の熱力学的性質と強く相互作用している固体内の電子系の量子力学的性質（主に基底状態）の解明である。計算機がこれらの研究にどの様に用いられてきているかを概観し、それぞれの特徴や問題点を議論する。特に、磁性体モデルの研究を例にして、古典系および量子系でのモンテカルロ法を紹介する。また、電子状態から固体物性を第一原理的に解明する方法も計算機の進歩と共に非常に進んで来ているがそれらの成果についても触れたい。

Computational Approach to Condensed Matter Physics

Seiji Miyashita (Kyoto Univ.)

In the condensed matter physics, the computational method has become a very powerful tool as well as in other fields. We are mainly interested in the thermodynamic properties of strongly interacting materials. In particular, the nature of phase transition is one of our main interests. Furthermore, the nature of the ground state of strongly correlated quantum systems is also our main theme. I will give a brief review how the computer has been made use in these fields and discuss their characteristics and problems. I take a model of magnetic material as an example and introduce classical and quantum Monte Carlo methods. Furthermore, I will refer to recent developments of the first-principle calculations for material properties based on the electronic state.

Keywords (Monte Carlo method, Quantum Monte Carlo method, Molecular dynamics simulation)

1. はじめに

物性物理学の対象は多体系の熱力学的性質であるが、大きく分けて平衡系の性質と（非平衡な）トランジエントな状態の性質にわかれる。平衡系の性質に関してはカノニカル分布を実現しそこでの物理量を測定するモンテカルロ法やミクロカノニカル分布を用いる分子動力学法がある。前者はある種の確率過程の定常状態としてカノニカル分布を実現し、後者は通常の力学方程式に従い系を時間発展させエルゴード仮説により熱力学的性質を得ようとするものである。

2. モンテカルロ法

モンテカルロ法の実現は各分野によっていろいろ特徴があるので統計力学におけるそれを少し説明しよう。⁽¹⁾ 系の状態を $\{S_i\}_{i=1,N}$ 、 N は位相空間の大きさ（数）とする。ある確率過程を用意し、そこで S_i が S_j に遷移する確率を L_{ij} としよう。系の状態の分布関数を $P(i)_{i=1,N}$ とすると上の過程は

$$P' = L_i P$$

と表せる。今ここで L は 1 回のモンテカルロの操作による状態更新の演算子である。適当にモンテカルロの操作を繰り返し、 L_p, \dots, L_1 、それをまとめて L としてこれを 1 モンテカルロステップ（MCS）と呼ぶ。ここで L の ji 成分は 1 モンテカルロステップの後の $S_i \rightarrow S_j$ の遷移確率である。当然それらは正で、また全ての状態間の遷移が可能なので L は正値行列である。（もし、何らかの理由で全ての状態間の遷移ができないときは正しい結果が得られない。これをモンテカルロ法のエルゴード性という。）正値行列の最大行列は正で非縮退であることが知られており（ペロン・フロベニウスの定理）、そのためこの確率過程は一意的な定常状態を持つ。この定常状態がカノニカル分布 P_{eq} になるようにするのが詳細釣合の条件

$$L_{ji} P_{eq}(i) = L_{ij} P_{eq}(j)$$

である。 L がこの条件を満たすと P_{eq} が定常状態になっていることがわかる。以上がモンテカルロ法の原理である。このエルゴード性と詳細釣合の条件を満たす多くの実現方法があるが主なものに次の 2 つがある。

2.1.1 メトロポリス法

遷移可能な状態から 1 つの状態を選び、その状態 S_i ととの状態 S_j のエネルギーを比較し

$$\begin{aligned} E(i) < E(j) : L_{ij} &= 1 \\ E(i) \geq E(j) : L_{ij} &= \exp((E(j) - E(i))/k_B T) \end{aligned}$$

ここで T は温度、 k_B はボルツマン定数である。

2.1.2 熱浴法

$$L_{ij} = \exp(-E(j)/k_B T) / \sum_k \exp(-E(k)/k_B T)$$

ここで k は遷移可能な全ての状態について和をとる。

2.2 モンテカルロ法の応用1：イジングスピニ模型

モンテカルロ法の応用として強磁性イジング模型の解析例を紹介しよう。

$$\mathcal{H} = -J \sum_{\langle ij \rangle} S_i S_j, \quad S_i = \pm 1$$

和は全ての最近接対についてとる。図1にこの系の秩序変数である磁化 ($M = \sum_i S_i$) の帶磁率 χ

$$\chi = (\langle M^2 \rangle - \langle M \rangle^2) / k_B T$$

の温度・サイズ依存性を示す。ここで $\langle \dots \rangle$ はカノニカル分布での平均を意味する。このモデルは有限温度に相転移を持つことが知られているが、それに伴い帶磁率が発散する。しかし有限の系ではその発散は抑えられ図に見られるようなサイズ依存性となって現れる。このようにゆらぎが大きな系のシミュレーションは困難であるがこのサイズ依存性から逆に熱力学的極限での発散を知る方法が開発されている。それは有限サイズスケーリング法と呼ばれるもので自由エネルギーの特異部を系の相関距離 ξ を用いて

$$f_s(T, H) = \xi^{-d} f(\xi \epsilon^\nu, H \xi^{\frac{1}{2}(d+2-\eta)}, \xi/L)$$

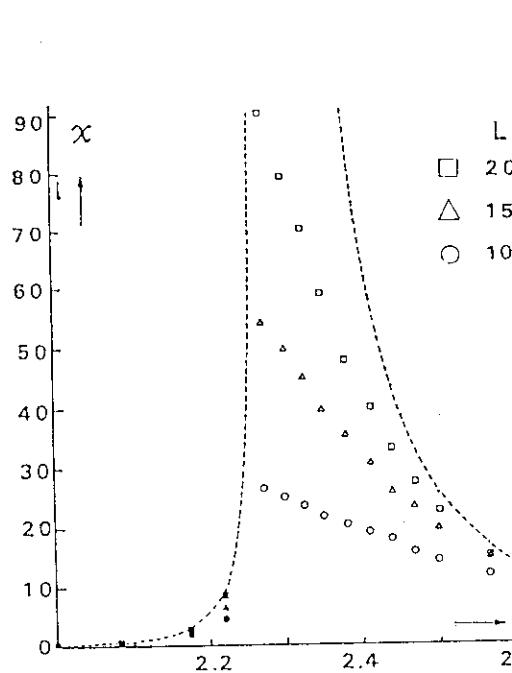


Fig. 1

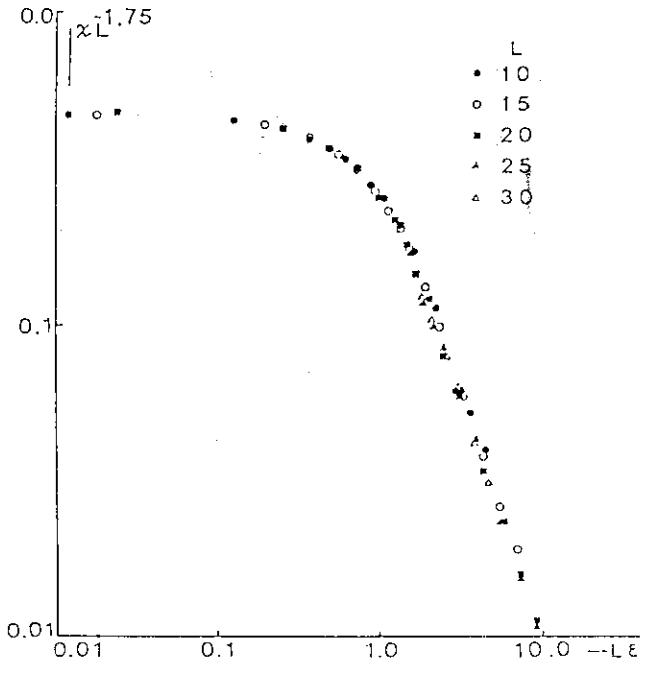


Fig. 2

Fig. 1 Size- and temperature-dependence of χ of the two-dimensional Ising model.
Dotted lines are due to expansion methods. (From ref. (1))

Fig. 2 Scaling plot of data shown in Fig. 1 (From S. Miyashita and H. Takano,
Prog. Theor. Phys. 73 (1985) 1122)

と表すものである。ここでは d 次元、また L は系の 1 辺の長さである。ここでたとえば、帶磁率は F_s を H で 2 回微分することで $\chi \sim L^{2-\eta} \phi(L\epsilon^\nu)$ (ここで ϕ は F から作られるあるスケーリング関数) であることがわかるが、そのため $(\chi/L^{2-\eta}, L\epsilon^\nu)$ の座標でデータをプロットすれば全ての点はスケーリング関数 $\phi(x)$ の上に来る。図 2 にその様子を示す。また低温側での自発磁化の大きさ m_s も

$$\langle M^4 \rangle / L^{2d} = m_s^2 + k_B T \chi / L^d$$

の関係から求められこの温度変化から臨界温度 T_c が相当正確に評価されている。この臨界温度は他にモンテカルロ繰り込み群の方法やスケーリング法でも詳しく調べられており全てよい一致を与えている。図 3 に 3 次元イジング模型の m_s の温度依存性を示す。⁽²⁾ 特にイジングスピニ系では、高速化のため計算機の各ビットを利用したマルチスピニコーディングと呼ばれるテクニックなどが用いられ 100^3 のオーダーの系で実用的なデータが採られている。⁽³⁾

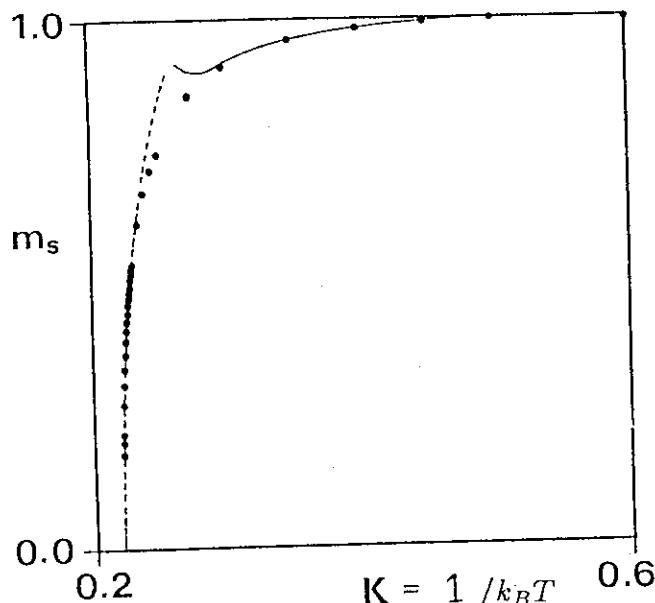


Fig. 3

Fig. 3 Temperature dependence of m_s of the three-dimensional Ising model. (From ref. (2))

2.3 連続スピン系でのモンテカルロ法

連続スピン系でも本質的には上で述べたものと同じであるが連続変数の中から状態を選び出す仕方に工夫が必要である。詳しくは文献(4)を参照されたい。ここでは層状三角格子反強磁性イジング的異方性をもつハイセンベルグ模型

$$\mathcal{H} = \sum_{\langle ij \rangle} J(S_i^x S_j^x + S_i^y S_j^y + S_i^z S_j^z) - D \sum_i (S_i^z)^2, \quad (*)$$

のモンテカルロ法の応用を紹介する。図 4 に系の非熱を示す。ここでの 2 つの鋭いピークはこの系が 2 回の相転移を持つことを示している。それらはそれぞれ z -成分、 xy -成分の秩序の形成に関

わるものであることが図5によってわかる。このデータもうまいりやレの値をとることで上述の有限サイズプロットで図6のようにまとめられる。これらの値は対応する物質の実験とよく一致している。

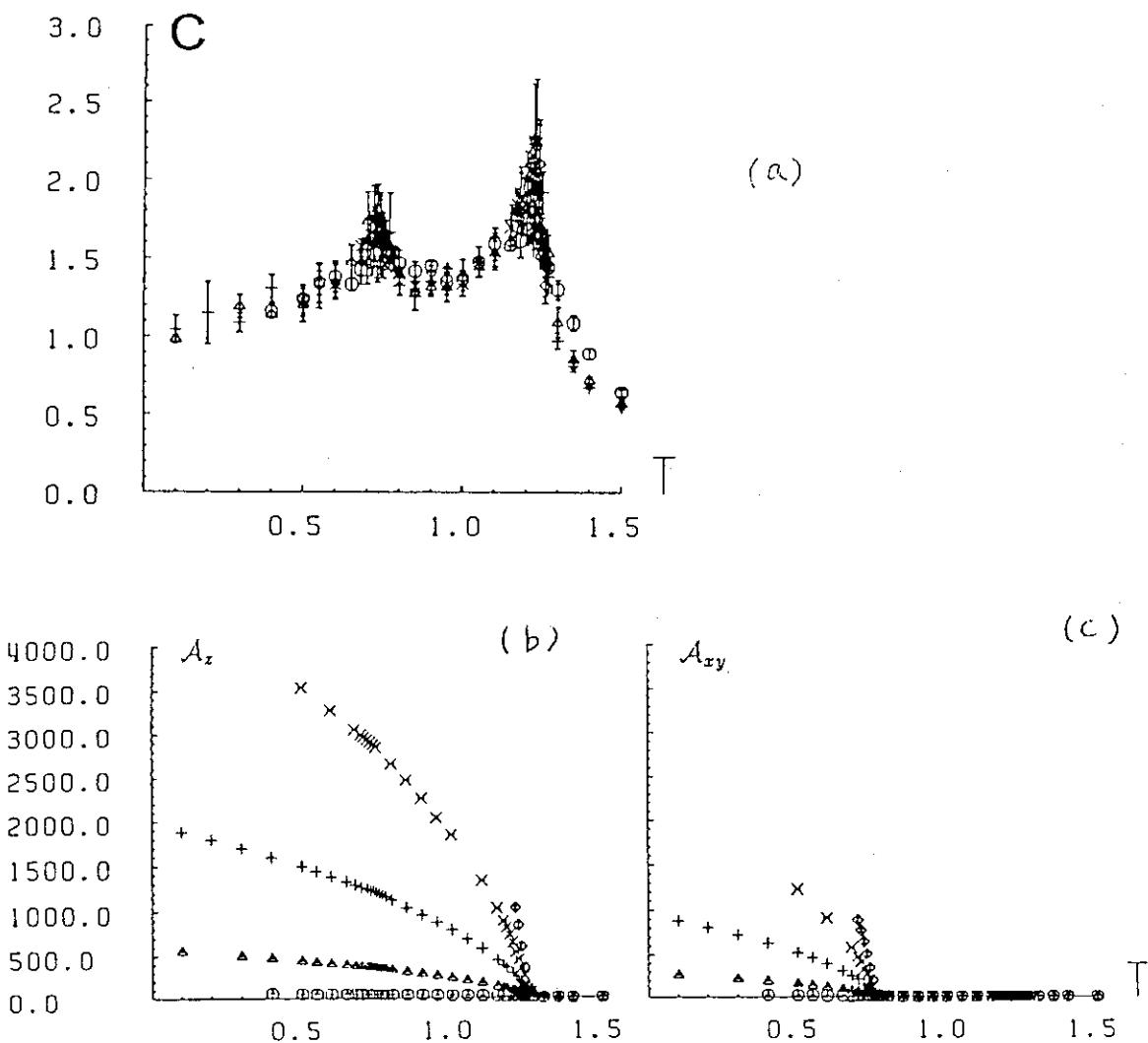


Fig. 4 Temprature dependence of the specific heat of the model (*). Fig. 5 Temprature dependence of the staggered susceptibility of the model (*).

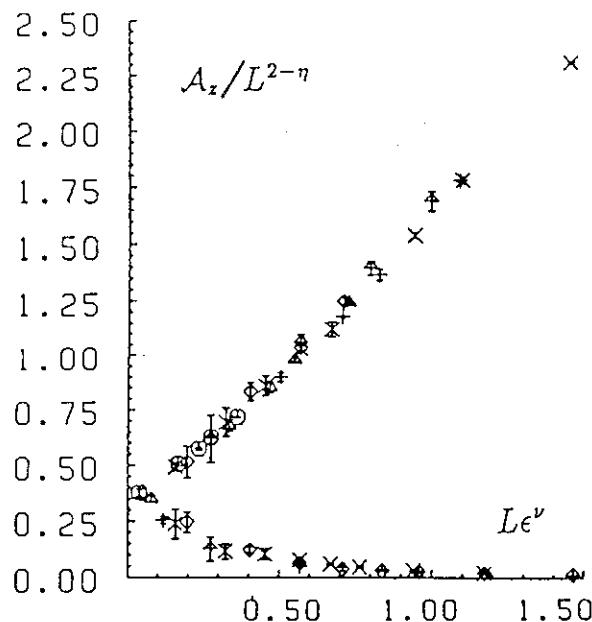


Fig. 6 Scaling plot of the data in Fig. 3(b)

2.4 量子系のモンテカルロ法

次に量子系のモンテカルロ法について述べる。ここではカノニカル分布を与えるボルツマン因子が古典的な場合と異なり評価が非常に困難になる。つまり $P(i)$ は

$$P(i) = \langle i | \exp(-\beta \mathcal{H}) | i \rangle \neq \exp(-\langle i | \beta \mathcal{H} | i \rangle)$$

であるため \mathcal{H} のすべての固有値、固有ベクトルがわからないとそれは求められない。しかしこの点は鈴木によって導入された Suzuki-Trotta 分割

$$\exp(\sum_i \mathcal{H}_i) = \lim_{n \rightarrow \infty} (\prod_i \exp(\mathcal{H}_i/n))^n$$

を用いることである種の経路積分として $P(i)$ を求められるようになった。⁽⁵⁾ この方法によりスピン $1/2$ の量子スピン系や電子系のタイトバインディング模型などが研究されている。(詳しくは文献(5,6)参照) 図 7 にこの方法で得られた 2 次元反強磁性 X X Z 模型 (スピン $\frac{1}{2}$) ,

$$\mathcal{H} = - \sum_{\langle ij \rangle} (J_{xy}(S_i^x S_j^x + S_i^y S_j^y) + J_z S_i^z S_j^z) \quad (**)$$

の基底状態での長距離秩序の大きさの異方性依存性を示す。⁽⁷⁾ ここでは系の秩序変数である反強磁性磁化 (副格子 A, B) $N_k = \sum_{i \in A} S_i^k - \sum_{i \in B} S_i^k$, $k = x, y$ or z の対称性はハミルトニアンの対称性と共に鋭く変化するが、量子ゆらぎのためその値が大きく減少していることがわかる。対応する古典的な値は 0.25 である。現在、コネクションマシンなどを用いての 128×128 の系で 128 個の ST 分割程度まで計算可能であるが、十分低温の性質を知るためにより大きな分割が必要となってきた。

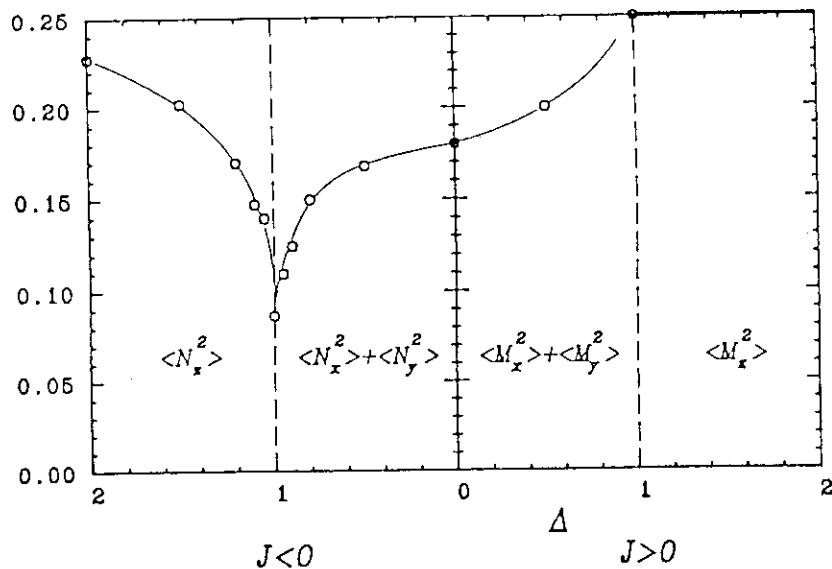


Fig. 7 Anisotropic-dependence of the orderparameter of the model (**). Here Δ is the ratio J_z/J_{xy} . (From ref. (7))

3. 分子動力学

分子動力学では系を通常の力学方程式に従って運動させミクロカノニカル分布を作るものであるがそこでの温度は運動エネルギーの平均で与えられる。しかしこの場合欲した温度をうまく実現することは比較的困難であり、また何らかの理由で温度が変化してしまうことなどがある。そこで系の温度変化を容易にするため、系にもう1つ自由度を付け加えそれをある種の熱浴として用いる方法が能勢によって導入された。また同様な手法で圧力も一定にする方法も考案された。(文献(8)参照) この方法により物質の結晶化やガラス転移の様子が詳しく述べられている。しかし、分子動力学法での一番の問題は分子間の力としてどの様なものを用いるかである。希ガスなどの場合は、レナード・ジョンズボテンシャルが用いられているが、より複雑な系では、これまでの多くは経験的な形、つまりシミュレーションが最もよく実験を再現する形にとてきている。しかしこれはある意味では本末転倒の感があり、また広いパラメタの範囲で有効な形を見いだすのは難しかった。そこで最近の計算機の発達に伴い出てきたのがいわゆるAb initioな分子動力学である。ここでは力を非経験的な計算で求めそれに基づいて分子(あるいは原子、イオン)を運動させるものである。そのためにはまず原子の位置(R_i)を決めてその時の電子状態のエネルギーを量子力学的に計算しそれによって(R_i)に対する断熱ボテンシャルを求める。各(R_i)に関してこのようにエネルギーが求められればよいが今のところそれは困難である。そこでいろいろな方法が考えられるがここでは2つの例を紹介する。

3.1 対ボテンシャル近似

常行らは⁽⁹⁾、シリカ系の結晶変態を捕らえるために分子動力学法を用いたが、それまでシリカの多形を再現するボテンシャルは知られていなかった。そこで彼らは非経験的な方法で断熱ボテン

シャルを計算しそれを最もよく再現する2体ポテンシャルを決め、それによってシミュレーションを行ったところ非常に多く結晶構造を再現した。図8に彼らの得た結晶構造を示す。これら結晶構造のみならず圧縮率なども非常によく再現され、ここで2体ポテンシャルが非常によく多体エネルギーを表していることがわかった。しかし、どの様な場合にこのような2体近似がうまくいくのかなど一般性はよくわかっておらず今後の問題である。

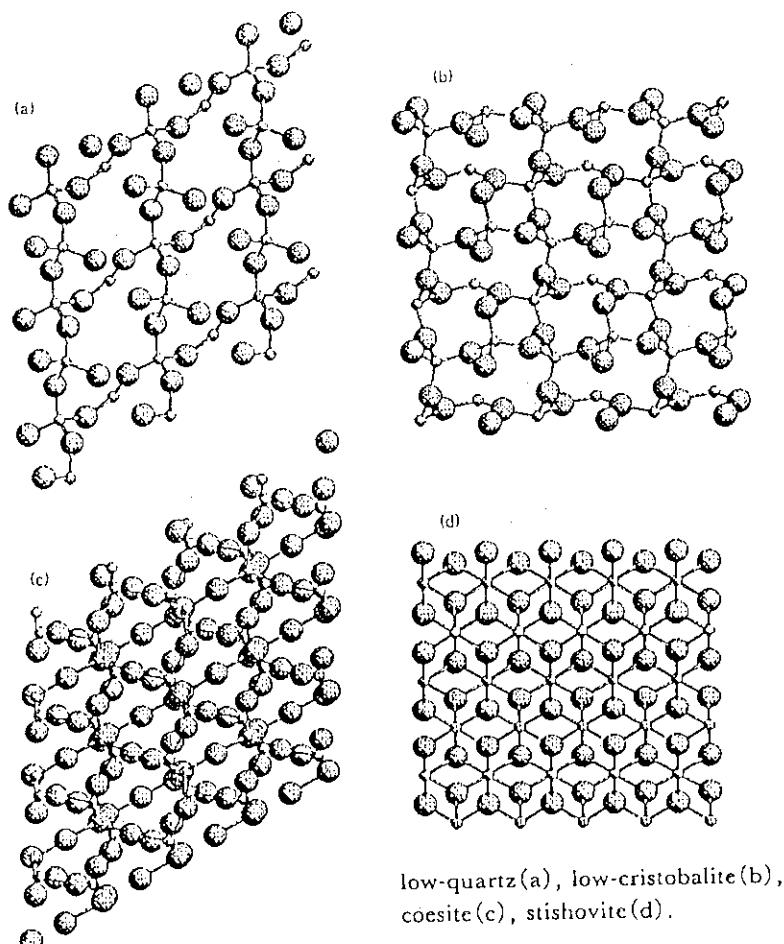


Fig. 8 Crystal structures for silica. (From ref. (9))

3.2 Dynamical Simulated Annealing の方法

上で各 $\{R_i\}$ に対する電子エネルギーの計算は困難であると述べたがこの計算も分子動力学的 $\{R_i\}$ の計算と平行して行う方法が Car と Parrinello によって導入されている。⁽¹⁰⁾ 系の動的性質を調べる上では非常に有力な方法と期待されている。図9にこの方法で求められた液体シリコンの電子密度分布を示す。

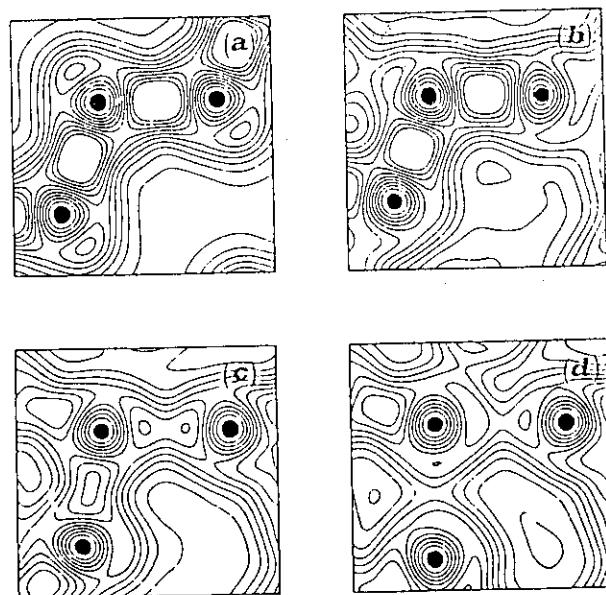


Fig. 9 Contour plots of the valence electronic density. (From ref. (10))

3.3 完全な第一原理計算物理の方法への挑戦

上の2つの方法では電子状態の計算にハートリホック近似や密度汎関数法での局所密度近似などの近似方法がとられているが、電子相関が強い場合などは不十分である。そこで次世代の *A b - i n i t i o* はそれさえも厳密に扱う必要があるがその一つの候補として、2.4で述べた量子モンテカルロ法を電子系に応用することが考えられる。電子系では負符号問題と呼ばれる困難があるが補助場の導入等によって徐々に扱える範囲が広がってきており、これらの方法の融合によってより進んだ方法が近い将来開発されることが期待される。

参考文献

- (1) 齊藤幸夫、宮下精二: 日本物理学会誌 第40巻 第11号 p843.
- (2) 伊藤伸泰: 私信
- (3) N. Ito and Y. Kanada: IEEE Computer Society Press, Los Alamos, (1990) p753.
- (4) 宮下精二: 固体物理(計算物理特集号) Vol. 24 No. 3 (1989) 144.
- (5) M. Suzuki: in *Quantum Monte Carlo Methods in Equilibrium and Non-equilibrium Systems*, ed. by M. Suzuki (Springer, Berlin 1987) p2.
- (6) 宮下精二: 固体物理 Vol. 23 No. 4 (1988) 231, No. 5 (1988) 315,
Vol. 24 No. 2 (1989) 81.
今田正俊、宮下精二: 固体物理 Vol. 24 No. 10 (1989) 746.

- (7) S. Miyashita, in *Quantum Simulations of Condensed Matter Phenomena* ed.
by J. D. Doll and J. E. Gubernatis (World Scientific 1990) p228.
- (8) 能勢修一: 固体物理(計算物理特集号) Vol. 24 No. 3 (1989) 232.
- (9) 塚田 捷: 固体物理(計算物理特集号) Vol. 24 No. 3 (1989) 259.
松井義人: 固体物理 Vol. 23 No. 12 (1988) 889.
- S. Tsuneyuki, H. Aoki, M. Tsukada and Y. Matsui: Phys. Rev. Lett. 64
(1990) 776.
- (10) R. Car and M. Parrinello: Phys. Rev. Lett. 55 (1985) 2471.
I. Stich, R. Car and M. Parrinello: Phys. Rev. Lett. 63 (1989) 2240.

8. 3次元輸送コードの開発の現状とベンチマーク計算

竹田 敏一, 山崎 正俊, 池田 秀晃, 錦織 賢夫 (大阪大学)

要旨：大阪大学から提案された NEACRP 3次元中性子輸送ベンチマークテストの結果、および、開発中の 3次元中性子輸送コードの現状について報告する。

3次元中性子輸送ベンチマークテストは、1988年に竹田、他が NEACRP に4つの問題を提案し、20機関より 22の結果が寄せられたもので、計算手法はモンテカルロ、 S_n 、 P_n 、合成法、ノード法が用いられている。実効倍率、制御棒反応度価値、領域平均中性子束について、結果の整理を行った。その結果、(1)XYZ体系では S_n (n=8) 法はモンテカルロ法と良く一致し、場合によっては良い結果を与える、(2) P_n 法では空間メッシュ効果が大きい、(3)六角-Z体系では S_n 法は問題があり、より精度の高い輸送計算法の開発が必要である、等が判明した。

3次元中性子輸送コードの開発に関しては、ベンチマークテストで明らかになった六角-Z 体系での問題を解決するために、ダイヤモンド差分の新しい方法、六角体系での修正粗メッシュ法、およびノード法の適用を試みている。

Development of a 3D Neutron Transport Code and Benchmark Tests

T. Takeda, M. Yamasaki, H. Ikeda, and T. Nishigori
(Dept. of Nuclear Engineering, Osaka Univ.)

Summary: Results are reported of NEACRP "3D Neutron Transport Benchmarks" proposed from Osaka University, and of recent progress in the development of a 3D neutron transport code. Takeda et al. proposed four problems to NEACRP as 3D neutron transport benchmarks, and 22 results from 20 organizations were submitted. A variety of methods have been used, such as the Monte Carlo, S_n , P_n , synthetic, and nodal methods. The results for k-eff, control-rod worths, and region-averaged fluxes are summarized with the conclusions that (1) in XYZ geometry the S_n method with n=8 shows a good agreement with the Monte-Carlo method, and gives even better results in some cases, (2) the P_n method has significant spatial mesh effects, and (3) the S_n method is not satisfactory in hexagonal-Z geometry, and improvements in accuracy are desirable. Improvement of a 3D neutron transport code is in progress to resolve the problem in the hexagonal-Z geometry by considering new diamond difference schemes and an improved coarse-mesh method, and also by applying the nodal method.

1. はじめに

「三次元中性子輸送ベンチマーク」問題が 1988 年に大阪大学より NEACRP に提案された。その目的は、多くの三次元輸送計算の精度を比較検討し、三次元体系における中性子輸送計算のベンチマークを確立する事である。ベンチマーク問題は、XYZ 体系の小型軽水炉（モデル 1）、小型FBR（モデル 2）、軸非均質大型FBR（モデル 3）、および、六角-Z 体系の小型FBR（モデル 4）の 4 炉心よりなり、モデル 1 は 2 群、他は 4 群の断面積が与えられている。10カ国、20機関より、22件の参加があった。Table 1 に参加者のリストを計算コード名、計算法とともに示す。結果を検討するために、1990年10月22-23日にサクレーの NEA データバンクにおいて、ワーキンググループ会合を持った。会合には Table 2 に示すように 12 機関から 15 名が参加した。会合の主旨説明及び参加者の自己紹介の後、モデル 1 ~ 4 の輸送計算についての発表及び議論がされた。⁽¹⁾、⁽²⁾さらに、今回の会議の結論及び将来の研究と勧告について話し合った。

ベンチマークの結果、六角-Z 体系での Sn 法に問題があることが分ったので、現在、大阪大学において、ダイアモンド差分の新しい方法、六角体形での修正粗メッシュ法、⁽³⁾ノード法等により、3 次元中性子輸送コードの開発が進められている。

2. ベンチマークの結果

Figures 1a-1d にモデル 1 ~ 4 の平面図を示す。モデル 1 ~ 3 はそれぞれ小型軽水炉、小型高速炉、大型軸非均質高速炉の XYZ モデルで、モデル 4 は小型高速炉 K N K - II のヘキサーZ モデルである。計算手法としてはモンテカルロ、Sn、Pn、合成法、ノード法等が用いられた。断面積としてはモデル 1 は 2 群、モデル 2、3、4 は 4 群の群定数が与えられている。計算の収束性についてはモデル 1 ~ 4 とも k_{eff} 、中性子束 ϕ に対し

$$\Delta k_{eff} / k_{eff} < 10^{-5}$$

$$\Delta \phi / \phi < 10^{-4}$$

なる収束条件を採用したが、モデル 1 のような熱中性子炉体系では収束性をもう 1 衡厳しくすべきであると言う意見もあった。会議中に k_{eff} 、制御棒反応度価値について各手法で与えられた結果を比較し、各々の手法の平均値を計算することにした。モデル 1 ~ 4 に対する各ケース（制御棒 in, out）の k_{eff} 及び制御棒反応度価値の各方法での平均値を Table 3 に示す。

2.1 モデル 1

モデル 1 の k_{eff} に対する値は、制御棒引抜きの場合は Sn 法の平均値はモンテカルロ法平均値の 1 σ からわずかに外れているが、制御棒挿入の場合はモンテカルロ法の平均値と Sn ($n = 8$) の平均値はモンテカルロ法の 1 σ 内で一致している。制御棒反応度価値もモンテカルロ法の 1 σ 以内で一致している。Pn 法に関しては制御棒引抜き時（制御棒領域 void）の k_{eff}

がモンテカルロ法、 S_n ($n = 8$) 法に比べて小さいので、制御棒反応度価値も 1.3 % 減少している。Fletcher は P_n の次数 (n) 効果及びメッシュ効果を調べ、共に大きな効果となることを示した。

2.2 モデル 2

小型高速炉のモデル 2 ではモンテカルロ法の標準偏差はモデル 1 に比べ半分以下になっており、高速炉体系では熱中性子を追跡する必要がないため精度良く固有値が計算できることがわかる。 k_{eff} に対するモンテカルロ法と S_n ($n = 8$) 計算との差は 0.0003 程度で小さいが、標準偏差で見ると 2 σ 程度の差がある。 S_n 計算では表には示していないが S_4 計算の k_{eff} の結果は制御棒引抜き、挿入時でそれぞれ 0.9735, 0.9594 であり、 S_8 計算結果と良く一致している。 P_n 法では $n = 7$ と $n = 5$ の結果は良く一致しているが、空間メッシュ効果がある。 P_7 の 5 cm メッシュ、2.5 cm メッシュ、0 メッシュ（外挿）と S_n 法の 5 cm メッシュ、2.5 cm メッシュの k_{eff} の値を比較すると、 P_n 法の空間メッシュ幅による誤差は S_n 法に比べ大きいことが分った。ANL の NTT（ノーダル輸送理論）は k_{eff} を ~ 0.2% 過小評価しているが、制御棒反応度価値についてはモンテカルロ法と良く一致している。

2.3 モデル 3

大型軸非均質高速炉のモデル 3 の k_{eff} 及び制御棒反応度価値の結果を検討する。モンテカルロ法の標準偏差は小型高速炉（モデル 2）と同程度に小さい。また、制御棒挿入時を除くと、 S_8 計算の結果と良く一致している。制御棒挿入時でもその差は 0.0005 であり、 S_n の 1 σ 程度である。 P_7 法の結果はモンテカルロ平均値に比べると 10 % 程度低く、空間メッシュ効果が強く残っていると思われる。NTT 法、輸送等価断面積を用いた CCRR 法による制御棒反応度価値は 1 ~ 2 % 以内で平均値と一致している。このように固有値に関しては各計算手法間の一致は良いが領域平均の各エネルギー群ごとの中性子束を見ると S_8 法の間の一致は良いがモンテカルロ法ではそのバラツキが大きい。特に、 9.61×10^2 eV 以下の第 4 群になると制御棒領域（挿入時）の中性子束は平均値から 1.48% もずれていることが分かる。この点、 S_n 法ではその一致がよい。このように大型炉の中性子束分布を計算するには S_n 法が優れていることが分かる。

2.4 モデル 4

このモデルは六角体系の小型高速炉であり、今回の 4 つのモデルの中で、唯一、ヘキサー Z 体系のモデルである。まず、モンテカルロ法については、 k_{eff} 、制御棒反応度価値ともに標準偏差が小さく、安定した結果が得られている。それに対し、 P_7 、 S_8 計算による結果は標準偏差が大きく、特に、 S_8 はモデル 1、2、3 に比べて、各々の結果にバラツキがあることが分かる。Fletcher の結果によると、通常メッシュ (5 cm) の P_7 計算では、制御棒半挿入と全挿入の結果は、モンテカルロ法による結果から大きく外れているが、half メッシュで計算することで、結果がモンテカルロに近づき、その結果、制御棒反応度価値もモンテカルロ法の平均値に近付くことが分かった。また、 S_8 法の結果は、モンテカルロ法、 P_n 法の結果から外れており、

制御棒反応度価値では～2%減少している。今後、六角-Z体系でのS_n法の精度の改善が望まれる。また、このモデルではWagnerが輸送ノード法による計算結果を出しており、この手法では、全ての場合において、モンテカルロ法に比べ過小評価になっているが、制御棒反応度価値はモンテカルロ法による結果と良く一致している。

2.5 結論および今後のベンチマーク

以上のモデル1～4の結果に対する結論として次のことが示された。

モデル1：k_{eff}に対するモンテカルロ法とS_n法の精度は良いが、制御棒反応度価値はモンテカルロが 1.63×10^{-2} 、S_nで 1.58×10^{-2} であり、相対差は4%で大きいが1σ内にある。S_n法による熱中性子束の収束性がこの差に影響しているとは思われない。

モデル2：ソ連のモンテカルロ計算結果を除くとモンテカルロ法によるk_{eff}の結果及び制御棒反応度価値の結果は良く一致しており、標準偏差も小さい。しかし、S_nとモンテカルロ法の結果は 0.05×10^{-2} の差がある。この差を問題にするには、モンテカルロ法計算のヒストリー数を増加する必要がある。

モデル3：S_n法とモンテカルロ法のk_{eff}の一致は良い。制御棒挿入時を除けば制御棒反応度価値に対する両者の差は 0.05×10^{-2} であり、モンテカルロ法の2σの差がある。これについては領域平均について比較検討すべきである。

モデル4：モンテカルロ法とP_n法、NT法(Wagner)の一一致は良い。S_n法については六角-Z体系より三角-Z体系での取り扱いの方が良いであろう。S_n法、P_n法の空間メッシュ効果については検討が必要。

今後のベンチマークとしては

- 六角-Zの大型高速炉モデル
- 小型炉で冷却材の密度を変化させてその反応度価値を検討する。
- 格子内の非均質計算モデル

を考え、Fletcher、Doriath、Brunaが案を作ることが決まった。

3. 六角体系での3次元中性子輸送コード開発

六角体系でのS_n法の精度が悪いのは、空間メッシュ効果が効いているためである。XYZ体系のように簡単にメッシュを細かくできないので、解決策としては、修正粗メッシュ法、ダイアモンド差分近似の改良、ノード法の適用、等が考えられる。

3.1 修正粗メッシュ法

六角体系において、Fig. 2に示すように7個のサブメッシュを考えて、各サブメッシュにおいて、3組の対辺での中性子束に対し、それぞれ、線形差分を適用する。結果から不要な変数を消去すると、次の重みつきダイヤモンド差分式が得られる（z方向省略）。

$$2\psi^V = (1-\alpha_x)\psi^S_{-x} + (1+\alpha_x)\psi^S_{+x}$$

$$\begin{aligned}
 &= (1-\alpha_u) \psi^S_{-u} + (1+\alpha_u) \psi^S_{+u} \\
 &= (1-\alpha_v) \psi^S_{-v} + (1+\alpha_v) \psi^S_{+v}, \quad (1)
 \end{aligned}$$

$$\alpha_x = \frac{3}{16} \beta \frac{h_p}{\mu_x}, \quad \alpha_u = \frac{3}{16} \beta \frac{h_p}{\mu_u}, \quad \alpha_v = \frac{3}{16} \beta \frac{h_p}{\mu_v}, \quad \beta = \sum_t - \frac{s^v}{\phi^v} \quad (2)$$

3.2 ダイヤモンド差分法の改良

六角体系における通常の輸送差分計算法は、Fig. 3 に示した、辺における中性子束 ψ_{+x} , ψ_{+u} , ψ_{+v} を未知量として、3組の対辺について線形差分を仮定する。もし、頂点での中性子束の注目するならば、4頂点での中性子束 ψ_1 ~ ψ_4 を使って、2頂点 5, 6 での中性子束を計算すればよく、線形差分の仮定は2方向で済み、精度の改善が期待される。

3.1.1 手法1

仮定

$$2\psi = \psi_2 + \psi_5, \quad 2\psi = \psi_3 + \psi_6 \quad (3)$$

を差分輸送方程式 (z -方向省略)

$$\begin{aligned}
 &\mu^x \left(\frac{\psi_5 + \psi_6}{2} - \frac{\psi_2 + \psi_3}{2} \right) + \mu^u \left(\frac{\psi_6 + \psi_1}{2} - \frac{\psi_3 + \psi_4}{2} \right) \\
 &+ \mu^v \left(\frac{\psi_4 + \psi_5}{2} - \frac{\psi_1 + \psi_2}{2} \right) + \frac{3}{2} h_p \sum_t \psi = \frac{3}{2} h_p S \quad (4)
 \end{aligned}$$

に代入して ψ_5 , ψ_6 を消去すると、次式が得られる。

$$\psi = \frac{2\mu^x(\psi_2 + \psi_3) + \mu^u(-\psi_1 + 2\psi_3 + \psi_4) + \mu^v(\psi_1 + 2\psi_2 - \psi_4) + 3h_p S}{4\mu^x + 2\mu^u + 2\mu^v + 3h_p \sum_t} \quad (5)$$

3.1.2 手法2

手法1によると、(5)式で求まる ψ と、すでに求まっている ψ_1 , ψ_4 の間に線形性が成立たず、頂点 1, 4 で次式により ψ'_1 , ψ'_4 が定義される。

$$2\psi = \psi_1 + \psi'_4, \quad 2\psi = \psi_4 + \psi'_1 \quad (6)$$

差分方程式(4)において $(\psi_5 + \psi_1)/2$ は辺の中性子束 ψ^{+u} 、 $(\psi_4 + \psi_5)/2$ は ψ^{+v} であるが (Fig. 3 参照)、これらの辺の中性子束を計算するときは、中心の中性子束(5)を使って(6)式で定義される ψ'_1 、 ψ'_4 を使って、中性子束の不連続性を許した方が良いかもしれない。すなわち、(4)式の代りに

$$\begin{aligned} & \mu^x \left(\frac{\psi_5 + \psi_6}{2} - \frac{\psi_2 + \psi_3}{2} \right) + \mu^u \left(\frac{\psi_6 + \psi'_1}{2} - \frac{\psi_3 + \psi_4}{2} \right) \\ & + \mu^v \left(\frac{\psi'_4 + \psi_5}{2} - \frac{\psi_1 + \psi_2}{2} \right) + \frac{3}{2} h_p \sum_t \psi = \frac{3}{2} h_p S \end{aligned} \quad (7)$$

とし、これと、(3)、(6)の5式より ψ_5 、 ψ_6 、 ψ'_1 、 ψ'_4 を消去すると、次式が得られる。

$$\psi = \frac{2\mu^x(\psi_2 + \psi_3) + 2\mu^u(\psi_3 + \psi_4) + 2\mu^v(\psi_1 + \psi_2) + 3h_p S}{4\mu^x + 4\mu^u + 4\mu^v + 3h_p \sum_t} \quad (8)$$

3.3 検証計算

3.3.1 ベンチマーク・モデル4

モデル4の小型高速炉体系に手法1、2、および修正粗メッシュ法を適用した。Table 4に実効増倍率についての結果を、モンテカルロ法、通常法と比較して示す。制御棒引抜き時については、どの方法もほぼ 1% 以内でモンテカルロ法と一致している。しかしながら、制御棒挿入時には、通常法は約 6% の過大評価となり、これは修正粗メッシュ法によっても改善されない。手法1は2% の過小評価であり、この場合、修正粗メッシュ法は収束しなかった。これに対して、手法2は良好な結果を示し、修正粗メッシュ法を使わなくとも -0.9%、使った場合は -0.3% の誤差であり、制御棒反応度価値も大きく改善された。

3.3.2 高転換軽水炉模擬集合体

Figure 4 に示す模擬集合体について、7群断面積を使い、制御棒引抜き、挿入時について計算を行った。修正粗メッシュ法は手法2についてのみ適用した。基準計算はモンテカルロコード GMVP⁽⁴⁾ で行った。実効増倍率についての結果を Table 5 に示す。制御棒引抜き時には、この例でもすべての方法が基準値と 1% 以内で一致している。制御棒挿入時には、通常法は収束しない、手法1は良好（ただし後で示す出力分布は良くない）、手法2は大きな過小評価になるが、修正粗メッシュ法により大幅に改善される。出力分布の結果を Fig. 5 に示す。手法1は 2-4% の誤差を与えるが、修正粗メッシュ法を使った手法2では、それが大幅に改善されることが分る。

4. 結論

ベンチマークの結果、以下の結論が得られた。

- (1) XYZ 体系ではモンテカルロ法と S_n ($n=8$) 法は良く一致する。実効増倍率については、モンテカルロ法の 1-2% で一致し、領域平均中性子束では、モンテカルロとの差は 1% 以内である。
- (2) P_n 法は空間メッシュ効果が大きく、とくに、FBR の XYZ モデルでモンテカルロ法との一致が悪くなる。
- (3) 六角-Z 体系では S_n 法は計算精度が悪い。
- (4) ここで提案したベンチマークは、今後、3 次元中性子輸送コードの開発における精度の検証に有用と思われる。

ベンチマークの結果を受けて、六角-Z 体系での精度を改善するために、ダイヤモンド差分の新しい手法 1、2、および、修正粗メッシュ法を開発した。ベンチマークモデル 4 および高転換軽水炉集合体で検証計算を行ったところ、手法 2 による修正粗メッシュ法により、モンテカルロの基準計算と良く一致する結果を得た。

参考文献

- (1) T. Takeda and H. Ikeda: 3-D Neutron Transport Benchmarks, NEACRP-L-330 (1991).
- (2) T. Takeda and H. Ikeda: J. Nucl. Sci. Technol. 28(7), 656 (1991).
- (3) T. Takeda, M. Yamasaki, and H. Ikeda: A Weighted Diamond Difference Approximation for Improving 3-D Coarse Mesh S_n Calculations, to be reported at 1992 Topical Meeting on Advances in Reactor Physics, March 8-11, 1992, Charleston, SC.
- (4) M. Nakagawa et al.: Development of Monte-Carlo Code for Particle Transport Calculations on Vector Processors: Proc. Intn'l Conf. on Supercomputing in Nuclear Applications, Mito, Japan, March 12-16, 1990.

Table 1 List of Participants

Participant	Code	Method
R. W. Shaaffer	VIM	Monte-Carlo
P. A. Landeyro	KENO IV	Monte-Carlo
H. Rief	KENEUR	Monte-Carlo
U. Wehmann	MOCÀ	Monte-Carlo
E. Seifert	OMEGA	Monte-Carlo
V. I. Bryzgalov	MCU	Monte-Carlo
M. Nakagawa	GMWP	Monte-Carlo
R. E. Alcouffe	THREEDANT	SN
G. Bucket	TRITAC	SN
G. Palmiotti	TRITAC	SN
S. M. Lee	TRITAC	SN
L. N. Yaroslavzeva	JSP-SN	SN
T. Yamamoto	TRITAC	SN
Y. Kaise	ENSEMBLE-K	SN
T. Takeda	TRITAC	SN
J. K. Fletcher	MARC-PN	PN
G. Palmiotti	MARC-PN	PN
K. Kobayashi	PLXYZ	PN
P. J. Collins	DIF3D-NODE	Nodal
M. R. Wagner	HEXNOD	Nodal
Y. N. Mironovich	CMEZ	Synthesis
G. Palmiotti	CCRR	Synthesis
H. Ait Abderrahim	DORT3.5	Synthesis
R. Roy	DRAGON	Synthesis

Table 2 Participants to the Working Group Meeting

参加者	機関
H. A. Abderrahim	SCK/CEN(ベルギー)
R. Alcouffe	Los Alamos National Labo.(米)
G. Buruna	Framatome(仏)
J. Y. Doriath	CEN/Cadarache(仏)
J. J. Lautard	
J. K. Fletcher	Risley Nuclear Power(英)
P. A. Landeyro	ENEA(伊)
H. W. Rief	ANL(米)
R. W. Schaefer	Kurchatov Institute(ソ連)
I. S. Slesarev	Siemens AG/KWU(独)
日本	動燃
竹田 錦織 池田 阪大	
E. Sartori	NEA Secretariat OECD/NEA フランクフルト(独)

Table 3 Results for k-eff and the Control Rod Worths

Model	keff		CR-worth ($\Delta k/kk'$)
	rod out	rod in	
Small LWR			
	Monte-Carlo	0.9778	1.63E-02
		± 0.00046	$\pm 0.07E-02$
	Pn	0.9766	1.45E-02
		± 0.00058	$\pm 0.22E-02$
	Sn	0.9772	1.58E-02
		± 0.000068	$\pm 0.003E-02$
Small FBR			
	Monte-Carlo	0.9731	1.52E-02
		± 0.00020	$\pm 0.03E-02$
	Pn	0.9794	1.56E-02
	Sn	0.9734	1.52E-02
		± 0.00017	$\pm 0.007E-02$
	Nodal Transport	0.9714	1.54E-02
Large FBR			
	Monte-Carlo	1.0005	3.05E-02
		± 0.00020	$\pm 0.03E-02$
	Pn	1.0040	2.74E-02
	Sn	1.0005	3.11E-02
		± 0.00047	$\pm 0.04E-02$
	Nodal Transport	0.9996	3.10E-02
Small FBR with Hexagonal-Z			
	Monte-Carlo	1.0951	2.23E-01
		± 0.00035	$\pm 0.005E-01$
	Pn	1.0942	2.20E-01
		± 0.0015	$\pm 0.12E-01$
	Sn	1.0887	2.02E-01
		± 0.0043	$\pm 0.10E-01$
	Nodal Transport	1.0889	2.25E-01

Table 4 Results of the Present Coarse Mesh Method
for Model 4

	制御棒 引き抜き時	制御棒 挿入時	制御棒 反応度価値 (%Δk/k')	
Monte-Carlo [*]	1.0951 (±0.004)*	0.8799 (±0.0003)	22.3	
通常法	1.1033 (+0.75%)**	0.9355 (+6.32%)	16.3 (-26.9%)	通常法 (+0.21%)**
手法1	1.0888 (-0.58%)	0.8584 (-2.44%)	24.7 (+10.8%)	手法1 (+0.36%)**
手法2 (修正なし)	1.1038 (+0.79%)	0.8723 (-0.86%)	24.0 (+7.6%)	手法2 (修正なし) (-0.05%)
手法2 (修正あり)	1.1034 (+0.76%)	0.8768 (-0.35%)	23.5 (+5.4%)	手法2 (修正あり) (-0.21%)

* 標準偏差

** Monte-Carlo計算との差

※ Wehmann氏によるMOCA、Bryzgalov氏によるMCU、
Seifert氏によるOMEGA、中川氏によるGMVPの平均値

Table 5 Results of the Present Coarse Mesh Method
for a High Conversion Fuel Assembly

	制御棒 引き抜き時	制御棒 挿入時	制御棒 反応度価値 (%Δk/k')	
GMVP [*]		1.1616 (±0.0008)*	0.7866 (±0.0004)	41.0
通常法		1.1640 (+0.21%)**		Not converge
手法1		1.1658 (+0.36%)**	0.7920 (+0.69%)	40.5 (-1.2%)
手法2		1.1610 (-0.05%)	0.7687 (-2.28%)	43.9 (+7.1%)
手法2 (修正あり)		1.1592 (修正あり) (-0.21%)	0.7836 (-0.38%)	41.3 (-0.7%)

* 標準偏差

** GMVPとの差

※日本原子力研究所の奥村啓介氏のご協力による計算結果

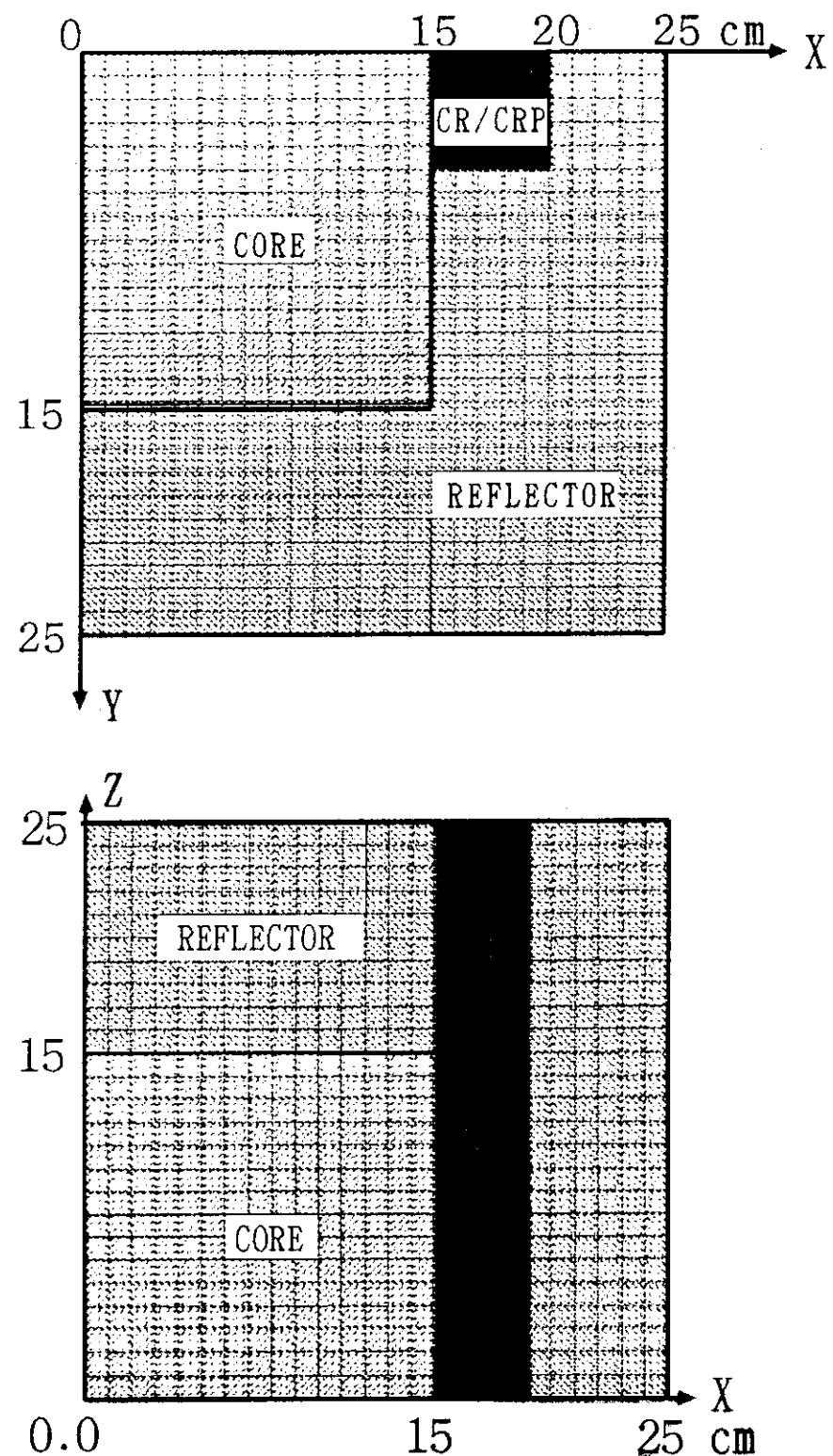


Fig. 1a Core configuration of Model 1 (small LWR)

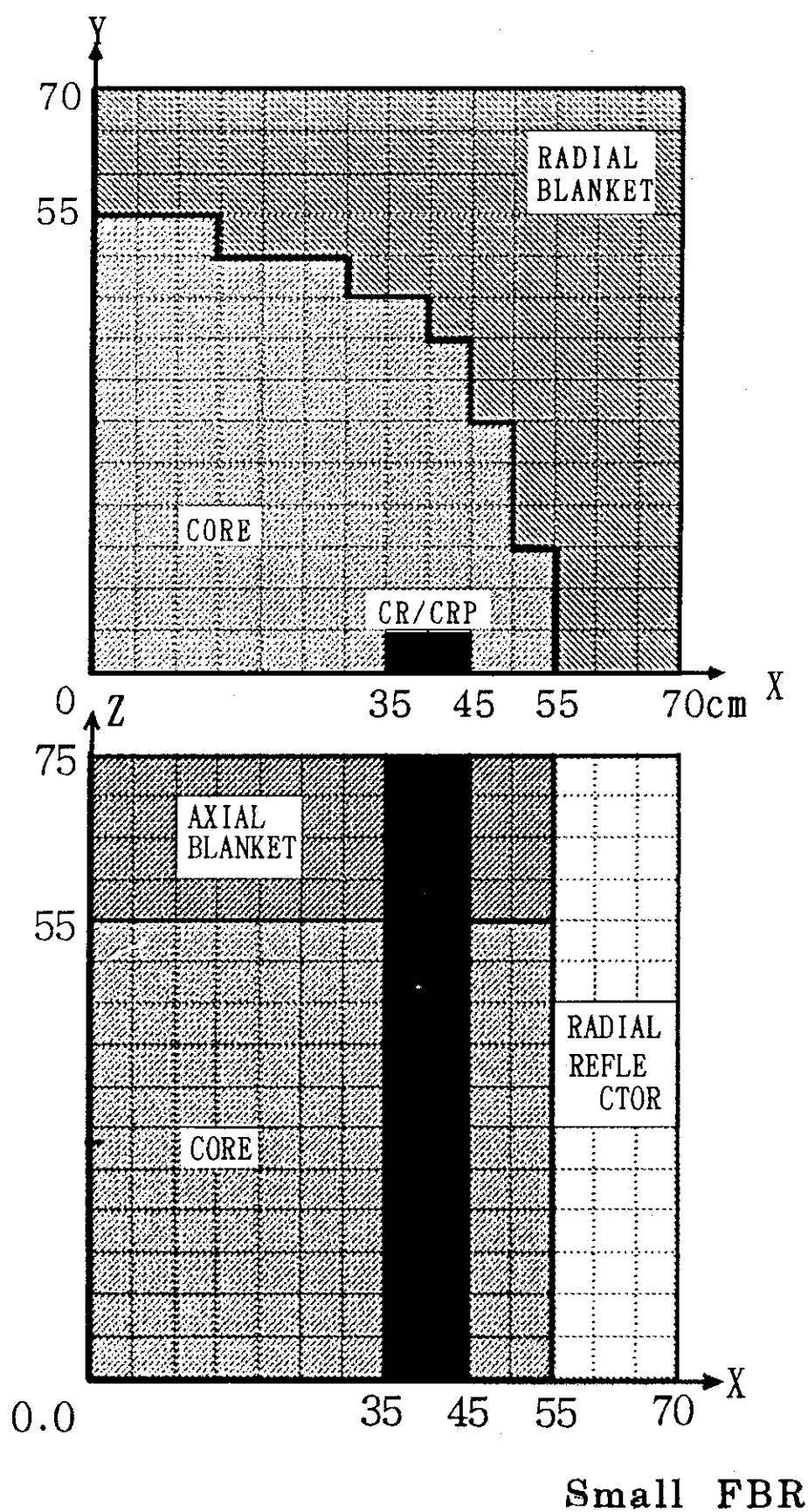
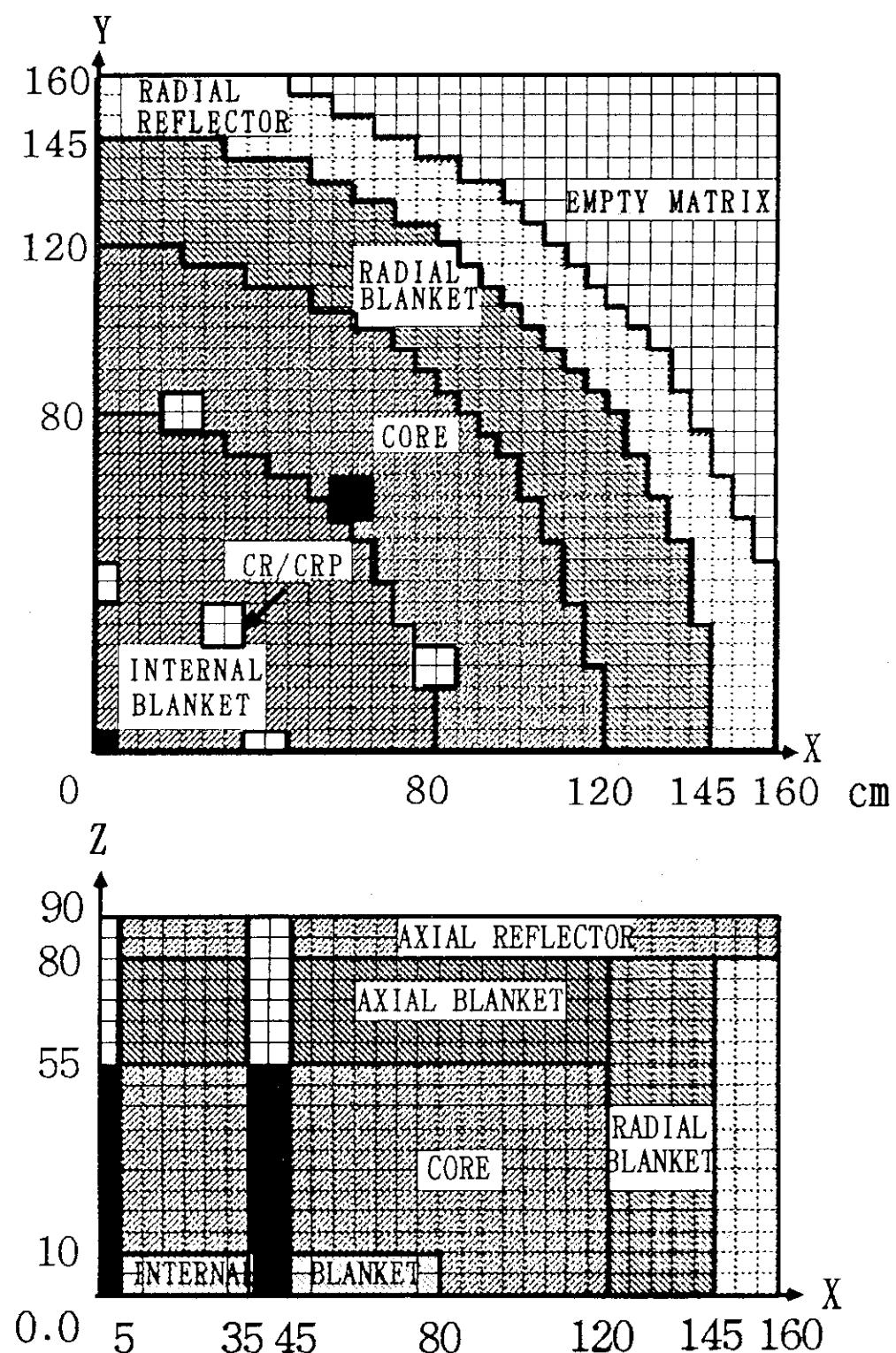
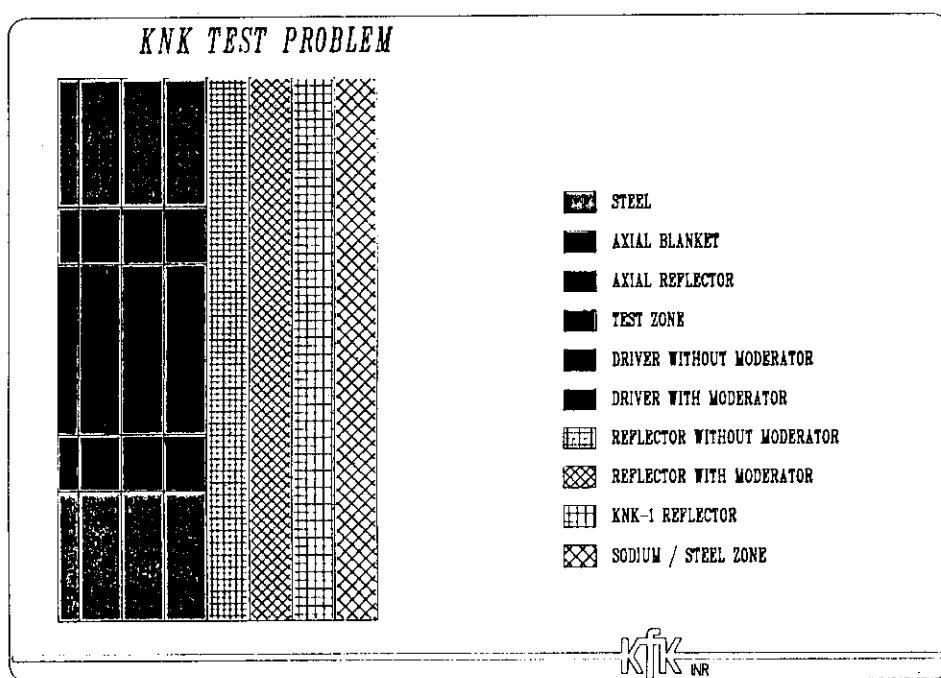
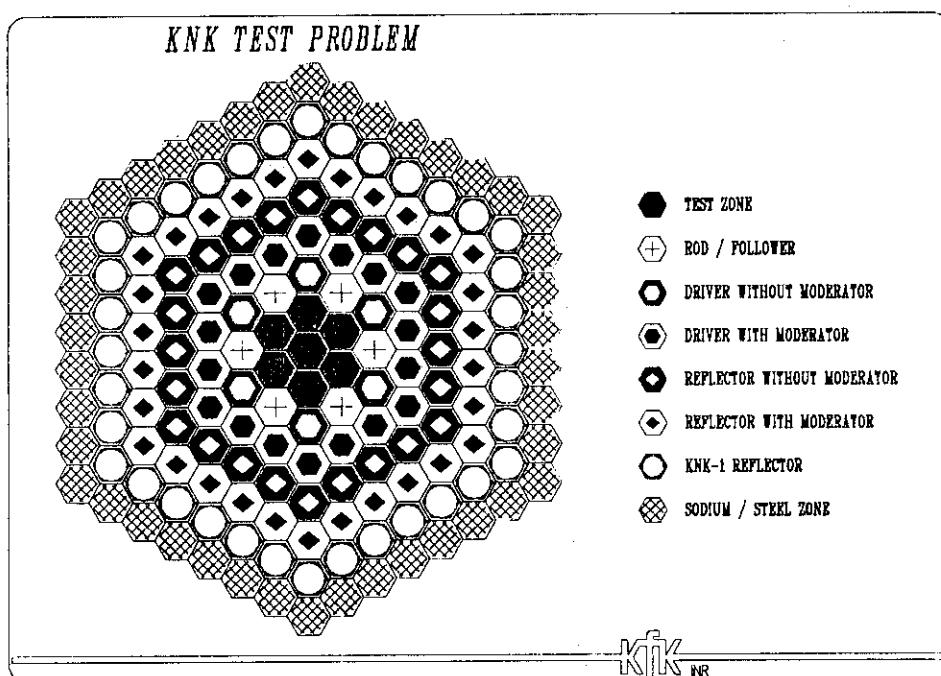


Fig. 1b Core configuration of Model 2 (small FBR)



**Fig. 1c Core configuration of Model 3
(axially heterogeneous large FBR)**



**Fig. 1d Core configuration of Model 4
(small FBR with hexagonal-z geometry)**

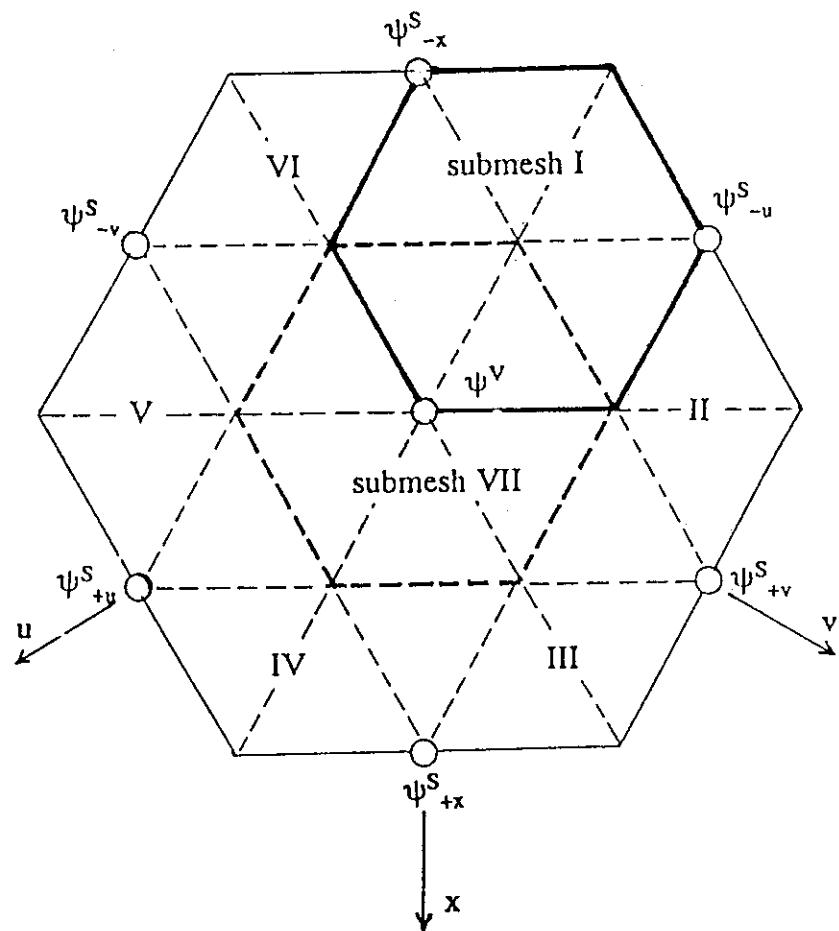


Fig. 2 Seven submeshes in a hexagon

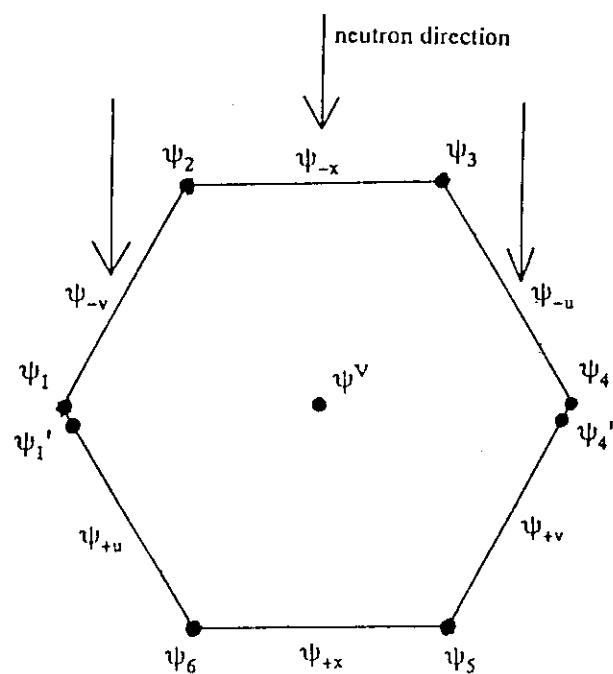


Fig. 3 Surface fluxes and vertex fluxes

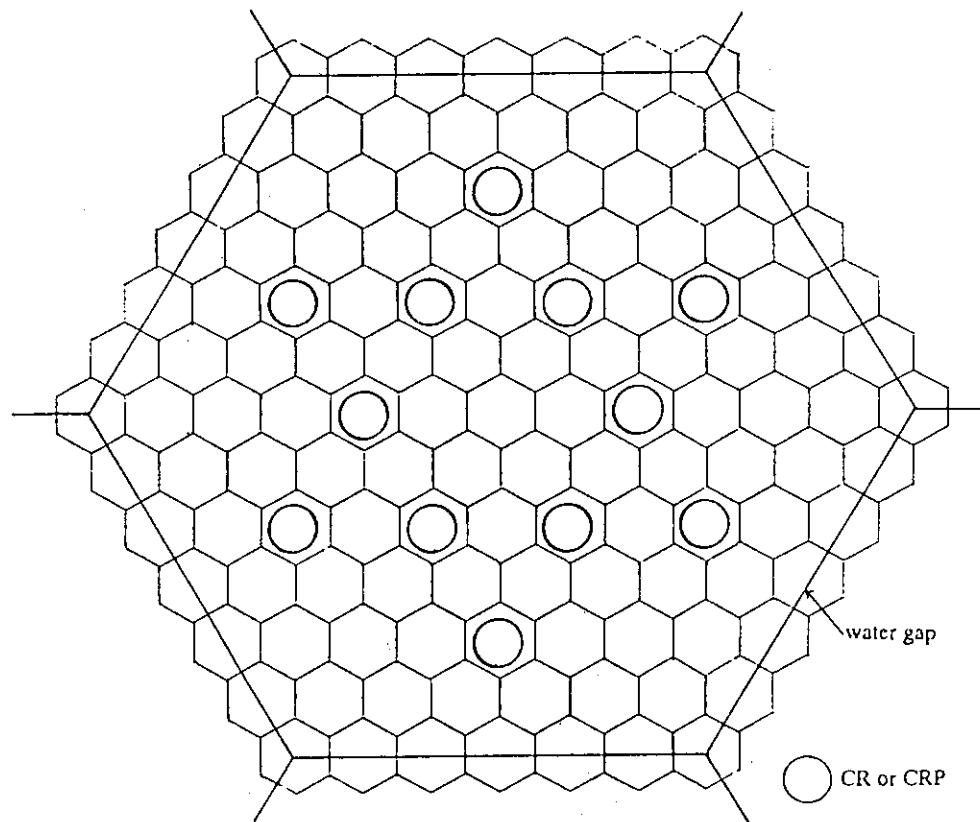
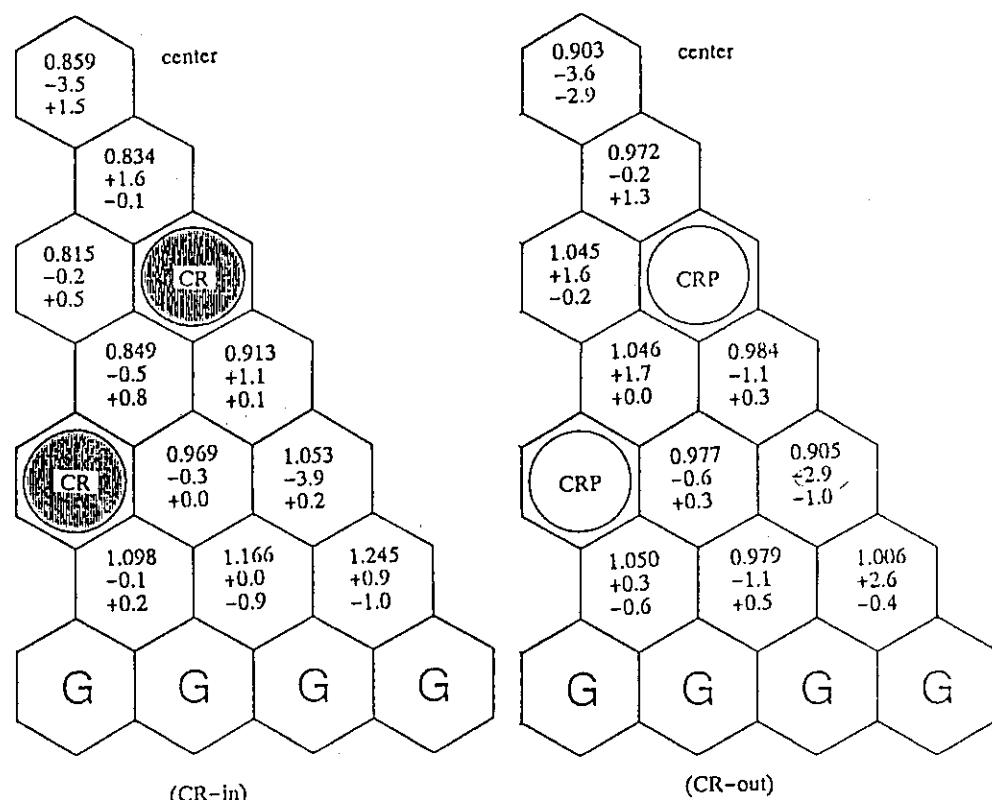


Fig. 4 A model of a high conversion fuel assembly



Reference
 Scheme 2
 Scheme 3 (with present method) (%error from reference)
 (%error from reference)

Fig. 5 Power distribution in a hexagonal fuel assembly

9. 3次元動特性解析コードによるBWRの安定性の解析

瀧川 幸夫, 竹内 豊, 江畠 茂男 (東芝)

近年、国外の幾つかの沸騰水型原子炉（BWR）において2種類の異なったモードの中性子束振動現象が観測されている。1つは中性子束が炉心全体同位相で振動するモード（炉心一体振動モード）で、La Salle 2号炉（米）においてはスクラムに至るほどの大きな振幅の振動であったためにBWRの安定性に対する関心を呼び起こした。他の1つはCaorso炉（伊）で代表される、炉心の半分が他の半分と逆相で振動するモード（領域振動モード）である。これらの中性子束振動現象は核熱水力的な不安定事象であり、振動の検出性や燃料健全性への影響を十分評価する必要がある。特に、領域振動は局所的に振幅の大きな出力振動を引き起こす可能性があるため3次元的な解析評価が必要となる。そこで3次元動特性解析コードTOSDYN-2による解析を行ない領域振動の発生条件について検討すると共に、2つのモードの中性子束振動時における燃料の熱的余裕評価を行なった。

BWR Stability Analysis with Three-Dimensional Transient Code

Yukio TAKIGAWA, Yutaka TAKEUCHI, Shigeo EBATA (Toshiba)

Recently, neutron flux oscillations of two different modes were observed in several foreign BWR plants. One is corewide oscillation mode which is characterized by a phenomenon that neutron flux oscillates in-phase over a whole core. At La Salle 2 plant (U.S.A.), the amplitude of core wide neutron flux oscillation grew considerably large to result in a reactor scram, which aroused great concern about BWR stability. The other is regional oscillation mode which is characterized by the phenomenon, as typically observed at Caorso plant (Italy), that neutron flux of a half core oscillates out-of-phase to that of the other half core. These neutron flux oscillation phenomena were caused by nuclear-thermalhydraulic coupled instability and requires an evaluation study on oscillation detectability and effect on fuel integrity. Particularly, the regional oscillation mode requires three-dimensional analysis since it may bring about locally large amplitude power oscillation. For this reason, analysis was done with the three-dimensional transient code TOSDYN-2 to study reactor condition which causes the regional oscillation and also to evaluate fuel thermal margin under the neutron flux oscillations of these two instability modes.

Keywords (BWR stability, corewide oscillation mode, regional oscillation mode, La Salle 2, Caorso, three-dimensional analysis, TOSDYN-2, thermal margin)

1. はじめに

1998年3月に米国の La Salle-2 号炉で発生した中性子束振動現象⁽¹⁾は、再循環ポンプ2台停止後の自然循環状態において給水加熱器の隔離による給水温度低下が重なって生じたもので、中性子束振動の振幅が大きく増加して、炉心平均出力検出器（APRM）がスクラム設定値を超えたためスクラムにより自動停止した事象であるが、これを契機として、世界的にBWRの安定性が注目されるようになった。BWRの安定性については、不安定状態では原子炉出力や流量の振動が生じて燃料健全性に悪影響を及ぼす可能性があるために多くの研究が行なわれてきた。BWRの炉心は多数の燃料チャンネルが配置された並行流路体系をなしており、冷却材は水-蒸気の2相流となって炉心で発生する熱を除去するが、この2相流には、流量、ボイド率、圧力損失間のフィードバックに基づく熱水力的振動、所謂、密度波振動の可能性があることが知られている。また、ボイド率や燃料棒温度の変化を通しての核的なフィードバック効果も存在する。このように、BWRは2相流熱水力特性と核特性及び再循環流路の動特性が複雑に絡み合ったフィードバック系を構成しているため、条件によっては不安定状態になる可能性がある。従来、BWRの安定性設計では、「通常の運転領域内では安定とする」との基準が採用されており、この基準が満たされているか否かを解析により評価している。従って、以前は安定性に関する研究も主に不安定状態が生じる限界の条件や安定限界に対する余裕を精度良く評価することに焦点が置かれていた。

国外においては、数多くの実機プラントにおいて安定度の評価を行なうために、プラントに外乱を与えて応答を見る安定性試験が行なわれている。最初に行なわれた1977～78年のPeach Bottom 2号炉⁽²⁾では全て安定な領域内の運転点での安定度測定試験であったが、それ以後の安定性試験では安定度測定試験に加え、安定限界評価のために不安定振動発生の運転点での試験も行なっている。これらの不安定振動点のデータによると、中性子束はいずれも比較的小振幅のリミットサイクル振動を示しているが、その振動モードは2種類に分類されることがわかった。1つは、Vermont Yankee炉（米）⁽³⁾、TVO-II炉（スウェーデン）⁽⁴⁾に見られるような、炉心全体同位相の振動で、このような現象を「炉心不安定」、このようなモードに関する安定性を通常、「炉心安定性」と呼ぶ。他の1つは、Caorso炉（伊）⁽⁵⁾、KRB-B/C炉（独）⁽⁶⁾に見られるような、炉心の半分と他の半分が互いに逆相で応答する振動で、このような現象を「領域不安定」、このようなモードに関する安定性を「領域安定性」と呼んでいる。領域不安定に関しては、局所的に振幅の大きな出力振動を引き起こす可能性があり、且つ、炉心平均出力は逆相部分がキャンセルされて小さくなるため、振動の検出性や燃料健全性への影響を十分評価する必要がある。

前述のLa Salle-2号炉の中性子束振動現象は炉心不安定に属するが、振幅がこれまでの安定性試験時に比べて大きかったことから、領域不安定に関しても、これまで観測されている以上に大きな振幅の局所的出力振動を引き起こすことを懸念して引き続き検討が行なわれている。領域不安定はその性質上、3次元的な解析評価が必要となる。

以下では、3次元動特性解析コードTOSDYN-2の解析モデル、実機データによる検証及び、領域不安定の発生条件について検討や、2つのモードの中性子束振動時における燃料健全性評価として熱的余裕評価を行なった結果について述べる。

2. TOSDYN-2モデル

TOSDYN-2コードは3次元時間領域動特性解析コードであり、次の4つの部分モデルから構成されている。

- 1) 核特性モデル
- 2) 炉内熱水力モデル
- 3) 燃料棒熱伝達モデル
- 4) 炉心外再循環モデル

各部分モデルはデータのやりとりを通じて結合されているが、各モデル間のデータの流れを Fig.1 に示す。

2.1 核特性モデル

中性子動特性モデルは3次元修正1群拡散モデルで遅発中性子先行核は6群を考慮しており、核設計に用いている3次元定常計算コードと同じモデルである。基礎方程式は直交座標系のノードに関して差分化し、初期定常計算は固有値問題として、また、過渡計算は反復法により解いている。ノードサイズは2つのオプションを有しており、通常は計算時間の短縮化をはかるために、径方向には、制御棒の周囲4バンドルを1ノード、軸方向には12ノードとする標準ノード分割を採用しているが、核設計コードと同じように、径方向には燃料バンドル1体を1ノード、軸方向には24ノードとする詳細ノード分割による計算を行なうことも可能である。反応度フィードバック効果は、熱水力モデルで計算される減速材密度と、燃料棒熱伝達モデルによって計算される燃料棒温度に対する核定数の依存性を通して考慮される。また、制御棒の挿入、引き抜きによる反応度変化も考慮できる。即ち、

$$\Sigma = \Sigma (U, T, C)$$

ここで、	Σ	： 核定数
	U	： 減速材密度
	T	： 燃料棒温度
	C	： 制御棒密度

また、実機データとの比較を行なうために、局所中性子束検出器 (LPRM) やAPRM等の核計装系モデルも含んでいる。

2.2 炉内熱水力モデル

炉心内の燃料チャンネル内外の流れを扱う熱水力モデルは、軸方向1次元5保存式の分離流モデルで、質量とエネルギーの保存式は蒸気と混合流に対して、運動量保存式は混合流に対して考

慮している。但し、圧力の軸方向変化の物性値に及ぼす影響は小さいことを考慮して、運動量保存式は質量、エネルギー保存式とは分離して計算している。尚、軸方向のノード数は24である。また、2相流に関しては、サブクール沸騰を考慮した熱的非平衡モデルであり、ボイドークオリティ相関式としてドリフトフラックスモデルを採用している。

更に、燃料健全性を評価するための熱的余裕は設計で用いているGEXL相関式⁽⁷⁾により計算する。BWRにおいては、燃料健全性は沸騰遷移(Boiling Transition: BT)の発生の有無により判断している。BTは燃料棒被覆管表面の液膜が消失して熱伝達が急激に悪化する現象であり、BTが生ずる燃料バンドル出力を限界出力と呼んでいるが、熱的な余裕は、この限界出力と実際の運転時のバンドル出力の比として定義される限界出力比(Critical Power Ratio: CPR)で評価している。

炉心内の燃料バンドルは、出力や軸方向出力分布等の熱水力条件が様々であるが、数百本の燃料バンドルがあるため、全バンドルを個別に扱うのは、現状の計算機性能では実用的とは言えない。そこで、熱水力条件や、炉心内位置を考慮に入れて特性の類似した燃料バンドルを幾つかのグループに分類し、計算はこの熱水力グループの代表バンドル(以下、チャンネルタイプと呼ぶこととする。)について行なう。従って、熱水力的には、選定したグループ数のチャンネルタイプが、炉心の上下プレナムを共通ヘッダーとして結合した並行多チャンネル体系として取り扱っていることに相当する。これらのチャンネルタイプは流量配分を通して熱水力的に結合している。即ち、各チャンネルタイプの流量は、炉心の上下プレナム間の圧力損失が共通になるように炉心入口流量を配分する繰り返し計算により決定している。

2.3 燃料棒熱伝達モデル

燃料棒の温度計算は軸方向の熱伝導を無視した径方向1次元の熱伝導方程式を差分化して解いており、燃料ペレット、被覆管の物性値に関してはその温度依存性を考慮している。計算は、各熱水力チャンネルタイプ内の平均燃料棒について行なう。

2.4 炉心外再循環モデル

炉心外モデルはBWRの過渡解析設計コードのモデル⁽⁸⁾と同じで、Fig. 2に示すようにプラントの主要な系統、機器を全て含んでいる。即ち、気水分離器、ジェットポンプ等の圧力容器内の機器、主蒸気管、給水系統、再循環系統の各モデル、及び、プラントの各種制御系モデル等を含んでいる。このモデルにより、炉心計算のための境界条件となる、圧力、入口流量、入口エンタルピーが計算される。

2.5 核熱水力モデル結合方式

前述の如く、核特性モデルと、熱水力／燃料棒熱伝達モデルではノード分割方法が異なるために両者間の入出力データは1対1には対応しない。そこで、以下的方式によりデータの受け渡しを行なう。

1) 核特性モデル→熱水力／燃料棒熱伝達モデル

核特性モデルからは、燃料棒内、及び、冷却材中の発生熱が、各々、燃料棒熱伝達モデルと熱水力モデルに対して出力される。基本的には同一熱水力グループ内に属する各バンドルの量を平均化するが、ホットテストチャンネルの取り扱いを考慮して、特定のバンドルで代表させることも可能である。

$$\begin{aligned} Q_n &= \sum Q_{i,j} / N_n \quad (i, j) \in n && \text{(通常)} \\ Q_n &= Q_{i,j} \quad (i, j) \in n && \text{(オプション)} \end{aligned}$$

ここで、 Q は発生熱を、添字の (i, j) は核特性モデルの径方向ノードの座標を表わす。また、 n は熱水力グループを、 N はグループ内の燃料バンドル数を表わす。

2) 热水力／燃料棒熱伝達モデル→核モデル

熱水力モデルと燃料棒熱伝達モデルからは、各々、減速材密度(U)と燃料棒温度(T)が核特性モデルに対して出力される。ここで、 U と T の過渡変化中の値は、初期の三次元分布に、各熱水力グループにおける計算結果の初期からの偏差分を加えることにより、等価的に三次元効果を考慮する。即ち、

$$\begin{aligned} U_{i,j}(t) &= U_{i,j}(0) + [U_n(t) - U_n(0)], \quad (i, j) \in n \\ T_{i,j}(t) &= T_{i,j}(0) + [T_n(t) - T_n(0)], \quad (i, j) \in n \end{aligned}$$

以上のように、ノード分割方法の異なる核特性モデルと熱水力／燃料棒熱伝達モデルの結合を行なうが、この対応関係を Fig. 3 に示す。

3. TOSDYN-2コードの実機データによる検証

TOSDYN-2コードに関してはこれまでに、多数の実機安定性試験データを用いた検証を行なっている。^{(9) (10) (11)} また、熱流動試験装置を用いた熱水力安定性試験データによる検証や、プラントの起動試験データを用いた検証⁽¹²⁾ も行ないコードの有効性を確認している。ここでは、実機での不安定振動データの中から、3次元効果が顕著に現われている領域不安定の例として Caorso 炉を、また、炉心不安定の例としては発振後の振幅が最も大きくなった La Salle-2 号炉をとりあげ各々に関する解析について述べる。

3.1 Caorso 炉領域不安定事象の解析

3.1.1 実機データ

領域振動は、自然循環状態で出力を増加させていく過程で発生している。測定されたLPRMの時系列データを見ると、炉心の比較的周辺部で局所的に発生した小振幅の振動が徐々に成長していくと同時に炉心中心に対して対称な位置で逆相の小振幅振動が発生し、次第に振幅も増加しながら炉心全体に広がっていき、最終的に炉心の半分が他の半分と逆相で振動する定常的なリミットサイクル振動となっている。測定されたLPRMの中で振幅が最大のものは約60%であった。APRMは平均化によるキャンセリング効果により、約10%程度であったが、十分検出可能なレベルであった。Caorso炉において測定されたLPRMの位置をFig.4に、最大振幅を示したLPRMを含む対角線上のLPRMの応答をFig.5に示す。APRMの応答はFig.6に見られるような不規則な波形となっているが、これは高次モード振動ではなく、単に、平均するLPRMの応答が必ずしも0度や180度ではない位相差を有しているためである。また、炉心全体の逆相振動の様子をFig.7に示す。

3.1.2 TOSDYN-2コードによる解析

核特性モデルのノードは標準ノード分割を採用しているが、この分割方法の妥当性を見るために領域振動が開始した炉心状態での出力分布について実機データと解析結果を比較してFig.8に示す。両者は良く一致しており、標準ノード分割でも十分であることが確かめられた。

TOSDYN-2による安定性解析を行なう際には、チャンネルタイプ分けをどのように決めるかが重要となる。通常はまず、燃料タイプやオリフィスタイプのような形状の相違によって分類した後に、燃料バンドルの出力や、軸方向出力分布を考慮して分類している。また、今回の検証解析のように振動モードが明らかである場合には、その振動モードにみあった空間的分割を行なう。どのようなチャンネルタイプ分けをするにせよ任意性が含まれることは避けられないが、従来は、チャンネルタイプ分けの方法を幾通りか変えて解析を行ない、評価したい項目に対する影響を見て結果の妥当性を判断している。

ここでは、2通りのチャンネルタイプ分けの結果について述べることとする。ケース1は6チャンネルタイプ、ケース2は10チャンネルタイプである。

ケース1のチャンネルタイプ分けをFig.9に示す。チャンネルタイプ1、2は逆相振動の各領域内での振幅が最も大きかったLPRM近傍の高出力バンドルで、熱水力的安定度の悪いホットチャンネルを、また、チャンネル3、4は平均チャンネルを表わしている。チャンネルタイプ5、6はオリフィスが他と異なる周辺チャンネルである。このような6チャンネルタイプは領域振動の解析を行なう場合の最少チャンネルタイプ数である。この場合のTOSDYN-2によるLPRM、APRMの応答を各々Fig.10、Fig.11に示す。尚、解析では、計算時間を短縮するために、初期外乱として、チャンネルタイプ1、2に半周期分の逆相の正弦波状の熱流束外乱を印加してその後の応答を見る方法を採用している。Figure 10より明らかなように持続的な領域振動が現われているが、実機データと比較するとリミットサイクルの振幅が小さい。これはチャンネルタイプ分けが不十分であることが考えられる。

ケース2では、初期出力分布の検討により炉心内の他の高出力バンドルも独立なチャンネルタ

イプとして選び、且つ、各領域の平均チャンネルタイプも更に分割して10チャンネルタイプとした。チャンネルタイプ分けをFig. 12に示す。この場合のTOSDYN-2によるLPRM、APRMの応答を各々Fig. 13、Fig. 14に示す。実機データの図との比較により、ケース2ではLPRM、APRMの応答共に実機データを良く模擬している。

定量的な評価を行なうために、対角線上のLPRMのリミットサイクル振幅を実機と解析で比較してFig. 15に示す。ケース2が実機のLPRM振幅の分布を良く模擬していることが確認される。これらの解析結果により、リミットサイクル振幅の大きさや分布を精度良く評価するためにはチャンネルタイプ分けについて十分検討して決める必要があるが、基本的にはTOSDYN-2コードは領域不安定現象の解析に有効であることが確かめられた。

3.2 La Salle-2号炉炉心不安定事象の解析

3.2.1 実機データ

炉心不安定による中性子束振動は、検査員の誤操作により再循環ポンプ2台がトリップして自然循環高出力の運転状態となり、その後生じた給水温度低下に伴って発生し、スクラムに至っている。振動が炉心内で同相であったこと、及び、運転再開後のオフガス、炉水の検査により燃料破損の兆候は見られなかったことが確認されている。実機で測定されたデータをFig. 16に示す。

3.2.2 TOSDYN-2コードによる解析

炉心不安定振動の解析の場合には、初期外乱としては炉心全体に影響を及ぼすドーム圧外乱を採用している。ここでは、核特性モデルのノードとして詳細ノード分割を採用した解析結果について述べる。チャンネルタイプ分けに関しては領域不安定の解析の場合のように中心対称な領域分けをする必要はなく、且つ、少数のチャンネルタイプ数でも十分であるが、詳細ノード分割に対応させて比較的多数のチャンネルタイプを選び、Fig. 17に示す8チャンネルタイプとした。チャンネルタイプ1は後述する熱的余裕評価のために選んだバンドル出力が最大のホットテストチャンネルであり、チャンネルタイプ2～7はバンドル出力の値を考慮して分割した。チャンネルタイプ8は周辺オリフィスチャンネルである。TOSDYN-2の解析結果をFig. 18に示す。実機ではスクラムにより出力が急減しているが解析では計算を続行させている。Fig. 18より、スクラムに至る炉心不安定振動を良く模擬できており、TOSDYN-2コードは炉心不安定現象の解析にも有効であることが確かめられた。

また、上述の検証結果以外に、安定な運転点での安定性試験データを用いて安定度と周波数を比較した検証結果をFig. 19に、KRB-B/Cプラントの領域不安定データの解析結果から、リミットサイクルの最大振幅の値を実機と比較してFig. 20に示す。TOSDYN-2の解析結果はいづれも実機の結果と良く一致しており、TOSDYN-2コードを安定性解析へ適用する妥当性が示された。

4. 領域不安定発生条件の検討

領域不安定現象に関しては、中性子束の未臨界な高次モードと関連づける研究⁽¹³⁾ や、炉心の半分を独立した炉心と見做し、実効的なボイド係数を定義して安定度を評価する研究⁽¹⁴⁾ 等があるが、ここではTOSDYN-2を用いて領域不安定の発生条件を検討した結果について述べる。

領域不安定現象は、3次元的な核特性と熱水力特性のカップリングによって生じるが、実機の領域不安定現象発生時の炉心状態における出力分布に共通に見られる特徴は、逆相で振動している2つの領域の中で、振幅が最も大きな部分には出力の大きな燃料バンドルが存在していることである。即ち、出力が大きくて熱水力的安定度（チャンネル安定性と称している）の低い燃料バンドルが炉心中心から離れたところに中心に対して対称的な位置に存在している。そこで、このような高出力バンドルの炉心内の位置、即ち、径方向の出力分布の効果について検討した。また、核特性による反応度フィードバック効果がチャンネル安定性に及ぼす影響についても検討した。

解析には、国内の代表的なBWRプラントデータを用いたが、通常の運転範囲では十分安定であるので、自然循環状態で出力を増加させた炉心状態を作成してそのデータを用いた。今回解析した仮想的運転点を Fig. 21 に示す。この状態を基準にして前述の領域不安定解析に標準的に採用する6チャンネルタイプ分けにより、以下の4ケースの解析を行なった。

ケース1： 基本ケース

ケース2： 基本ケースで高出力バンドルの位置を周辺部へ移動

ケース3： ケース2で対称位置にある2つのホットチャンネルの片側のみに外乱印加

ケース4： ケース2で核的フィードバック効果無視（チャンネル安定性）

ケース1のチャンネルタイプ分けと、ホットチャンネルタイプを含む対角線に沿った径方向出力分布、及び解析結果を各々 Fig. 22, Fig. 23, Fig. 24 に示す。これらの図より、径方向出力分布は全体的に滑らかなドーム型の分布であり、領域振動は減衰しており、安定であることがわかる。

ケース2についてのケース1と対応する図を各々 Fig. 25, Fig. 26, Fig. 27 に示す。ホットチャンネルの炉心周辺部への移動は無限倍率の調整により行なった。これらの図より、ホットチャンネルの出力ピーピング係数はケース1と同じであるが、径方向出力分布は周辺ピーク型の分布となっている。このこのケースでは領域振動は少し発散傾向の持続振動になっていることがわかる。ケース1とケース2の比較により、高出力バンドルの炉心内位置、換言すると径方向出力分布が領域不安定現象に対して大きな影響を及ぼすことが明かとなった。

ケース3は、外乱印加方法が結果に及ぼす影響を見たもので、解析結果を Fig. 28 に示す。計算時間が十分でないために発達したリミットサイクルに至っていないが、対称な領域間で応答に位相差が現われ次第に逆相の振動に近づいて行く様子が見られる。この結果により、解析で採用している逆相の外乱印加方法は計算時間短縮に有効であることがわかる。

ケース4は、核的フィードバック効果を評価するために出力を一定とした計算で、チャンネル安定性の評価に相当する。流量の応答結果を Fig. 29 に示すが、振動は減衰しており、熱水力的には安定であることがわかる。従って、核的フィードバック効果は不安定化の作用があることがわかる。

以上の検討により、領域不安定現象は、「出力が大きくてチャンネル安定性の安定度の低い燃料バンドルが炉心中心から離れたところに中心に対して対称的な位置に存在する」場合に発生しやすいことが確認され、実機データに共通に見られた傾向を裏付けた結果となっている。

5. 不安定振動時の熱的余裕評価

5.1 La Salle-2号炉炉心不安定事象の解析

La Salle-2号炉炉心不安定振動時の熱的余裕は、Fig. 18 のCPRの応答によって評価されるが、CPRの最小値は、出力振幅の増大に伴い減少しているが、スクラム時においても設計上の余裕を考慮して定めた安全限界最少限界出力比（SLMCPR）よりも大きく、余裕があることがわかる。この熱的余裕を定量的に評価するために、中性子束の振幅とCPRの関係をFig.30に示す。図より、スクラム時のCPRの余裕は約0.4、逆に、スクラムせずに振幅が更に増加したとすると、CPRがSLMCPRを下回るのは振幅が約240%以上になった時であることがわかる。以上の結果により、La Salle-2号炉炉心不安定振動時には中性子束の振幅がかなり増加したものの熱的余裕は十分あり、燃料健全性上問題はなかったことが示された。

5.2 振動の検出可能性と熱的余裕

不安定振動時のCPRの応答は、燃料バンドルの出力、流量の振動の振幅に大きく依存するが、BWRにおいては、LPRMにより局所的な中性子束応答は計測されているが、流量に関しては、炉心入口流量のみが計測され、局所的な燃料バンドルの流量は計測されていない。そこで、振動モードによってバンドル出力とCPRの応答間の関係がどのようにことなるかの検討を行なった。解析では前と同様に国内の代表的なBWRプラントデータを用いて、自然循環状態での出力増加や、ホットチャンネルの出力増加で炉心不安定振動と領域不安定振動が生じる状態を定め、各々の振動状態において、ホットチャンネルの出力と流量の振幅とCPRの初期値からの偏差分との関係を評価した。尚、流量としては、通常BTは炉心上部の燃料バンドル出口近傍で生じるため、バンドル出口流量の振幅で評価した。解析結果をFig. 31に示す。図より、出力の振幅が同じである場合には、振動モードが領域不安定であるほうがCPRの変化は大きく熱的余裕の面からは厳しくなることがわかる。流量に関しては、逆に、同一振幅に対して炉心不安定のほうがCPRの変化は大きくなる。この傾向の相違を直接的に検討するために、流量から出力への伝達関数のゲインを2つの振動モードで比較した。結果をFig. 32に示すが、炉心不安定のほうが領域不安定の場合よりゲインが大きくなっていることによりCPRの応答に差が生じていることが明かとなった。

以上は燃料バンドルの出力と流量とCPRの関係であったが、実機で測定されているのはLPRMであり、LPRMはその周囲の燃料バンドルの応答が反映される平均化されることになるため、ホットチャンネルの変動はなまらされることになる。そこで、振動の検出可能性を考慮に入れた熱的余裕評価の面からは、不安定振動発生と同時に振動が成長してCPRがMCPRを下回る様

なケースが最も厳しく、且つ、領域不安定のほうが同一振幅のLPRM変動に対してCPRの変化が大きいので、限界ケースとしてバラメータサーベイにより、領域不安定が発生する限界の状態でCPRがSLCPYを下回るような炉心状態を仮想的に作成して、LPRMの振幅と、CPRの変化との関係を求めた。結果を Fig. 33 に示す。図より、通常のノイズレベルを考慮しても、熱的余裕が確保されている状態でLPRMにより十分検出可能であることがわかった。また、このような限界ケースにおけるLPRM振幅の分布を Fig. 34 に示すが、大多数のLPRMにより領域振動が検出可能であることがわかる。

6. まとめ

La Salle-2 号炉で発生した炉心不安定による中性子束振動現象を契機として、BWRの安定性について関心が集まり、不安定振動、特に領域不安定振動の検出性や、燃料健全性に関する検討に対する要求が高まっている。本論文においては、3次元動特性解析コードTOSDYN-2を用いて領域不安定の発生条件について検討や、2つのモードの中性子束振動時における燃料健全性評価として熱的余裕評価を行なった結果についてまとめた。また、これらの解析結果の有効性を示すために、TOSDYN-2の解析モデルや実機データによる検証結果についてもまとめた。これらの検討から得られた結論を以下に述べる。

- TOSDYN-2コードは実機不安定現象を良く模擬でき、BWRの安定性解析に広く使用可能であることが明かとなった。
- 領域不安定現象発生には、径方向出力分布が大きな影響を及ぼし、ホットチャンネルが炉心周辺部に配置された周辺ピーク分布の場合に生じやすい。
- 核的フィードバック効果はチャンネル安定性に対して不安定化効果をもたらす。
- 2つの振動モードを比較すると、領域不安定のほうが炉心不安定よりも流量から出力への伝達関数のゲインが小さい（逆に出力から流量への伝達関数のゲインは大きい）ため出力の振幅が同じである場合には、振動モードが領域不安定であるほうがCPRの変化は大きく熱的余裕の面からは厳しい。
- 領域不安定が発生する限界の状態でCPRがSLCPYを下回るような限界ケースを仮定しても、振動は熱的余裕が確保されている状態で大多数のLPRMにより十分検出可能であることがわかった。

参考文献

- (1) USNRC, "AECD Concerns REgarding The Power Oscillation Event at La Salle-2 (BWR-5)", AEOD Special Reprot S803, (1988)
- (2) L. A. Carmichael and R. O. Niemi, "Transient and Stability Tests at Peach Bottom Atomic Power Station Unit 2 the End of Cycle 2", EPRI NP-564, (1978)
- (3) S. A. Sandoz and S. F. chan, "Vermon Yankee Stability Tests During Cycle 8", Tran. Am. Nucl. Soc.,

45, 754, (1983)

- (4) Y. Waaranpera and S. Andersson, "BWR Stability Testing : Reaching the Limit-Cycle Threshold at Natural Circulation", Trans. Am. Nucl. Soc., 39, 868, (1981)
- (5) E. Gialdi, S. Grifoni, C. Parmeggiani and C. Tricoli, "Core Stability in Operating BWR : Operational Experience", SMORN IV, Dijon, France, (1984)
- (6) D. Dayal, G. Preusche, L. P. Brandes, M. Frank, H. Heckmann, R. Seepolt and K. P. Wachter, "Stability Tests at KWU Nuclear Power Plants", ANS Topical Meeting, Atlanta, USA, (1987)
- (7) GE, "General Electric BWR Thermal Analysis Basis (GETAB) Data, Correlation and Design Application", NEDO-10958, Nov (1973)
- (8) R. B. Linford, "Analytical Methods of Plant Transient Evaluation for the General Electric Boiling Water Reactors", NEDO-10802, Feb (1973)
- (9) S. Tsunoyama, A. Tanabe and S. A. Sandoz, "Vermont Yankee Stability Simulation with a Three-Dimensional Transient Model", Tran. Am. Nucl. Soc., 46, 752, (1984)
- (10) Y. Takigawa, Y. Takeuchi, S. Tsunoyama, S. Ebata, K. C. Chan and C. Tricoli, "Caorso Limit Cycle Oscillation Analysis With Three-Dimensional Transient Code TOSDYN-2", Nucl. Tech., 79, 210, (1987)
- (11) Y. Takeuchi, S. Tsunoyama, Y. Takigawa, S. Kasai and H. Namba, "Fuel Integrity Evaluation at Instability Phenomenon", International Workshop on BWR Stability, Holtsville, USA, (1990)
- (12) Y. Takeuchi, Y. Takigawa, S. Tsunoyama and A. Kojima, "Analysis of Selected Rod Insertion Test in BWR plant with Three Dimensional Transient Code TOSDYN-2", J. of Nucl. Sci. and Technol., Vol. 28, No. 3, pp.199~207, March (1991)
- (13) J. March-Leuba and E. D. Blakeman, "A Mechanism for Out-of Phase Power Instabilities in Boiling Water Reactors", NSE, 107, 173-179 (1991)
- (14) O. Yokomizo, I. Sumida, T. Anegawa, Y. Yoshimoto and T. Fukabori, "Examination of Nuclear Thermal Hydraulic Oscillation Modes in BWR Core", Proc. of Int. Workshop on Boiling Water Reactor Stability, 5. 1, Holtsville, USA, (1990)

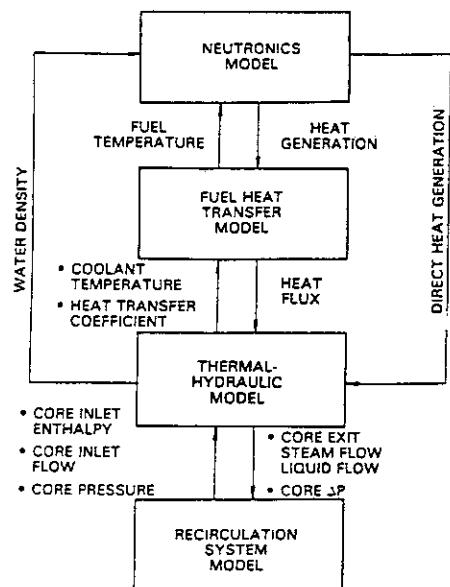


Fig. 1. TOSDYN-2 model.

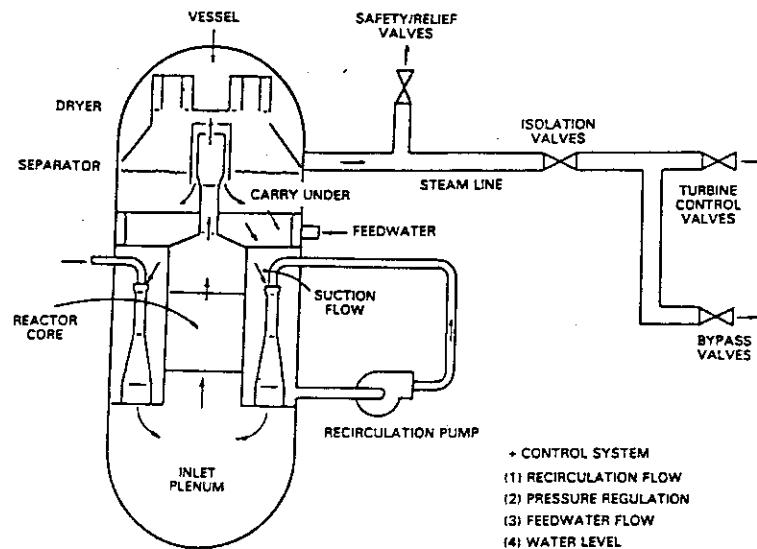


Fig. 2. TOSDYN-2 ex-core model.

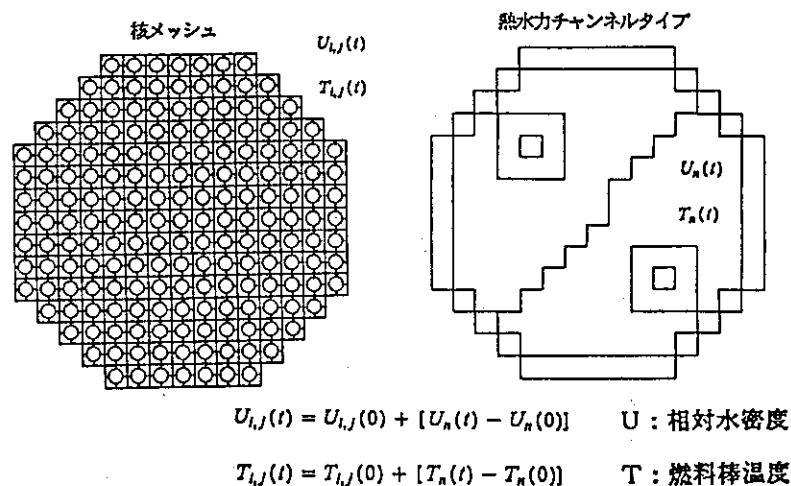


Fig. 3 Coupling between Neutronics Model and Thermal-Hydraulic Model

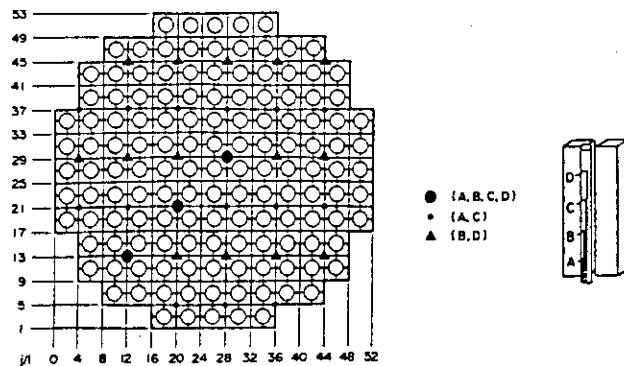


Fig. 4 Recorded LPRM positions.

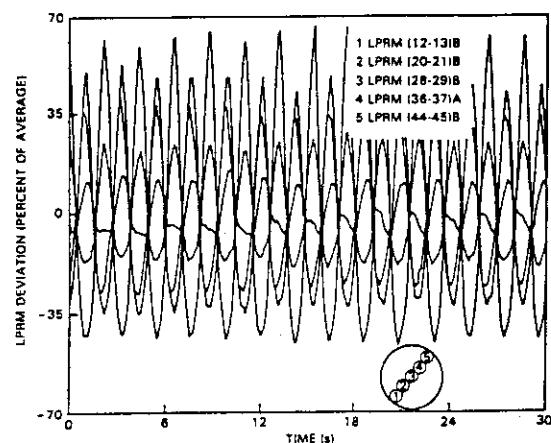


Fig. 5 LPRM responses (test data).

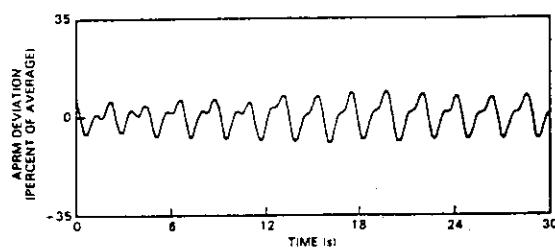


Fig. 6 APRM response (test data).

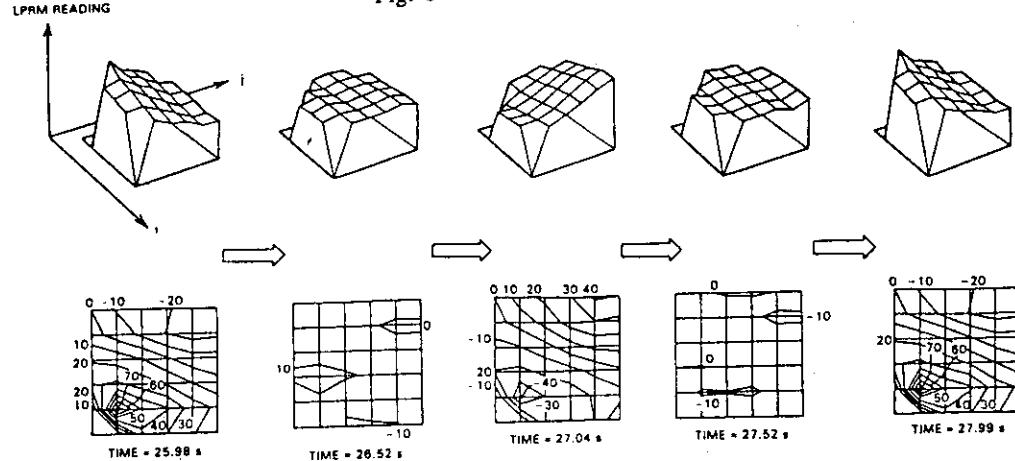


Fig. 7 Spatial change of the LPRM oscillations (test data).

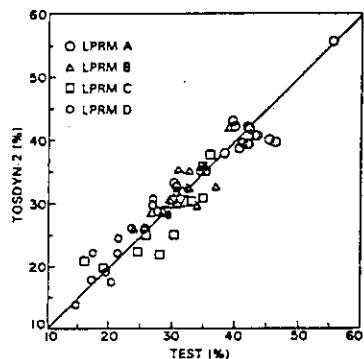


Fig. 8 LPRM initial values (test versus TOSDYN-2).

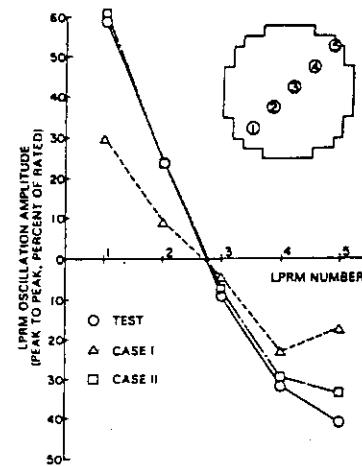


Fig. 15 Comparison of LPRM amplitude distribution on a diagonal line.

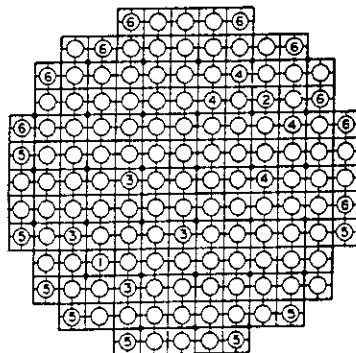


Fig. 9 Thermal-hydraulic channel types (case I: six channel types).

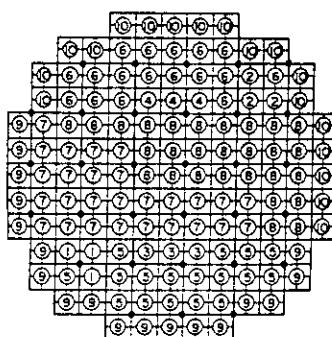


Fig. 12 Thermal-hydraulic channel types (case II: ten channel types).

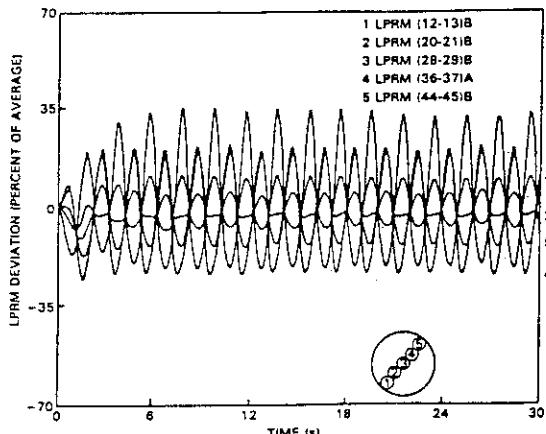


Fig. 10 LPRM responses (case I).

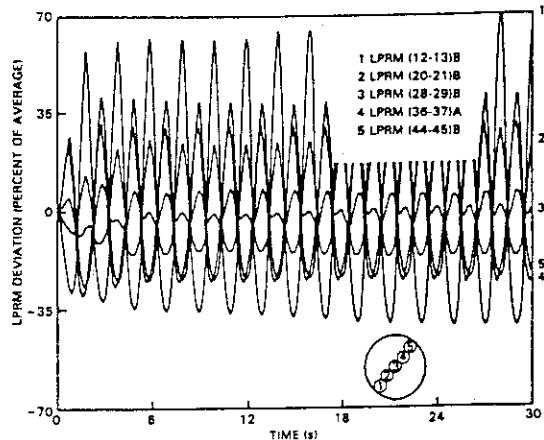


Fig. 13 LPRM responses (case II).

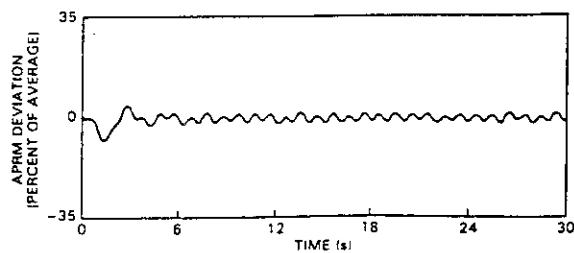


Fig. 11 APRM response (case I).

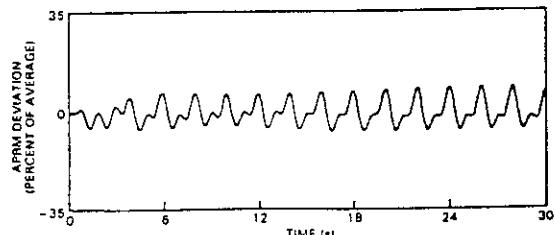


Fig. 14 APRM response (case II).

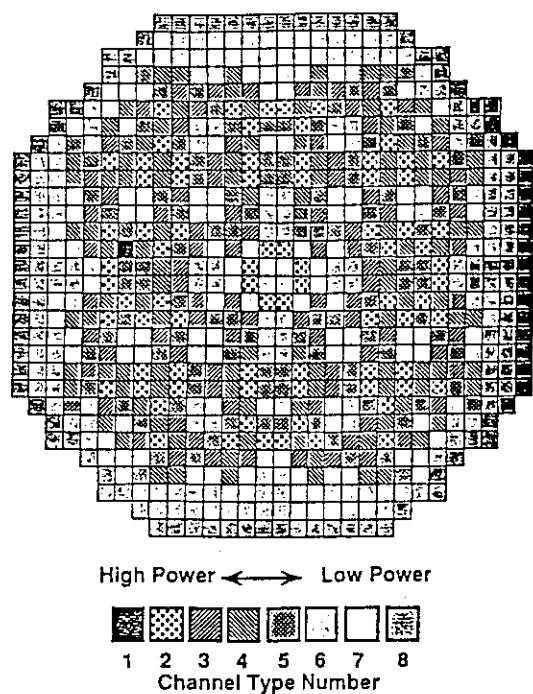


Fig. 17 Thermal-Hydraulics Channel Group Division for LaSalle-2 Analysis

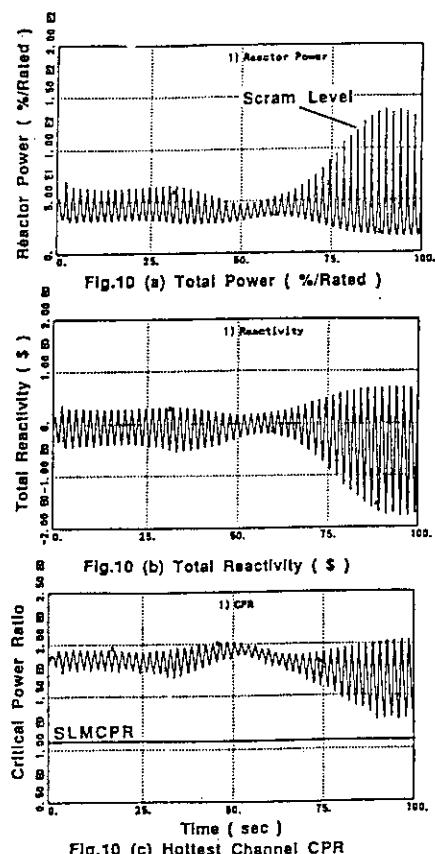


Fig. 18 LaSalle-2 Analysis with TOSDYN-2

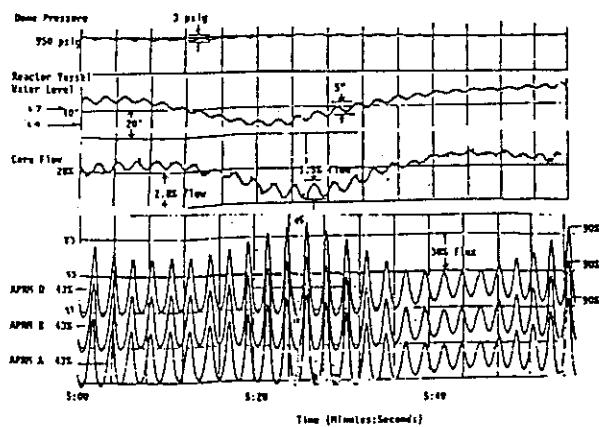


Fig. 16 La Salle-2 measured data

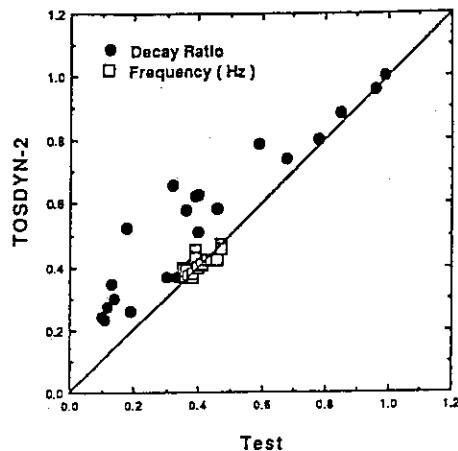


Fig. 19 TOSDYN-2 Qualification with BWR Stability Test Data

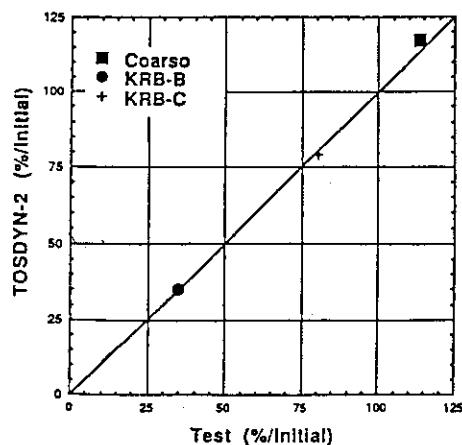


Fig. 20 TOSDYN-2 Qualification with Regional Oscillation Data (Largest LPRM Limit Cycle Amplitude)

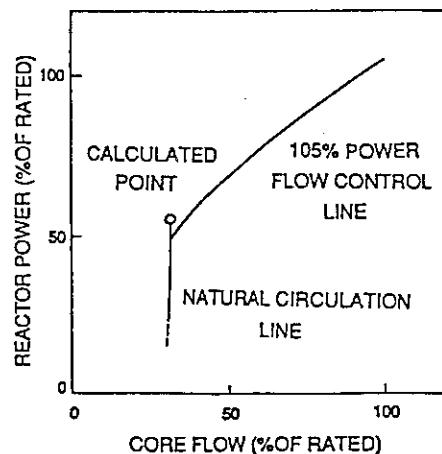


Fig. 21 Operating Point for Calculation

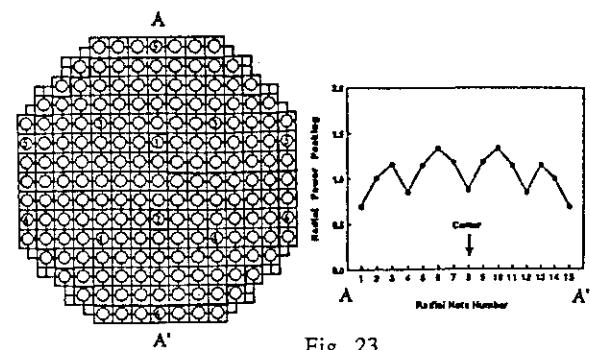


Fig. 22 TOSDYN-2 thermal-hydraulic channel types (Case 1 : base case) Radial power distribution (Case 1 : base case)

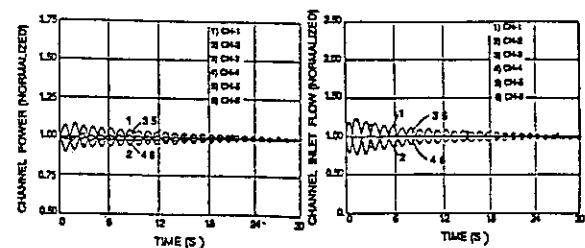


Fig. 24 TOSDYN-2 regional oscillation results (Case 1 : base case)

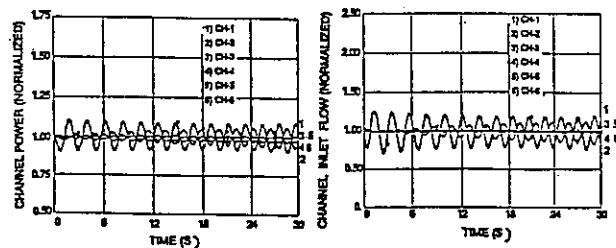


Fig. 28 TOSDYN-2 regional oscillation results (Case 3 : single perturbation to ch-1)

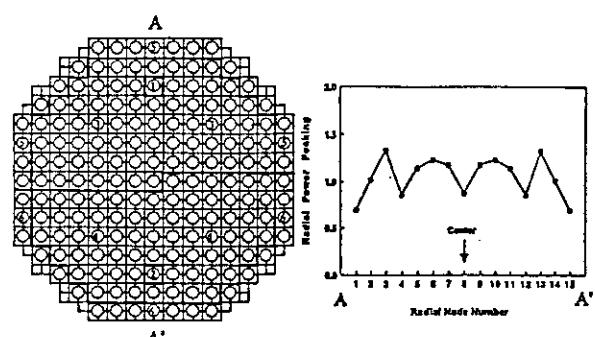


Fig. 25 TOSDYN-2 thermal-hydraulic channel types (Case 2) Radial power distribution (Case 2)

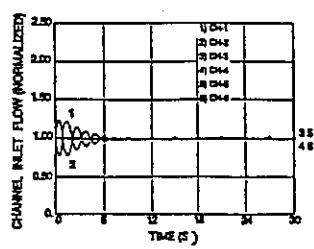


Fig. 29 TOSDYN-2 regional oscillation results (Case 4 : without neutronics)

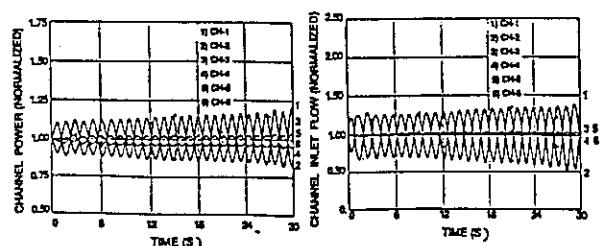


Fig. 27 TOSDYN-2 regional oscillation results (Case 2 : peripheral hottest channels)

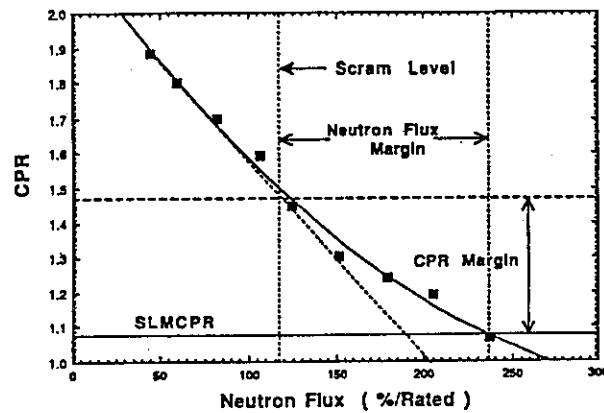


Fig. 30 Neutron Flux Peak vs. CPR Response at LaSalle-2 Analysis

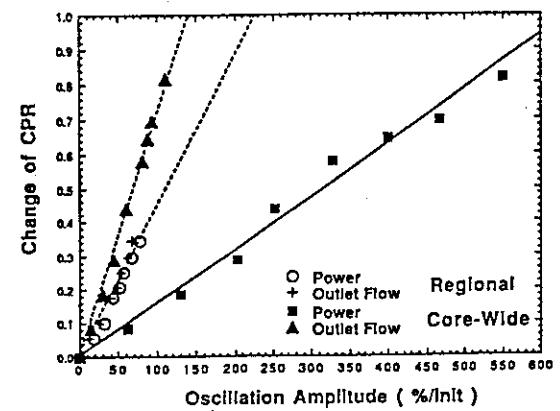


Fig. 31 Correlation between Δ CPR and Oscillation Amplitude

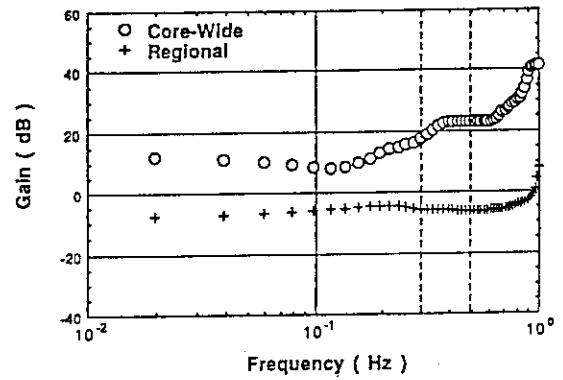


Fig. 32 Comparison of Flow-to-Power Gain

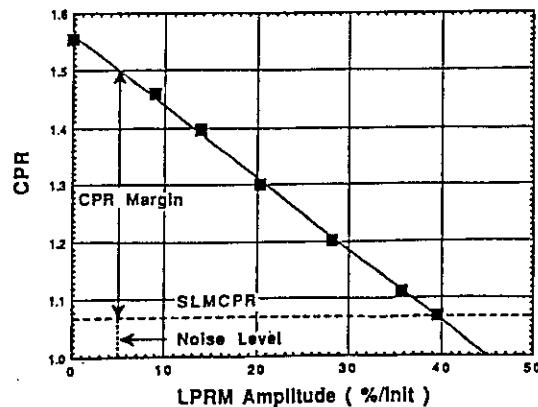


Fig. 33 Correlation between CPR and LPRM Amplitude

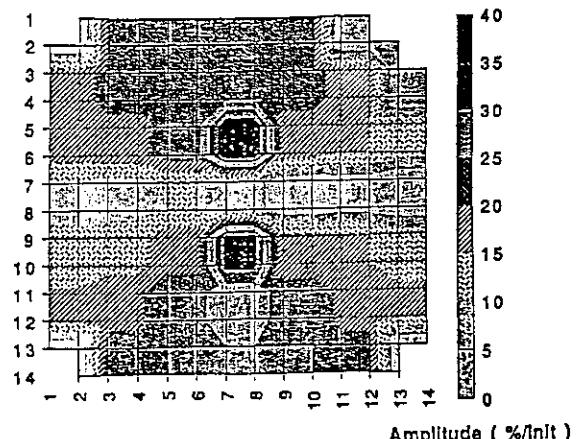


Fig. 34 LPRM Amplitude Distribution During Regional Oscillation

10. SIMMER-III コードと高速炉の空間依存動特性

近藤 悟, 石川 真 (動燃・大洗)

動燃事業団では、高速炉のシビアアクシデントに対する安全裕度を合理的に評価するために、次世代安全解析コード SIMMER-III の開発を進めている。本コードは炉心損傷事象推移を解析するために、多相多成分流体力学モデルを空間-エネルギー依存の動特性モデルと結合した総合コードである。核計算部については、空間依存性については TWOTRAN-II を基礎とした S_n 輸送理論を用い、時間依存性については改良型準静近似を採用するとともに、流体力学計算部との有機的な結合を図っている。本報では、安全評価の観点から重要となる炉物理上の課題のいくつかのハイライトを概括するとともに、SIMMER-III コードにおける核断面積の処理及び空間-エネルギー依存の動特性の取り扱いを紹介する。

SIMMER-III Code and Space-Energy Dependent Dynamics of Fast Reactors

Satoru KONDO and Makoto ISHIKAWA (PNC)

A next-generation safety analysis code SIMMER-III is being developed at the Power Reactor and Nuclear Fuel Development Corporation to realistically assess severe accidents in liquid-metal fast breeder reactors. To analyze core disruption sequences systematically, the code couples a multi-field multi-component fluid-dynamics model with a space- and energy-dependent reactor dynamics model. In the neutronics portion of the code, the space dependence is modeled by an S_n transport theory based on the TWOTRAN-II code, and the dynamics is treated by an improved quasi-static method. Coupling with the fluid-dynamics portion is carefully designed, because the core neutronic state is determined primarily from time-dependent mass and energy distributions of the core materials. In this presentation, some of the important topics on safety-related fast reactor physics will be highlighted, with emphasis on the neutronics model of SIMMER-III including the cross-section handling and space-dependent dynamics.

1. 序言

動力炉・核燃料開発事業団（動燃）では、高速増殖炉（以後単に「高速炉」と記述する。）のシビアアクシデントに対する安全裕度を合理的に評価するために、次世代安全解析コード SIMMER-III の開発を進めている^{(1)~(3)}。本コードは炉心損傷の事象推移を総合的に解析するために、多相多成分流体力学モデルを空間一エネルギー依存の動特性モデルと結合したシステムコードである。核計算部については、空間依存性については TWOTRAN を基礎とした S_n 輸送理論を用い、時間依存性については改良型準静近似を採用するとともに、流体力学計算部との有機的な結合を図っている。本コードの主要部分でかつ特徴的な部分は燃料ピンやラッパ管の構造材の挙動を含む流体力学計算部であり、核計算部は従来よりの比較的標準的な取り扱いとなっている。しかしながら、シビアアクシデントの安全評価における非常に速い過渡事象を対象とする場合、核計算部に具備すべき機能としては、通常の炉心設計又は制御において考慮すべき炉物理上の問題とは違った側面を合わせ持つことになり、本報ではこれらに着目した紹介を行う。

一般に、原子炉の動特性を考える場合、対象とする核的現象の時間スケールは以下の 3 種に大別される。

- (1) 炉心の燃焼に伴う組成の変化のような、数カ月オーダーに及ぶ長時間挙動
- (2) 原子炉の起動・停止、負荷変化、熱中性子炉で重要な FP の反応度効果等の原子炉の運転・制御で問題となる数時間～数日程度の中時間挙動
- (3) 事故解析や安全評価で重要となる ms ~ s オーダーの短時間挙動

ここで対象とするのは、(3) の短時間挙動に関する動特性の問題である。短時間の動特性は、kinetics 又は dynamics と呼ばれるが、最近では前者は反応度フィードバックを含まない動特性問題に限定して使われる例もある⁽⁹⁾。

高速炉は低圧系であるため、軽水炉における制御棒飛び出しのような反応度事故は無く、一般に異常状態（いわゆる設計基準事象）に対しての安全裕度は高い。したがって、一部の例外的な事象を除けば、ここで述べる動特性を問題にするのは、仮想的なシビアアクシデント条件下ということになる。この場合、高速炉の炉心は反応度が最大となるように構成されていないことから、冷却材のボイド化、炉心の損傷等の仮想的な物質の再配置が正の反応度フィードバック効果を持つ場合があり、これによる炉心の核的ポテンシャルとその機械的影響（エナジエティクス）が歴史的にみても安全評価上の重要な課題の一つとなっている⁽¹⁰⁾。

本報では、先ずこのようなシビアアクシデントの安全評価の観点から重要となる炉物理上の課題のいくつかのハイライトを概括し、動特性解析技術の最近の動向を復習する。次に、SIMMER-III コードの全体概要を説明し、同コードにおける核断面積の処理及び空間一エネルギー依存の動特性の取り扱いを紹介することにより、高速炉の動特性を議論する。

2. 高速炉安全評価における炉物理上の課題

2. 1 高速炉炉心の一般的核特性

本章では、高速炉の安全評価の上で重要となる炉物理上の課題について述べるが、ここでは先ず熱中性子炉である軽水炉とは異なる炉心の核的特性の特徴について簡単にまとめておく。図-1に典型的な高速炉の炉心構成例（「もんじゅ」等で採用された均質炉心及び米国CRBRで検討された径方向非均質炉心）を示す。

(1) 高速中性子炉

プルトニウムーウラン混合酸化物燃料の使用、冷却材にナトリウムの使用、高出力密度達成による炉心のコンパクト化等により、炉心の中性子スペクトルが硬い。図-2に100万kW級高速炉における典型的な中性子エネルギースペクトルを示す。エネルギー分布は100keV付近にピークを持ちながらに分布している。これにより、高速炉炉心は以下の核的特性を有することになる。

- ①即発中性子寿命 τ_1 が短い ($\sim 4 \times 10^{-7}$ s ; 軽水炉では $\sim 2 \times 10^{-5}$ s)。
 - ②遅発中性子割合 β が相対的に小さい ($\sim 4 \times 10^{-3}$)。
 - ③炉心の核的結合が強いため、炉心物質の再配置が生じない間は空間依存性が比較的小さい。
 - ④燃料ピン内の中性子束減衰、FPの毒作用、非均質効果等が熱中性子炉に比べてはるかに小さい。
 - ⑤中性子スペクトルが広いエネルギー領域に分布し、かつその空間依存性も大きい。
- 以上のうちで、①及び②の特性は、即発臨界に達するような反応度上昇が生じる場合に重要となるが、これは後述のような設計基準事象を大幅に超えた発生確率がきわめて低いシビアアクシデント条件においてのみ起こり得る。③及び④は高速炉における炉物理の取り扱いを容易にする要素であり、逆に⑤は多群中性子束（数～20群）の取り扱いが不可欠であることを意味する。

(2) 低圧系で単相ナトリウムの使用

熱輸送特性の優れたナトリウムを単相で使用するため、圧力変動等の外乱に対して炉心及び冷却系は安定である。高速炉の冷却系は一次低圧系で、ナトリウムの使用温度は沸点に対して十分な余裕があり（サブクール度300K以上）、かつ冷却材中にガス気泡が巻き込まれないような設計上の配慮がなされることなら、一般に、運転時の異常な過渡変化や事故のような異常状態に対する安全裕度はきわめて高いとされている。また、低圧系であることにより、軽水炉における制御棒の飛び出し（又は急速落下）のような反応度事故が存在しないことも重要な特徴である（制御棒の誤引き抜きに対する反応度挿入率は通常数c/s以下に抑えられている。）。

(3) 炉心配置及び核的ポテンシャル

高速炉の炉心形状は、ポンプ容量と炉心圧力損失の制限、出口冷却材温度の制限、後述するボイド反応度の制限等の理由により、特に大型炉心では軽水炉と比べて偏平となっている。さらにお出口冷却材温度を均一化するために非均質炉心設計の採用、炉心周辺部への高 plutoniウム富化度燃料の配置等の工夫がなされている。その結果、高速炉の炉心は反応度が最大となるように構成されていないため、冷却材のボイド化、炉心の損傷等の仮想的な物質の再配置が正の反応度フィードバック効果を持つ場合があり、これによる炉心の核的ポテンシャルとその機械的影响（エナジエティクス）がリスクの観点からの重要な課題となっている。

2. 2 反応度フィードバック特性

炉心及び関連する冷却系が、すべての運転範囲で急速な固有の負の反応度フィードバック特性を有するように設計されることは高速炉においても当然の要求である。核特性としては、燃料のドッpler効果に基づく負の反応度係数及び燃料膨張に基づく負の反応度係数等の固有の安全性が具備されている。

高速炉の安全性に關係する重要なフィードバック効果として冷却材ボイド反応度がある。良く知られているように、原型炉クラス又は大型の炉心においては、炉心中央部でボイド反応度が正となる領域があり、炉心のボイド化（冷却材の沸騰）がシビアアクシデントの初期における反応度効果、ひいては機械的エネルギー発生（エナジエティクス）の観点から重要となる。ボイド反応度の大きさは炉心のサイズと炉心設計に強く依存し、リスク低減の観点からは、通常の炉心設計の範囲内でその大きさを適切に（事象推移が autocatalytic とならない程度に）抑制すること、評価の不確かさ幅の低減を図ることなどが今後の重要な課題である。なお、ボイド反応度は重要なテーマではあるが、これが問題となるのは事故の初期の過程であって、これのみで高速炉のリスク（又はエナジエティクス）全体が支配されるわけではない。

また、高速炉の受動的安全機能を強化するために、炉心設計及び安全評価において以下の反応度効果に対する研究が近年広く進められており、炉心の損傷を緩和・抑制し得る機構として注目を集めている⁽¹¹⁾⁻⁽¹⁶⁾。

- (1) 冷却材温度上昇に対応した炉心の径方向膨張による負の反応度効果
- (2) 冷却材温度上昇に対応した制御棒駆動軸の軸方向膨張、又は燃料の温度上昇に伴う燃料ピンの膨張と制御棒との相対膨張による負の反応度効果
- (3) 炉心の内部転換比の向上による制御棒燃焼補正反応度の低下に伴う誤引き抜き時の反応度挿入の制限
- (4) 炉心上部へのナトリウムプレナム設置（ソ連型固有安全高速炉）による冷却材ボイド化時の中性子の漏れの増加による負の反応度効果

2. 3 再臨界を含む出力過渡の取り扱い

シビアアクシデントを含む安全評価の対象となる出力過渡事象としては、制御棒の誤引き抜きに相当する数 \$ / s 程度の緩やかな反応度挿入から再臨界で想定されるような数十 \$ / s に達する厳しい反応度挿入までの広範囲に及び、動特性の解析手法もこれらをカバーする必要がある。再臨界については、歴史的にはいわゆる「Bethe-Tait 事故」のように全炉心の同時コンパクションを仮想して数百 \$ / s の出力バーストを扱った時代もある⁽¹⁰⁾。しかし、事故の初期の過程を解析する SAS コード⁽¹⁷⁾⁻⁽¹⁸⁾、炉心損傷の過程を解析する SIMMER コード⁽¹⁹⁾⁻⁽²⁰⁾等の機構論的評価手法の開発及び事象推移の現象論的理解度の向上により、現在ではより合理的な安全評価が可能となってきた⁽²¹⁾⁻⁽²³⁾。

2. 4 熱流動計算との結合

前述のように、炉心損傷時の反応度変化は主として炉心物質の再配置により決定される。したがって、動特性の評価に当たっても、物質の大規模移動を含む炉心の熱流動計算との有機的な結合とそれぞれのタイムステップ制御の最適化を図ることが不可欠である。流体力学計算からは炉心物質の質量（核計算で扱う同位体毎の原子数密度に換算）及び温度を渡し、逆に核計算からは計算された原子炉出力に対応した流体力学成分毎の内部発熱密度が渡される。

また、ここで議論している動特性問題では、炉心物質の質量及び温度の空間分布が時間とともに常に変化する状況を扱うことになる。したがって、動特性解析においても、物質の組成と温度の変化に対応した自己遮蔽の計算と実効マクロ断面積の評価を核計算のタイムステップ毎に行う必要がある。なお、核断面積自体についても、多群の取り扱いに加えて、500 K程度までの高温に対するデータが必要となる。

2. 5 安全評価における要求精度と効率

シビアアクシデントの安全解析は、炉心損傷と炉心物質の移動挙動及びこれに伴う炉心の核的挙動からなる事象推移を対象とする。一般に、炉心物質の熱流力挙動の評価における不確かさは相対的に大きく、単純なバランス論からは核計算の精度を必ずしも高くする必要はないと考えられがちである。しかし、即発臨界出力バーストを含む動特性においては、反応度変化の予測精度が結果としてのエネルギー放出に与える影響は非常に大きい。したがって、利用可能な技術で、計算効率が実用上許容できる範囲で核計算の精度は確保しておく必要があると考えられる。具体的には次のような内容である。

- (1) 物質の移動が小さい事故の初期の過程を除けば、中性子束の空間依存性の取り扱いが不可欠である。
- (2) 炉心がコンパクションした体系までを考えると、中性子輸送効果を含む解析が必要である。拡散近似は中性子の漏れを過大評価し、反応度の予測が非保守的となる場合がある。一例として、FCA における燃料スランピング実験⁽²⁴⁾の解析結果を図

－3に示す。

- (3) 中性子スペクトルは広範囲に及び、かつその空間的・時間的变化も大きくなり得るため、エネルギー依存性の取り扱い（多群近似）が併せて必要となる。
- (4) 時間的に速い過渡変化を取り扱い、かつ中性子束分布の変化も大きいため、1点近似動特性は全く適用できず、また、後述の断熱近似による空間依存動特性も精度上不十分である。計算効率を併せて考慮すれば、改良型準静近似による動特性が最適である。
- (5) 実効断面積の計算は物質の組成と温度の変化を反映して動特性計算と連動して行う必要がある。また、ドップラー効果を適切に計算するために、いわゆる f -テーブルの内挿法についても配慮が必要である。図-4に従来用いられた線型内挿及び現在 SIMMER で採用している関数フィッティング型によるドップラー定数を比較して示す。

以上の核計算精度確保のための努力は計算効率の観点からは大きなデメリットを生じる可能性がある。核熱流力総合解析の核計算部で計算時間の大半を要するならば、これは実用上問題があることになる。したがって、第5章で述べるように、準静近似の採用と適切なタイムステップ制御、断面積計算の全面的見直し等のモデル上の工夫、さらにはベクトル化による計算の高速化によりこの問題を克服する必要がある。

3. 空間一エネルギー依存動特性の解析技術の進展

3. 1 歴史的経緯と概要

空間依存の動特性に対する解析技術は、1960年代後半から1970年代にかけて大きく進展を遂げた。多くの理論的研究と解析コードの開発が行われ、主として軽水炉の反応度事故を対象とした中性子パルス実験や反応度外乱に対する中性子束検出器応答実験等によりその検証作業が行われている。これまでに確立されている動特性解析手法の理論的詳細度については、①直接解法 (direct method)、②準静近似解法 (quasistatic method)、③断熱近似解法 (adiabatic method)、④一点近似動特性 (point kinetics) の4種に大別され、この順序で近似の度合が大きくなる。さらに、空間次元数や拡散か輸送かの違いにより細分化されている。一般に、詳細な手法ほど原理的に精度は良いが数値解法上の困難や計算時間上の負荷が大きくなることから、対象とする事象推移と要求される精度に応じて最適な手法を選択する必要がある。

これら解析技術の進展の歴史的経緯と理論的詳細については、Karl Ott の教科書⁽⁹⁾、伊勢の報告⁽²⁵⁾、Stacey⁽²⁶⁾ の論文に詳しいので参照されたい。以下の節では各手法の概要を説明し、最後に各手法の比較を行う。

3. 2 直接解法

直接解法は厳密解法とも呼ばれ、時間依存の中性子拡散方程式を、動特性解法上の

近似を交えずに直接差分化し、数値的に解く方法であり、精度の観点からは最も厳密であるとされている。解析コードとしては、WIGLE（1次元拡散）²⁷⁾、TWIGL（2次元拡散）²⁸⁾、DISCOTHEQUE（2次元拡散）²⁹⁾等が開発され、これらの実験的検証も行われている³⁰⁾。

これらのコードは手法としては一応確立され、近似解法の精度評価のための基準解析コードとして用いられる場合もあるが、計算時間及び数値解法上の問題により、1970年代以降のこの分野の研究の主流は以下の近似解法の開発に向けられた。

3. 3 準静近似解法

準静近似解法³¹⁾⁻³²⁾では、時間依存の中性子拡散（又は輸送）方程式を、時間のみに依存する振幅関数（amplitude function）と、主として空間に依存する形状関数（shape function）とに変数分離できるという仮定を置く。前者が原子炉出力に対応し、後者が初期値で規格化した中性子束分布に対応するが、中性子束分布の時間的变化が相対的に遅いということでこの分離が可能となり、近似解法の中では最も詳細な手法である。元々の準静近似ではある時間幅において形状関数が変化しないと仮定するが、線型外挿等によりその時間微分項を考慮するものを改良型準静近似と呼ぶ。解析コードとしては、最初にこの手法を確立した QX1（1次元拡散）³³⁾、これを拡張した FX2-TH（2次元拡散）³⁴⁾、VENUS³⁵⁾⁻³⁶⁾との結合図った FX2-VENUS-III³⁷⁾、本報で主に扱う SIMMER-II（2次元輸送又は拡散）³⁸⁾、3次元コードである SPARK（3次元拡散）³⁸⁾等が開発され、適用実績、検証例ともに多い。

高速炉のシビアアクシデント評価のためには、要求精度及び計算時間の両面から本手法が最適であり、2次元体系ではあるが、輸送理論を含む手法が SIMMER-II コードとして確立され実用に供されている。

3. 4 断熱近似解法

断熱近似は、前節の準静近似に加えて、形状関数の時間変化に対する振幅関数の時間微分項の寄与及び即発中性子と遅発中性子の区別を無視した近似解法である。この解法の考え方とは、中性子束分布の変化をあらかじめ静的計算で求めておき、振幅関数の計算（一点近似の動特性）と組み合わせることにより炉心内の出力分布の時間変化を計算するというものである。解析コードとしては軽水炉の反応度事故解析用の APEX（2次元拡散）³⁹⁾、高速炉用に CITATION（3次元拡散）⁴⁰⁾と EUREKA（一点近似動特性）⁴¹⁾を組み合わせたシステム等があり、実用に供されている。

この手法は定性的には出力上昇を過大評価する傾向があるが、簡便で計算時間の負荷も小さいことから、動特性の簡易評価の有効な手段となる。また、高速炉における制御棒の誤引き抜きのような過渡変化の比較的緩やかな動特性問題に対しては精度的にも十分な適用性を有すると考えられている。

3. 5 一点近似動特性

炉心動特性の空間依存性を全く考慮しない近似であり、初期の中性子束がそのままの空間分布を維持すると仮定して出力の時間変化を計算する。静的な核計算（拡散又は輸送）と摂動計算で出力分布及び反応度係数・物質価値分布を評価しておき、熱流力計算の結果を基に反応度変化を計算し、一点近似動特性方程式を解いて出力を計算する。代表的解析コードとしては、高速炉の安全評価で利用される HARHO-IN⁽¹⁸⁾、SAS⁽¹⁹⁾、水炉の安全評価で利用される EUREKA⁽⁴¹⁾などがある。

一般に、一点近似動特性は出力上昇を過小評価（すなわち非保守的）する場合があるため、空間依存性の効果が大きい場合には不適切であるが、簡易評価の道具として、又は事故の初期の過程のように物質移動が小さく中性子束分布のひずみが十分許容できるような事象に対しては有効で、現在でも広く利用されている。

3. 6 各動特性解析手法の比較

上記4種の動特性解析手法の特徴を表-1に要約する。手法の比較検討の例として、大型高速炉における制御棒誤引き抜きに対する各モデルの概念的比較を形状関数のひずみ及び遅発中性子の空間分布の変化の取り扱いに着目して図-5に示す。直接解法と準静近似との間には基本的違いはなく、また、断熱近似も中性子束のひずみについては適切に評価することが可能である。

定量的な比較の例として、Ottらが行った熱中性子炉に約100\$ / sの反応度投入を行った場合の反応度変化の解析結果を図-6に示す⁽³²⁾。このような非常に厳しい条件においても準静近似及び断熱近似は直接解法と非常に良く対応している。断熱近似は一般に反応度上昇を過大評価する傾向にあるが、これは遅発中性子先行核の空間分布の変化の時間遅れによる中性子束分布の変化の抑制を無視していることなどによるものである。

4. SIMMER-III の概要

本章では、動燃で開発を進めている次世代安全解析コード SIMMER-III の概要を記述する^(1)-⑧)。本コードは高速炉の炉心損傷事象推移における炉心の核熱流力挙動の総合的解析の評価精度の向上と適用範囲の拡大を目的としている。高速炉の損傷炉心では固気液の炉心物質の流動と燃料ピン及びラッパ管の挙動、及びそれに伴う炉心の核的挙動が相互に影響を及ぼし合う状況となっており、SIMMER-III では図-7に示すように、これらの計算を有機的に結合（相互にデータ交換）し、それぞれ最適な時間ステップを使用している。SIMMER-III は R-Z 2 次元体系を計算対象とし（図-8）、コード全体は以下の主要な要素モデルから構成されている。

- (1) 中核となる流体力学部は、炉心物質の固気液多相流を解析するために必要となる27密度成分、16エネルギー成分の保存式を扱う。また、燃料とスティールの液体間

の相対運動が事象進展に大きな影響を与えるため、液体2種と蒸気から成る3速度場の運動量保存式を採用した。

- (2) 流体力学の数値解法アルゴリズムは AFDM⁽⁴²⁾ で開発した4ステップ法を採用した。4ステップ法は多成分間の熱及び質量移行を効率良く扱えるように、セル内の移行とセル間の対流を分離したことに特徴がある。空間差分法としては、1次のドナーセル差分又は高次差分が使用できる。
 - (3) 各成分間の熱、質量及び運動量交換は、エネルギー成分間で定義される42種の接触界面積に基づいて計算される。熱及び質量移行モデルでは各境界で重要となる蒸発／凝縮、溶融／固化等の相変化プロセスを抽出しモデル化した。
 - (4) 流体力学アルゴリズムと熱及び質量移行モデルとの整合性を確保し、数値的な精度、安定性及び効率の観点で改善を図った関数形式の状態方程式モデルを開発した。
 - (5) 事故の初期過程との整合性及び連続性を確保するため、径方向熱伝導、ピン内燃料移動、燃料ピン破損・崩壊等をモデル化することとした。
 - (6) 核計算部では大規模な物質移動の影響を扱う必要があるため、空間依存性については輸送理論、時間依存性については準静近似を用いる。また、未臨界時の炉心拳動を解析するための崩壊熱モデルを組み込む。核計算部のモデル及び手法についてはすでに SIMMER-II の段階でほぼ十分な完成度に達している。
 - (7) コード設計において、メモリーの節減とベクトル化を考慮して最適化を図る。
- 以上により、損傷炉心の核熱流力拳動解析の精度の向上、事故の初期過程からの整合性の取れた事象推移評価を可能とするためのモデルの詳細化、及び実用的なコードとするための計算効率向上の各観点の検討を併せて、平成元年より5年計画（平成2年までは日米共同開発）で鋭意開発を進めている。

5. SIMMER コードの核計算モデル

5. 1 断面積計算モデル

SIMMER では、高速炉用の 70 群の標準核データライブラリである JFS-3-J2 (JAERI Fast Set)⁽⁴³⁾ を、安全解析用核計算システム NEUSAPE⁽⁴⁴⁾ により少数群（通常数～十群）に縮約し使用する。SIMMER で扱う動特性問題では、計算セルの組成と温度が時間とともに変化するので、無限希釈のミクロ断面積及び遮蔽因子を入力し（米国標準の CCCC フォーマット⁽⁴⁵⁾ の ISOTXS 及び BRKOKS）、自己遮蔽の計算と実効マクロ断面積の生成はコードの内部で行う。したがって、縮約する際にも、実効ミクロ断面積を生成してから行う通常の手法ではなく、無限希釈のミクロ断面積及び遮蔽因子のそれぞれに対して縮約を行う必要がある。断面積計算は、後述する反応度計算ステップ毎に必要となり、かつ高速炉の解析では 20～30 の核種を対象とすることから、この計算には膨大な演算量が必要となる。

この問題に対処するため、新たに断面積前処理コード MXS を開発し、以下に示すいくつかの新概念及び手法を導入した⁽⁴⁶⁾。

- (1) 「物質 (material) 断面積」の概念導入：SIMMER で扱う 5 種類の基本的炉心物質 (fertile fuel、fissile fuel、steel、sodium、control material) の質量は変化するが同位体組成は変化しないため、物質毎に断面積を定義する（前処理で同位体断面積の混合を計算する。）ことで演算量を低減する。
- (2) 自己遮蔽計算をテーブル内挿型から関数近似型に改良：遮蔽因子 (f - テーブル) のバックグラウンド及び温度領域毎に関数形（バックグラウンドは 15 種、温度は 4 種の中から選択）を前処理であらかじめ求めておく。これにより精度・効率を改善する。
- (3) ベクトル化
前処理を含めた断面積処理の流れを図-9 に示す。

5. 2 動特性計算（準静近似）モデル

準静近似では、時間一空間依存の中性子束 $\Phi(t, r, z, E)$ を、時間依存性の強いスカラーの振幅関数 $N(t)$ と空間依存性は強いが時間変化は遅い形状関数 $\phi((t), r, z, E)$ とに変数分離する。

$$\Phi(t, r, z, E) = N(t)\phi((t), r, z, E)$$

準静近似の利点は、計算時間のかかる形状関数の計算を大きな時間間隔（中性子束計算ステップ）でしかも通常の定常計算により行うことでの計算効率の大幅な改善ができることがある。これを解く際には、初期における随伴中性子束 $\phi_{ij,z}^*$ を重み関数として使用し、次の拘束条件を満たすようにする。

$$\sum_{ij} V_i \sum_{\epsilon=1}^{EM} \frac{\dot{\phi}_{ij,z}(0, r, z, E) \phi_{ij,z}^*((t), r, z, E)}{V_z} = 1, \text{ IGM : 中性子エネルギー群数}$$

形状関数の計算手法は静的な核計算と同様であり、次節で述べる。ただし、過渡時の中性子束の計算は前の中性子束ステップの値を初期値として計算するため、収束は比較的速い。

SIMMER の改良型準静近似では、形状関数の時間微分項は線型であると仮定し直線外挿しながら振幅関数を計算する（反応度ステップ）。振幅関数は、輸送方程式に随伴中性子束をかけて、空間メッシュとエネルギー群に対して和をとることにより、結局、次の一点近似動特性の式に変形できる。

$$\begin{aligned} \frac{dN}{dt} &= \frac{\rho - \beta}{\Lambda} N + \sum_{d=1}^6 \lambda_d c_d \\ \frac{dc_d}{dt} &= -\lambda_d c_d + \beta_d \frac{N}{\Lambda}, \quad d = 1, 2, \dots, 6 \end{aligned}$$

ここで、反応度、遅発中性子割合、中性子生成時間、遅発中性子先行核濃度等の、いわゆる動特性パラメータは、前の中性子束計算ステップから外挿された形状関数、初期の随伴中性子束、流体力学で計算された物質の質量及びエネルギーを反映して計算された実効マクロ断面積の関数として記述される。反応度ステップで計算された振幅は 2 次関数でフィッティングし、流体力学計算で使用する炉出力として外挿する。準

静近似における処理の流れを図-10に示す。

高速炉のシビアアクシデントの初期の過程は、通常 SAS のような燃料ピン挙動（破損及び破損誤挙動を含む。）をより詳細に取り扱うため、SIMMER の初期条件は必ずしも定格運転状態ではない。すなわち、初期条件としての炉心の反応度がゼロでない状態からの動特性を取り扱うことになる。そこで、SIMMER では過渡的初期条件を設定するために、反応度に「げた」をはかせるなどの特殊な初期化が必要となる。

なお、上記の中性子束を線型に外挿する改良型準静近似に加えて、「中性子束更新法（flux-shape update method）」という新たな改良型手法もオプションとして開発されている⁽⁴⁷⁾。これは、反応度ステップ毎に中性子束を近似的に更新する手法で、具体的には中性子束計算の外部反復（outer iteration）を 1 ~ 2 回（完全に収束するまでは計算しない。）行うことにより、中性子束の非線型な変化を近似的に反応度計算に反映するものである。計算時間はかなり増加することになるが、燃料の急速な移動により中性子束の変化が非常に速いような場合には有効である。

5. 3 中性子束分布の計算モデル

SIMMER の中性子束（形状関数）の計算は、世界的に広く利用されている輸送コード TWOTRAN-II⁽⁴⁸⁾に基づいている。反復計算（outer/inner iterations）、リバランス法、加速法、負の中性子束の補正法等の解法に至るまでほぼ同様である。SIMMER では、初期化の中で定常計算を行い固有値 (k_{eff}) を計算する機能もあるので、臨界計算に適用することも可能であり、TWOTRAN-II とのベンチマークでもほぼ完全な一致を得ている。ただし、SIMMER の計算対象は高速炉であるために、以下のいくつかの点で簡易化等を図っている。

- (1) 計算体系は 2 次元 R-Z 体系のみで、境界条件も真空に限られる。
- (2) 上方散乱は高速炉では重要でないため、取り扱わない。
- (3) 非等方散乱については P₁ 近似相当（BHS 近似⁽⁴⁹⁾）までしか取り扱えない。
- (4) ダイヤモンド差分で生じる負の中性子束の補正（fix-up）の外に、重み付けダイヤモンド差分法も利用できる。
- (5) 内部反復計算のベクトル化

モデル及び手法としては、SIMMER-II の段階で 1980 年頃までに確立したものを踏襲しているが、プログラム的には SIMMER-III の他の部分との整合を取って全面的に改訂している。なお、輸送方程式の外に拡散方程式を解くオプションも用意されているが、現在は使用されていない。

5. 4 タイムステップ制御

SIMMER におけるタイムステップ制御は基本的にステップ幅の短い順に、①流体力学ステップ（fluid-dynamics step）、②反応度ステップ（reactivity step）、③中性子束ステップ（shape step）、の 3 種で構成され、図-11 に示すような階層構造を持っている

る。準静近似を精度及び効率の両面で実用的なものとするためには、②及び③のタイムステップ制御を適切に行うことが重要である。

SIMMER における中性子束ステップは、(1) 中性子の漏れの変化量、(2) 反応度ステップ数、(3) 中性子束の分布形状の変化 (tilt) 、(4) 反応度変化、等により制御する。また、反応度ステップは、(1) 反応度の変化、(2) 中性子束ステップ、(3) 振幅関数 (炉出力) の変化、(4) 逆時間 (inverse period) の変化、(5) 物質の質量又はエネルギーの変化、等により制御する。以上の各制御因子を評価する際に、反応度のレベルに応じて制御の厳しさを変化させるという特殊な取り扱いを行う。すなわち、未臨界時には制御を甘くし、臨界近傍では厳しくし、即発臨界近傍ではさらに厳しくするという考え方である。SIMMER では、QX1 コードのモデル化と経験を発展させて、図 - 12 に示すような 2 種のタイムステップ低減因子を使用している。

5. 5 流体力学との結合

前節のタイムステップ制御で述べたように、流体力学の計算は核計算よりも小さいタイムステップで進められるが、反応度ステップで計算された振幅関数 (炉出力) を 2 次関数でフィッティングし、以後の流体力学タイムステップに内部発熱密度として外挿する。次の反応度ステップでは流体力学で計算された新たな物質の密度と温度を基に振幅関数を更新する。これと外挿された振幅関数との間には誤差が生じることになるが、流体力学計算との間で反復を行うことは現実的でないため、この誤差はエネルギー (核発热量) を保存させるように次の流体力学計算で振幅関数を外挿する際に調整することとしている。反応度ステップでは直前の中性子束ステップから線型外挿された中性子束分布を使用し、次に中性子束を計算する際にも同じようなエネルギーの誤差が生じ得るが、これについても同様に調整する。

両者を結合する際に渡される情報としては、流体力学計算からは炉心物質の密度成分毎の質量 (核計算で扱う同位体又は物質毎の原子数密度に換算) 及びエネルギー成分毎の温度 (核計算で扱う同位体又は物質毎の温度に平均) を渡し、逆に核計算からは計算された原子炉出力に対応した流体力学成分毎の内部発熱密度が渡される。これにより、タイムステップ制御と併せて両者の有機的な結合が行われる。

6. 結 言

本稿では、高速炉のシビアアクシデントの安全評価の観点から重要な炉物理上の課題のいくつかのハイライトを概括し、動特性の解析手法の進展について簡単に復習した。シビアアクシデントを含む安全評価の立場からは、改良型準静近似解法の使用が要求精度及び計算効率の両面から最適であり、熱流動計算との統合を図った SIMMER コードとしてすでに実用化されている。今後、熱流動計算部の全面的改良を図った SIMMER-III コードに統合することにより、次世代安全解析コードシステムを構築する計画である。

なお、SIMMER コードの核計算モデルの開発については、主担当であった米国ロスアラモス国立研究所の Dr. L. B. Luck の貢献がきわめて大きい。同氏の約 10 年に及ぶ研究努力に対し謝意を表したい。また、(株)三菱総合研究所の角田弘和、立石英一郎の両氏には SIMMER-III コードの核計算部のプログラム整備に際して多大なる御協力を頂いた。

参考文献

1. 近藤、他、”高速炉の核熱流力安全解析コード SIMMER-III の開発（1）—コード構成及び基本アルゴリズム—” 日本原子力学会、1991年春の年会要旨集、F 35、1991年3月。
2. 飛田、他、” 同 （2）—プール流流動様式及び運動量交換関数モデル—”、同上、F 36。
3. 守田、他、” 同 （3）—多成分状態方程式モデル—”、同上、F 37。
4. 白川、他、” 同 （4）—コード構成及び基本アルゴリズム—” 日本原子力学会、1991年秋の大会要旨集、D 27、1991年10月。
5. 飛田、他、” 同 （5）—チャンネル流流動様式及び運動量交換関数モデル—”、同上、D 28。
6. 守田、他、” 同 （6）—多成分溶融／固化モデル—”、同上、D 29。
7. Y. Tobita, et al., "Interfacial Area Modeling for a Multiphase, Multicomponent Fluid-Dynamics Code," Proc. Int. Conf. on Multiphase Flows, Tsukuba, Japan, Sept. 24-27, 1991.
8. K. Morita, et al., "Improved Modeling of Multicomponent Equations of State for a Multiphase Flow Code SIMMER-III," ibid.
9. K. O. Ott and R. J. Neuhold, "Introductory Nuclear Reactor Dynamics," American Nuclear Society, (1985).
10. A. E. Walter and A. B. Reynolds, "Fast Breeder Reactors," Pergamon Press, (1981).
11. J. Cahalan, et al., "Performance of Metal and Oxide Fuels During Accidents in a Large Liquid Metal Cooled Reactor," Proc. 1990 Int. Fast Reactor Safety Mtg., Snowbird, Utah, U.S.A., August 12-16, 1990.
12. P. Royl, et al., "Influence of Metal and Oxide Fuel Behavior on the ULOF Accident in 3500 MWth Heterogeneous LMR Core and Comparison with Other Large Cores," ibid.
13. 西村、他、” 金属燃料炉心の受動的炉停止能力の評価” 日本原子力学会、1989年秋の分科会要旨集、C 11、1989年10月。
14. 西村、他、” 金属燃料炉心の ATWS 応答の評価” 日本原子力学会、1990年年会要旨集、D 28、1990年4月。
15. 田中、他、” FBR 金属燃料サイクル技術、—その魅力と実現性—、Part III、金属燃料炉心設計検討と安全性評価の現状” 原子力工業、第36巻、第6号 (1990)。

16. V. I. Matveev, et al., "Physical Grounds for Further Improvement of Fast Sodium Power Reactor Safety," Proc. 1990 Int. Fast Reactor Safety Mtg., Snowbird, Utah, U.S.A., August 12-16, 1990.
17. M. G. Stevenson, et al., "Current Status and Experimental Basis of the SAS LMFBR Accident Analysis Code," Proc. Fast Reactor Safety Mtg., Beverly Hills, California, U.S.A., April 2-4, 1974.
18. 中井、他、"高速増殖炉の安全解析に用いる計算コードについて" PNC N241 81-02 (1981年11月).
19. L. L. Smith, et al., "SIMMER-II: A Computer Program for LMFBR Disrupted Core Analysis," LA-7515-M, NUREG/CR-0453 (October, 1978) Rev. 1980.
20. 近藤、"SIMMER コードと高速炉安全解析" 第2回原子力におけるソフトウェア開発研究会報告集 J A E R I - M 86-009 (1986年3月)
21. N. Nonaka, et al., "Improvement of Evaluation Methods for Initiating-Phase Energetics Based on CABRI-1 In-Pile Experiments," Proc. 1990 Int. Fast Reactor Safety Mtg., Snowbird, Utah, U.S.A., August 12-16, 1990.
22. C. R. Bell, et al., "Advances in the Mechanistic Assessment of Post-disassembly Energetics," Proc. Int. Mtg. on Fast Reactor Safety Technology, Seattle, Washington, U.S.A.,
23. Sa. Kondo, et al., "SIMMER-II Application and Validation Studies in Japan for Energetics Accommodation of Severe LMFBR Accidents," Int. Mtg. on Fast Reactor Safety, Knoxville, Tennessee, U.S.A., April, 1985.
24. M. Nakano, et al., "An Experimental Study of Reactivity Changes and Flux Distortion in Simulated LMFBR Meltdown Cores," Nucl. Sci. Eng., Vol.87, p.283 (1984).
25. 伊勢、"1点近似法・準静的近似法の数値解法アルゴリズムとベンチマークテストの現状" J A E R I - M 6853 (1976年12月).
26. W. M. Stacey, Jr., "Space- and Energy-Dependent Neutronics in Reactor Transient Analysis," Reactor Technology, Vol.14, No.2, 1971.
27. W. R. Cadwell, et al., "WIGLE - A Program for the Solution of the Two-Group Space-Time Diffusion Equations in Slab Geometry," WARD-TM-416, Westinghouse Electric Co. (1968).
28. J. B. Yasinski, et al., "TWIGL - A Program to Solve the Two-Dimensional, Two-Group, Space-Time Neutron Diffusion Equations with Temperature Feedback," WARD-TM-743, Westinghouse Electric Co. (1972).
29. J. W. Stewart, "Transient DISCOTHEQUE Calculation - Module DISC," The JOSHUA System, DPSTM-500, Vol.7, Savannah River Laboratory (1972).
30. H. H. Hassan and G. H. Miley, "Verification of Space-Time Reactor-Kinetics Analysis by Experiments," Nucl. Sci. Eng., Vol.40, p.449 (1970).
31. K. O. Ott, "Quasistatic Treatment of Spatial Phenomena in Reactor Dynamics," Nucl. Sci. Eng., Vol.26, p.563 (1966).
32. K. O. Ott and D. A. Meneley, "Accuracy of the Quasistatic Treatment of Spatial Reactor Kinetics," Nucl. Sci. Eng., Vol.36, p.402 (1969).

33. D. A. Meneley, et al., "Fast-Reactor Kinetics -- The QX1 Code," ANL-7769 (March 1971).
34. R. A. Shober, et al., "FX2-TH: A Two-Dimensional Nuclear Reactor Kinetics Code with Thermal-Hydraulic Feedback," ANL-78-97 (October 1978).
35. J. F. Jackson and R. B. Nicholson, "VENUS-II: An LMFBR Disassembly Program," ANL-7951 (September 1972).
36. D. P. Weber, "VENUS-III: An Eulerian Disassembly Code," Trans. Am. Nucl. Soc., Vol.21, p.219 (1975).
37. D. P. Weber and B. D. Ganapol, "Multi-Field Hydrodynamic Disassembly Calculations with Space-Time Kinetics: FX2/VENUS-III," Proc. Int. Mtg. on Fast Reactor Safety and Related Physics, Chicago, Illinois, U.S.A., October, 1976.
38. J. K. Fletcher, "SPARK: A Computer Program to Solve the Time Dependent Multigroup Neutron Diffusion Equations in One, Two or Three Dimensions," TRG Report 2707 (October 1975).
39. R. C. Stirn, et al., "Rod Drop Accident for Large Boiling Water Reactors," NEDO-10527, General Electric Co., (1972).
40. T. B. Fowler, et al., "Nuclear Reactor Core Analysis Code: CITATION," ORNL-TM-2496, Rev. 2, (July 1971).
41. 石川、他、" 軽水動力炉の動特性解析コード EUREKA" JAERI 123
5 (1974年9月) .
42. W. R. Bohl, et al., "AFDM: An Advanced Fluid-Dynamics Model," LA-11692-MS (September 1990).
43. H. Takano, et al., "Production and Benchmark Tests of Fast Reactor Group Constant Set JFS-3-J2," JAERI-M 82-135 (October 1982).
44. 石川 真、他、" FBR 安全解析のための標準核計算システム (NEUSAPE) の開発" 日本原子力学会、1986年年会要旨集、D 27、1985年3月。
45. R. D. O'Dell, "Standard Interface Files and Procedures for Reactor Physics Codes, Version IV," LA-6941-MS (September 1977).
46. F. Parker, et al., "MXS Cross-Section Preprocessor User's Manual," LA-10856-M, NUREG/CR-4765 (March 1987).
47. L. B. Luck, "An Improved Quasistatic Method for Transition-Phase Calculations," Trans. Am. Nucl. Soc., Vol.41, p.317 (June 1982).
48. K. D. Lathrop and F. W. Brinkley, "TWOTRAN-II: An Interfaced, Exportable Version of the TWOTRAN Code for Two-Dimensional Transport," LA-4848-MS (July 1973).
49. G. I. Bell, et al., "Multiple Treatments of Anisotropic Scattering in S_n Multigroup Transport Calculations," Nucl. Sci. Eng., Vol.28, p.376 (1967).

表-1 空間依存動特性解析手法の定性的比較

解析手法の比較項目	空間依存動特性の解法				
	直接解法	改良準静	準静近似	断熱近似	一点近似
初期中性子束	Yes	Yes	Yes	Yes	Yes
中性子束のひずみ	Yes	Yes	Yes	Yes	No
フイードバック反応度の時間変化の空間依存性への反映	Yes	Yes	Yes	No ⁽²⁾	No
形状関数算出における遅発中性子と即発中性子の区別	Yes	Yes	Yes	No	No
形状関数算出における振幅関数の時間微分項の取り扱い	Yes	Yes	Yes	No	No
形状関数の時間依存性の取り扱い、	Yes ⁽¹⁾	No	No	No	No

注：(1) 通常は線型近似、(2) 原理的には可能だが通常取り扱わない

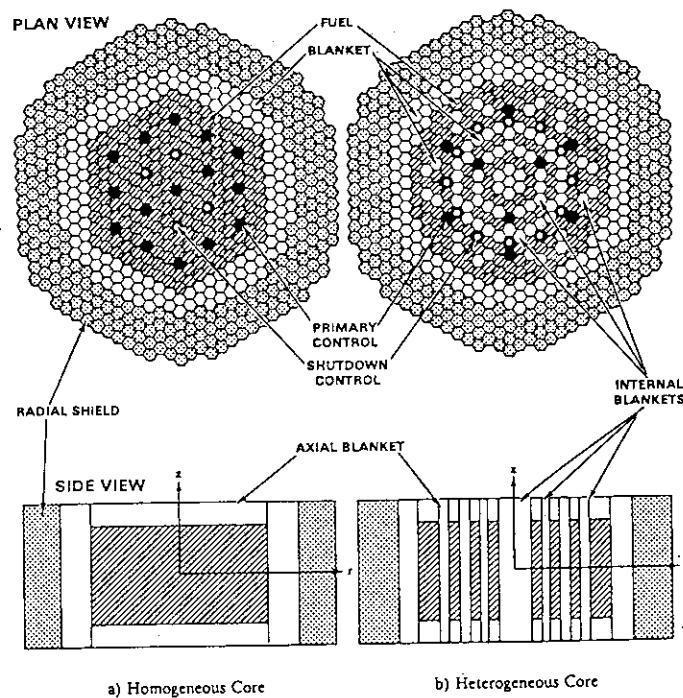


図-1 典型的な高速炉の炉心設計例（均質炉心と非均質炉心）
(Walter-Reynolds, Ref. 10)

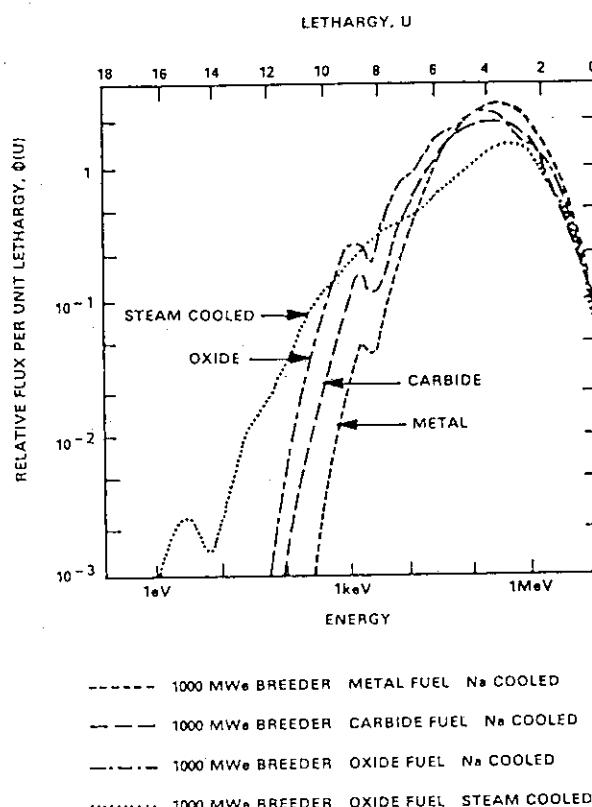


図-2 高速炉炉心の典型的な中性子スペクトル（各種設計オプションに対する比較）
(Walter-Reynolds, Ref. 10)

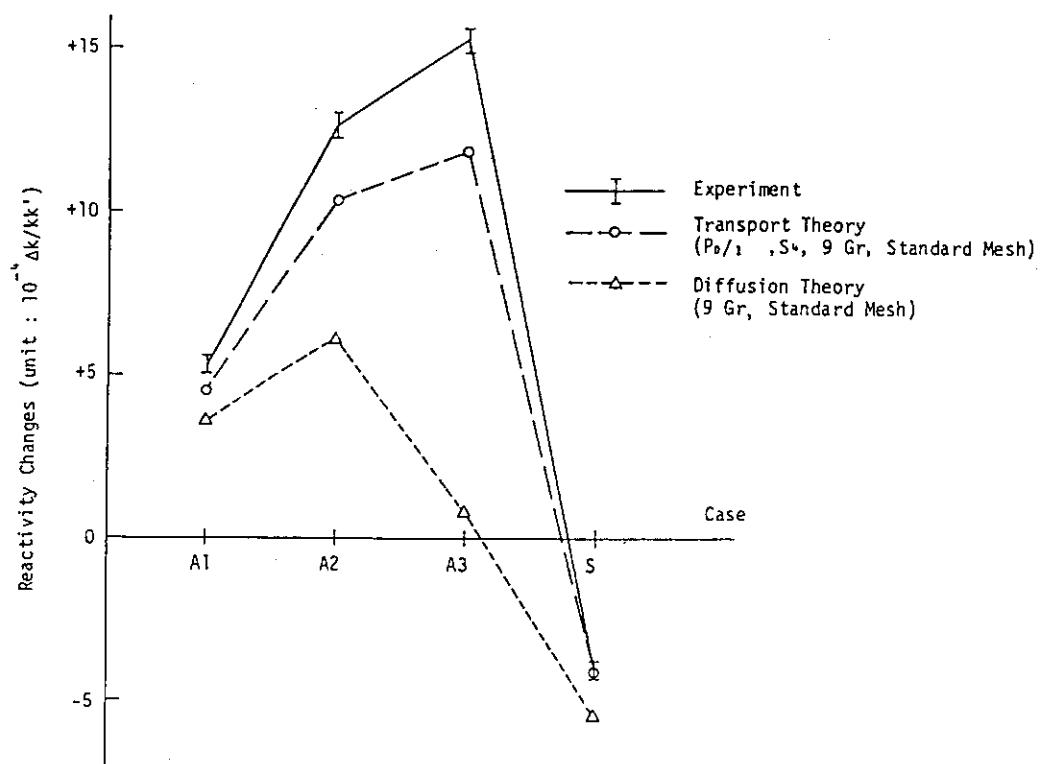


図-3 SIMMERによるFCA VIII-2燃料スランピング実験の解析

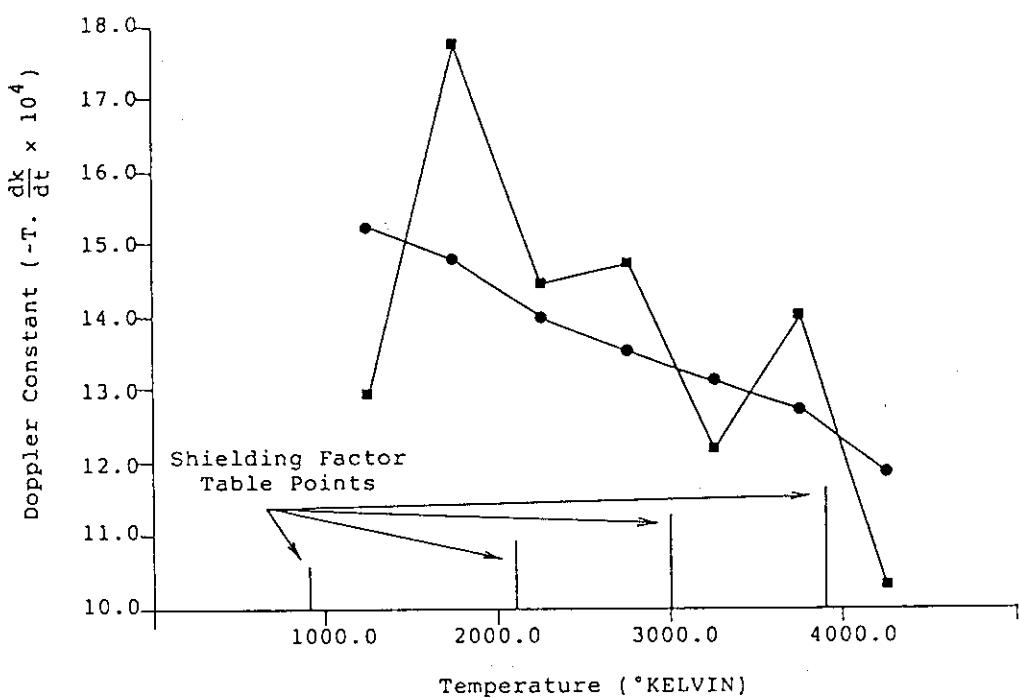


図-4 遮蔽因子内挿法の違いによるドップラー定数の比較

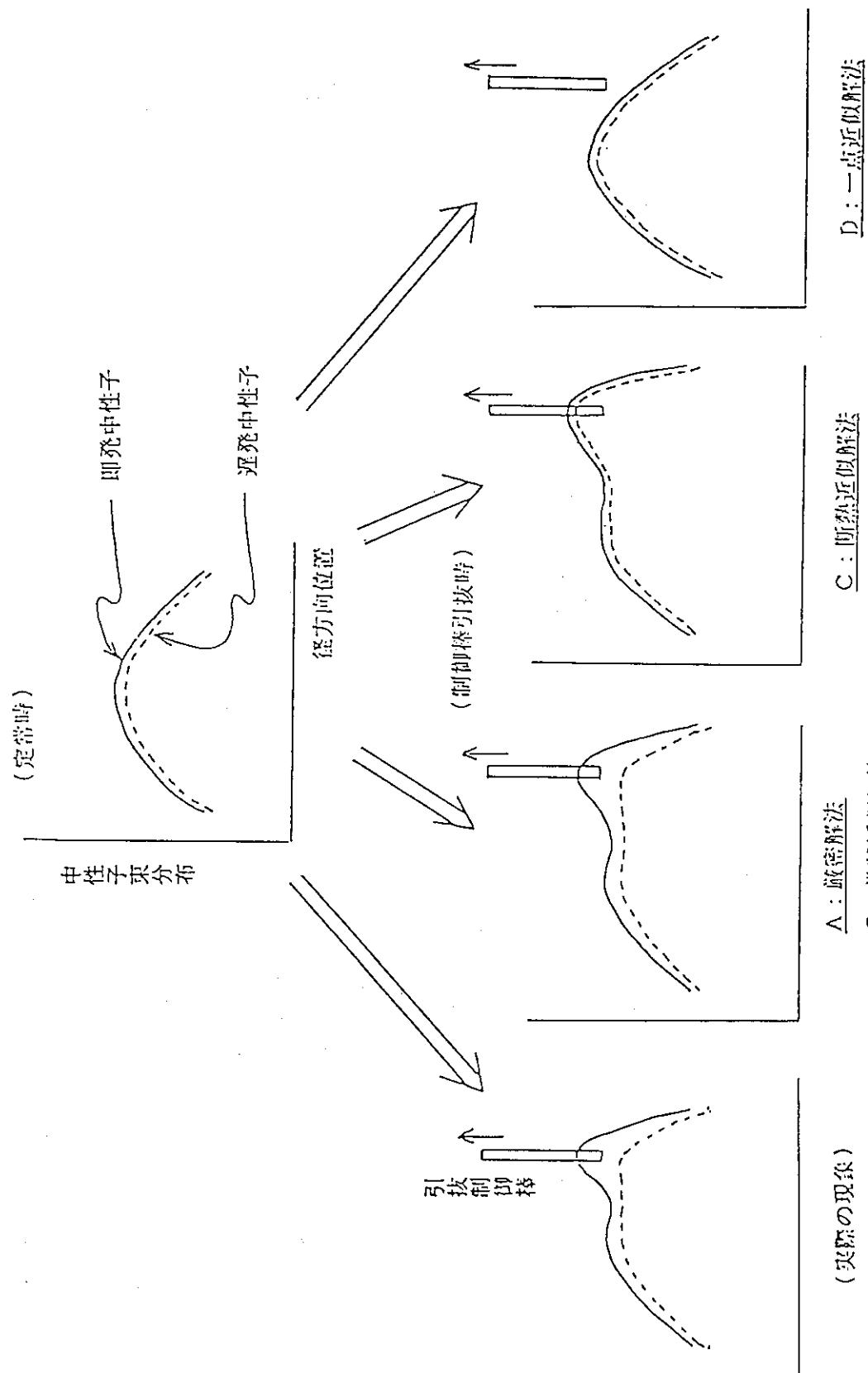


図-5 制御棒誤引きき抜きに対する中性子束応答の定性的比較

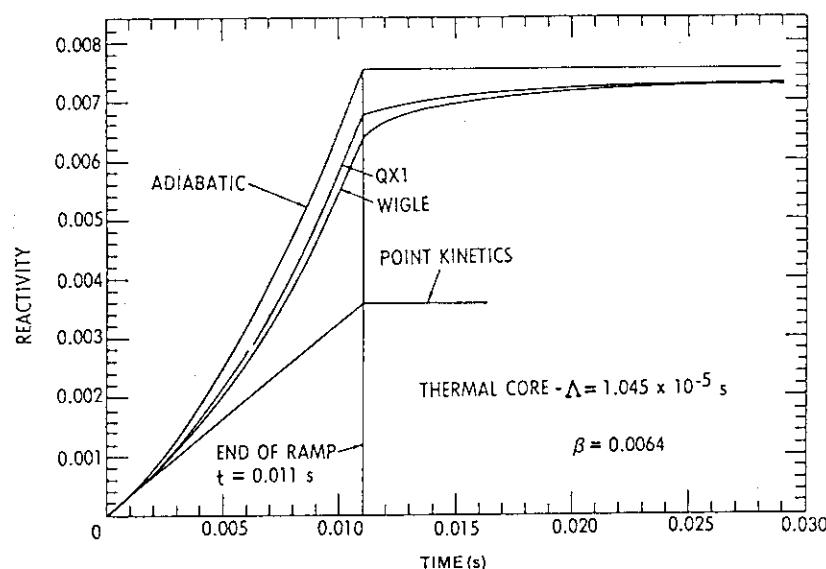


図-6 热中性炉の反応度挿入に対する動特性解析手法の比較
(Ott-Meneley, Ref. 32)

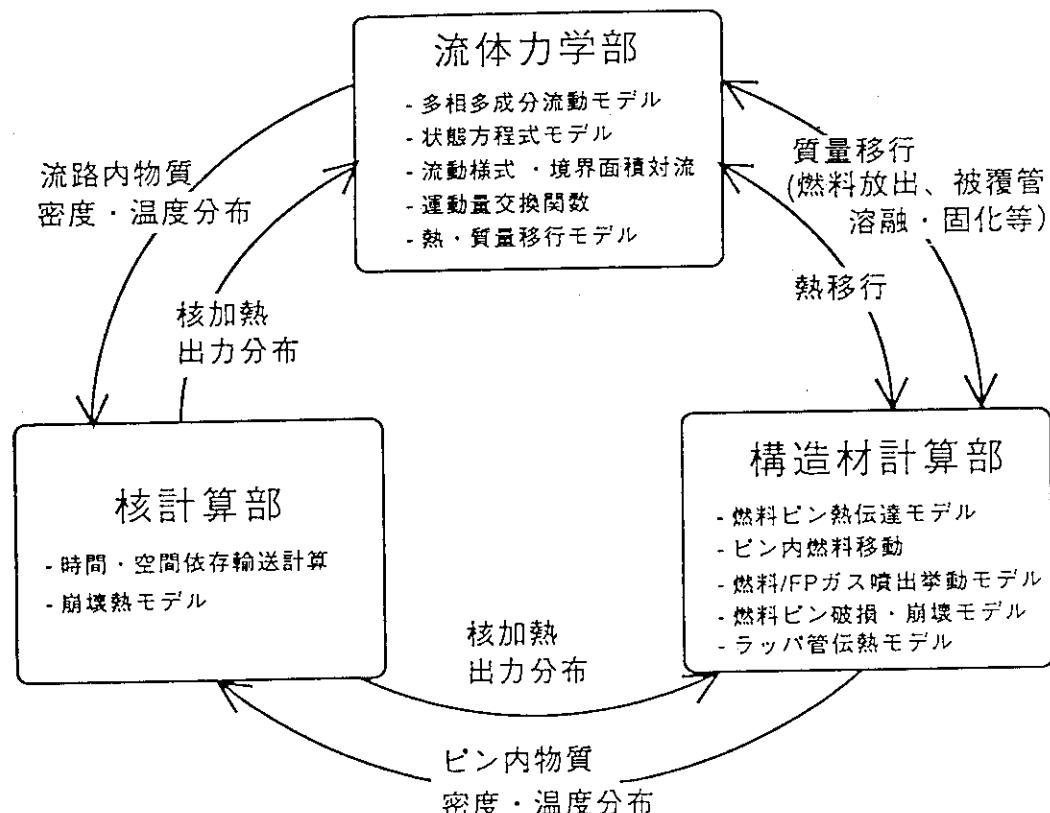


図-7 SIMMER-III コードの全体構成

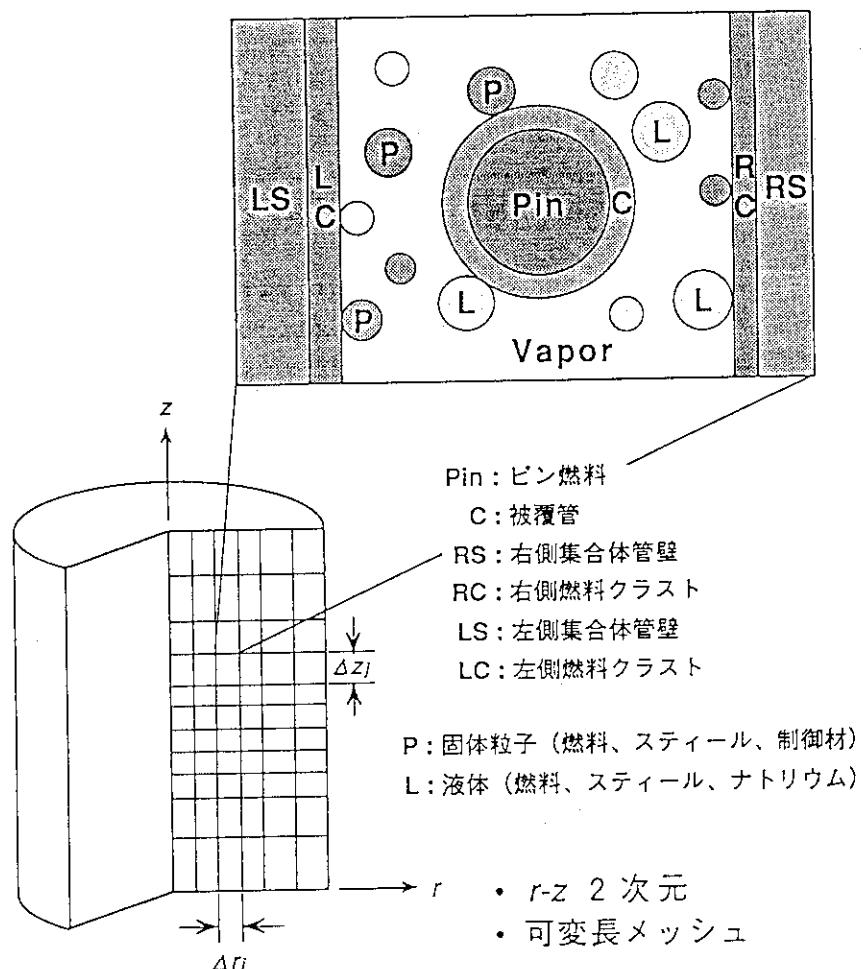


図-8 SIMMER-III コードの計算体系概念図

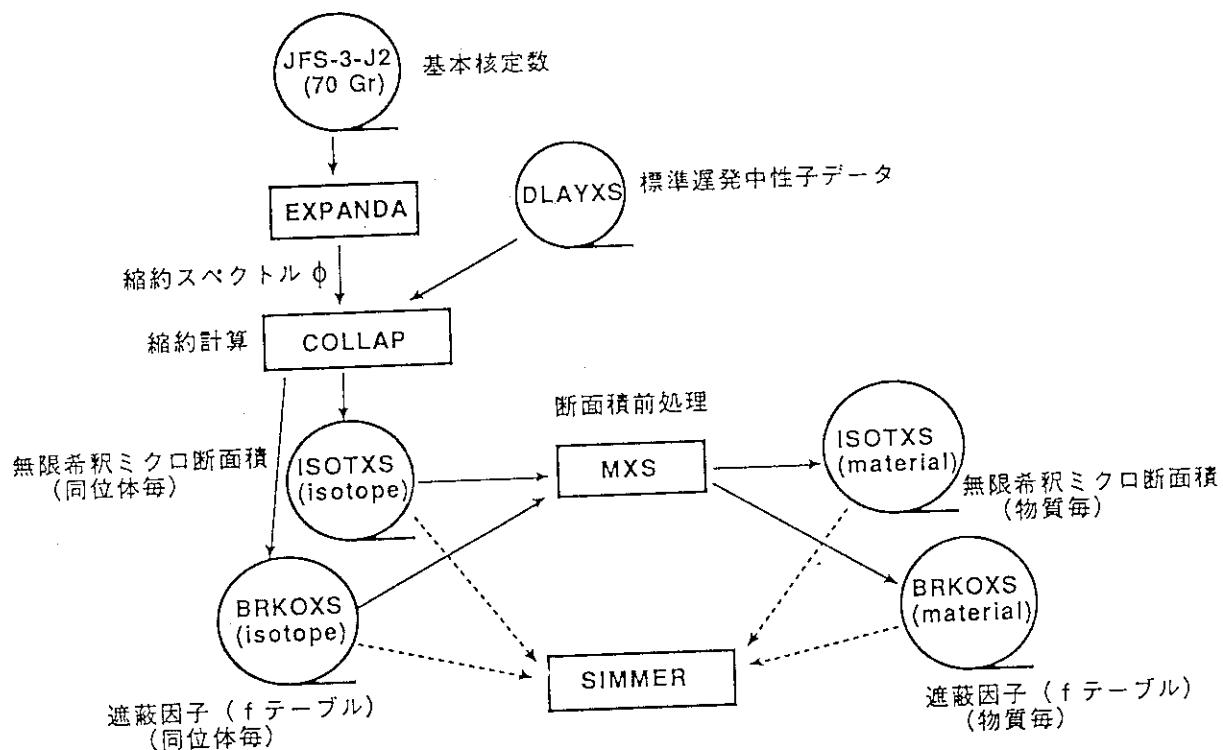


図-9 SIMMER コードにおける断面積処理の流れ

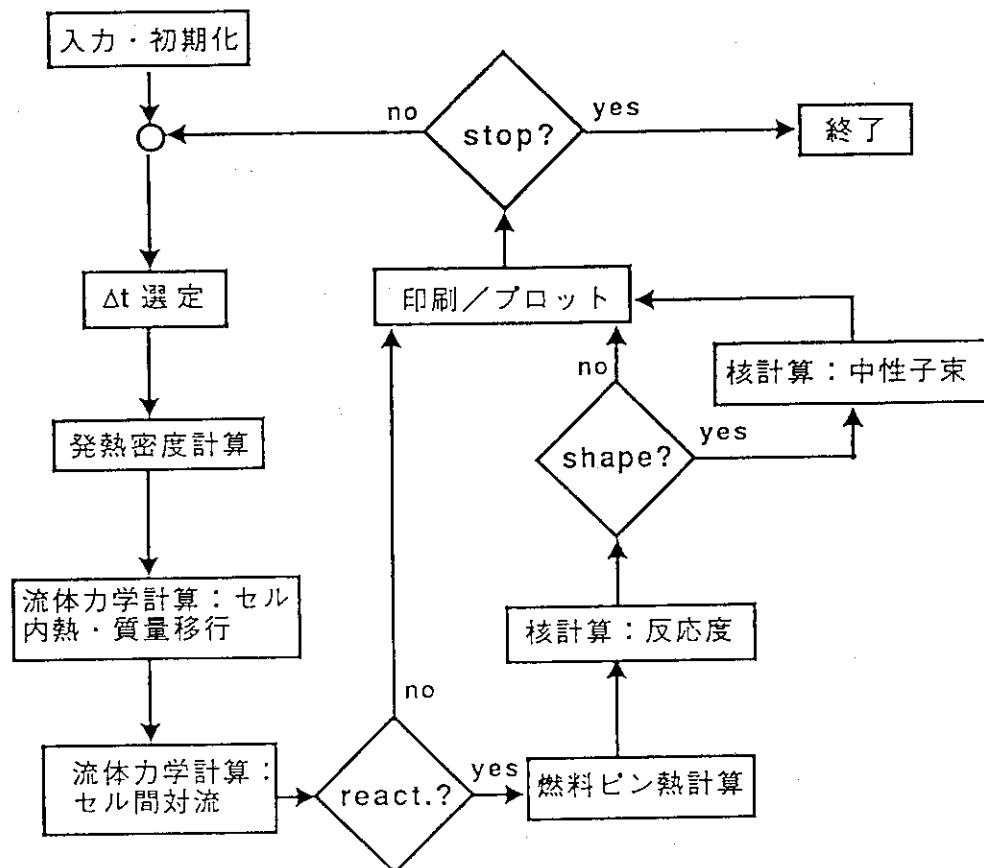


図-10 SIMMER コードの準静近似の計算の流れ

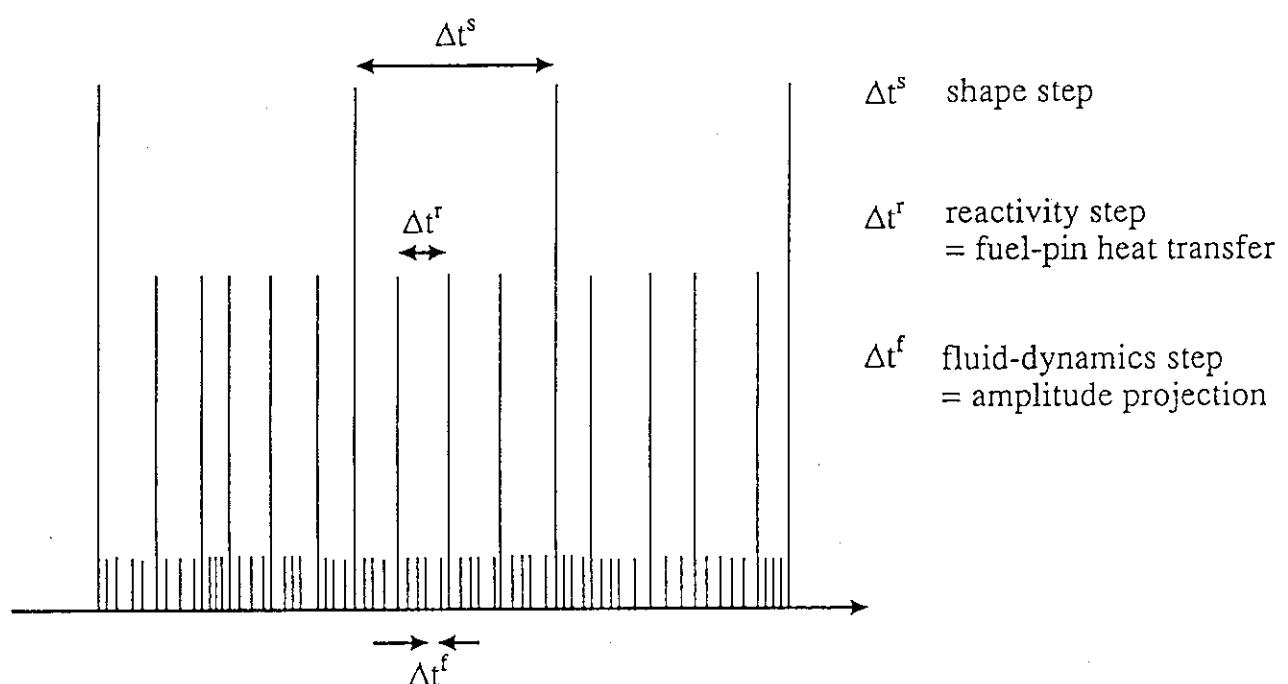


図-11 SIMMER コードにおけるタイムステップの階層構造

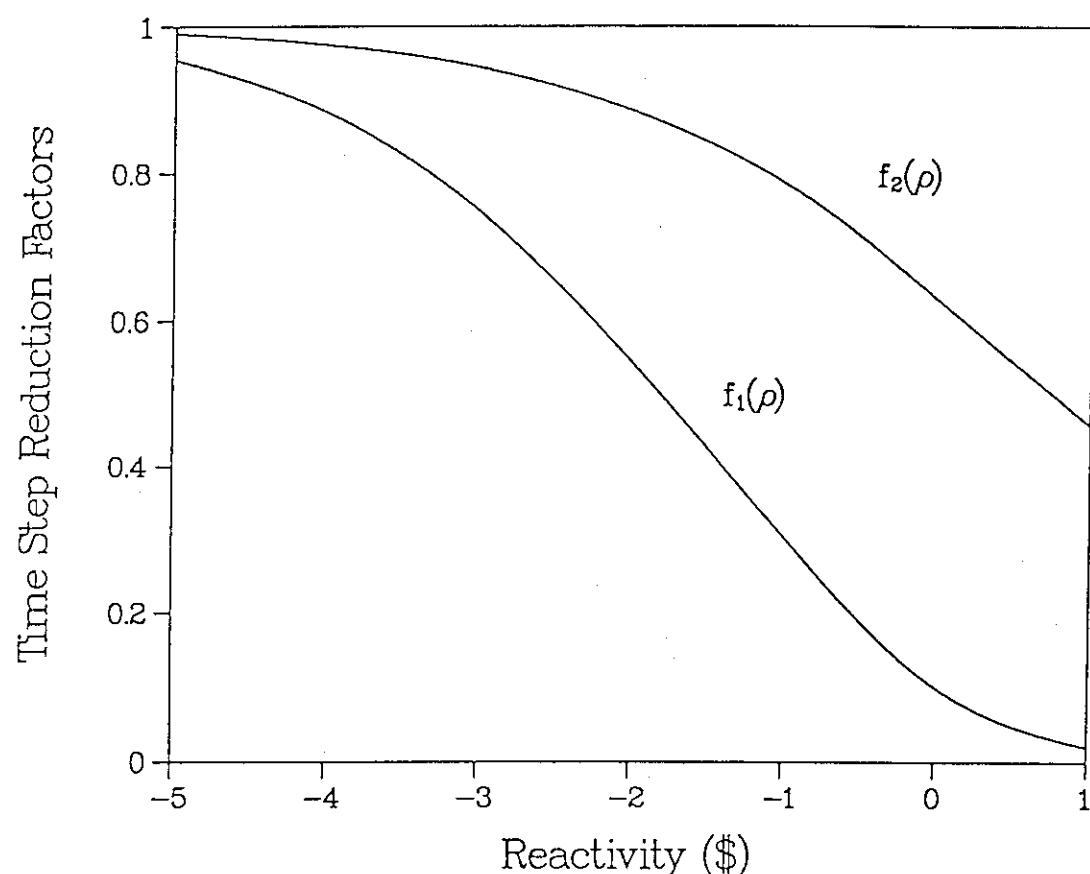


図-12 SIMMER コードで採用した核計算タイムステップの低減因子

11. パネル討論会 「原子力分野における専用計算機への期待」

11.1 プラズマ・シミュレーション専用計算機—METIS

谷 啓二 (原研)

最近の計算機科学、技術の発展により核融合研究も大きく進展した。特に、ベクトル計算機の出現により、非線形磁気流体(MHD)安定性解析など、従来不可能であった計算が可能となった。

しかし、実験データ解析、核融合炉設計に不可欠な解析データベースを早急に整備する必要があることや、現存する世界最高速ベクトル計算機をもってしても十分解析できない問題が山積していることなどから、プラズマシミュレーション専用マシンの導入が期待されている。

前者の問題としては

- 1) 3次元軌道追跡モンテカルロ計算
- 2) 非線型MHD安定性解析

後者の問題としては

- 3) プラズマ乱流解析
- 4) MHD振動と荷電粒子の相互作用解析

などがあげられる。これ等の問題はいずれも、最近目覚しい発展をとげているマイクロプロセッサの並列技術を応用した超並列計算機により極めて効率よく処理できることがわかっている。

ここでは、これ等の解析に必要とされる計算処理量とそれを十分まかなえる専用計算機として設計されたプラズマシミュレータMETIS(Multi-Element Tokamak-oriented Integrated Simulator)について報告する。

An Exclusive-Use Computer for Plasma Simulation "METIS"

Keiji Tani (JAERI)

Nuclear fusion research has been progressing by keeping step with the recent progress of computer science and technology. In particular, the advent of vector processors has allowed us to analyze many problems which have previously been difficult to investigate such as non-linear magnetohydrodynamic (MHD) stability. However, not only the necessity of databases to support analysis of experimental data and the design of nuclear fusion reactors but also the enthusiasm to study problems which are impossible to solve even with up-to-date vector processors, motivate us to introduce an exclusive-use computer for plasma simulation.

Examples of the former are:

- (1) Three dimensional orbit-following Monte-Carlo calculation, and
- (2) Non-linear MHD stability analysis.

Examples of the latter are:

- (3) Plasma turbulence analysis, and
- (4) Analysis of mutual interaction between MHD oscillation and particle motion.

All of the calculations mentioned above can be well executed by a massively-parallel computer which makes use of the recent remarkable progress in microprocessors. In my presentation, I will discuss the necessary amount of computation for these calculations, and the plasma simulator METIS (Multi-Element Tokamak-oriented Integrated Simulator) which has been designed to have enough performance capability for these analysis.

Keywords (plasma simulation, massively-parallel computer, exclusive-use computer)

11.2 並列計算機とモンテカルロ遮蔽計算

(Position Talk)

高野 誠 (原研)

モンテカルロ遮蔽計算では、線源から放出される中性子や光子の、ある着目した場所での通過量を評価する。しかし、放出された中性子や光子のごく一部のみが評価に寄与するため、通常多くの試行回数を必要とする。そのため、最高速のスカラー計算機を使用しても十分満足できる遮蔽解析を行うことは容易でない。また、ベクトル化の試みも飛躍的な計算速度向上までには至っていない。ここで、モンテカルロ遮蔽計算では各粒子のランダムウォークは互いに独立であり、並列計算機向きである。AP-1000によるこれまでの検討で、処理速度倍率は（セル台数） \times 0.8程度を期待出来そうであり、遮蔽計算だけでなくモンテカルロ法による炉心3次元の出力分布計算や動特性計算も実現可能となろう。

今後の並列計算機に対するハードやソフトへの要求として次のような点が挙げられよう。

- (1) 並列コンパイラ (既存コードの並列化。ベクトル機の利用でも主に既存コードがベクトル化され、新たに開発されたベクトル版コードは少ない。また、動的な負荷分散調整機能も必要。)
- (2) 並列機の標準化 (原子力コードのポータビリティ。ベンチマーク等で十分検証された大規模なコードシステムを各国で標準コードとして採用することが多くなっている)
- (3) 並列機用検定済み乱数発生パッケージ
- (4) Disk の利用 (連続エネルギーモンテカルロ用データ ~100 MB)

遮蔽解析用モンテカルロコードの並列化は、米国 ORNL や英国 BNFL 等でも行われているが、セル台数が最大でも数十台の小規模なものが多く、セル台数の増加と共に発生する種々の問題について検討するのは困難と思われる。

Parallel Computer and
Monte Carlo Shielding Calculation
(Position Talk)

Makoto TAKANO (JAERI)

In the shielding analysis by Monte Carlo method, the number of neutrons or photons crossing at a certain point in a system needs to be evaluated. But, only a small fraction of emitted particles from a source reaches to the evaluation point, hence in the Monte Carlo method, a substantial number of random walks are inevitable in nature. Therefore, to obtain a sufficient result of shielding problems is not an easy task even if we employ the latest large main frame computer extensively. And the vectorization technique seems to be failed to achieve remarkable speedup in Monte Carlo calculation. However, Monte Carlo calculation is essentially quite suitable for the parallel computer since random walks of particles are independent of each other. And, the experience of working with AP-1000 has made us confident of achieving the speedup of up to 80% of the total number of cell processors, for example 800 times of speedup by 1000 cell processors. This will enable us to perform core power distribution and core dynamics calculation in 3 dimensions.

The followings are our requirements for the next coming parallel computers;

- (1) Parallel compiler (Parallelization of existing large codes. As seen in the case of vector computers, many existing codes are vectorized, but few newly developed codes for the vector. And a function of automatic distribution of loads among cell processors.)
- (2) Standardization of Parallel Computers (To assure the portability of nuclear codes. Recently, the same code system, validated by large number of benchmarks and used extensively for licencing, is employed world wide since it is not possible for many countries to develop a large code system.)
- (3) Random number generator for parallel computers
- (4) Availability of disk storage (Point wise cross sections occupy ~ 100 MB)

In other countries, such as ORNL in the US, BNFL in UK, the similar studies are going, but the number of cells is much smaller than ours and they may not be able to find problems arising with the increment of cell processors.

11.3 超並列コンピュータへの展望

栗田 康男・黒川 利明（日本IBM）

超並列コンピュータへの取り組み方には様々なアプローチがありうるだろう。IBMの研究部門においても、例えばワトソン研究所（ヨークタウン）においては、GF-11のような専門計算機を物理系の研究者がイニシアチブを取って開発しているし、東京基礎研究所においては、小規模並列のワークステーション TOP-1 を汎用計算機と位置づけて開発している。

汎用的なハードウェアという考えを押し進めれば、既存のネットワークに多数の科学技術計算機用ワークステーションを並べて計算させるという方法も考えられるだろう。

「計算機屋」の観点からは、本当に特殊な「専用計算機」の開発は、半導体技術や通信技術、コンパイラ技術やOS技術といった高性能計算機を支える主要技術への投資を考えると、専用用途もカバーする「汎用計算機」の開発ほどには、うまくゆかないようと思える。

Towards Super-Parallel Computer

Yasuo Kurita・Toshiaki Kurokawa (IBM Japan Ltd.)

There are plural approaches towards so-called "super parallel" computers. For example, even within IBM Research, at T.J.Watson Research Center, Yorktown, USA, developed special purpose parallel machine, GF-11, whose initiatives those researchers in physics major took. On the other hand, at Tokyo Research Laboratory, Tokyo, Japan, developed a small-scale parallel workstation, called TOP-1, which aims to be a general-purpose machine. When pushed along the general-purpose machine, it can be possible to recognize a network with a lot of high-performance Engineering/Scientific workstations as a parallel machine to perform a parallel job.

With the prejudice of computer engineer, the development of very special-purpose machine is more difficult in its development justification than that of general-purpose machine on the necessary development burden (in terms of human resources and necessary budget) for semi-conductor technology, communication technology, computer technology, operating system technology, and so on.

11.4 スーパーコンピュータの動向について

小池 誠彦（日本電気）

スーパーコンピュータは、単一ベクトルプロセッサから、SX-3、CRAY-YMPのような10台以下の並列ベクトルプロセッサへと発展してきた。今後は以下の3つの方向へと発展していくと考える。

・低並列ベクトルプロセッサ型

SX-3の延長線上のプロセッサとして、高速デバイスを用いた超高性能プロセッサを用い、台数は数十台以下の低並列型。

・高並列ベクトルプロセッサ型

集積度の高いデバイスを用いた高機能な単体プロセッサを、数十台から数百台接続する。

・超並列スカラプロセッサ型

単体プロセッサは、可能なかぎり単純化して、プロセッサを数千台から数万台接続する。

これらは、現在のデバイスのトレードオフ(集積度、高速性)に起因するものであり、現時点では低並列のものほど、汎用性が高く、並列度が増す程、コストパフォーマンスが高くなる傾向にある。今世紀中は、これら3つの分野々々において、応用プログラムの適用範囲をふまえた研究開発を進める必要があり、21世紀を境に、デバイスの進歩に伴って、1つの方向に集約されていくものと考える。

Trends in Supercomputing

Nobuhiko Koike (NEC Corporation)

Supercomputers have evolved from single vector processors to multiple vector processors such as SX-3 and CRAY-YMP. We believe that the future supercomputers will evolve into the following three types.

・ Low-Level Parallel Vector Processors

Directly evolving from the SX-3 line, these processors will use ultra-high speed devices. The number of processors will be limited to less than 100.

・ Highly Parallel Vector Processors

Connect tens to hundreds of processors made of highly integrated devices.

・ Massively Parallel Scalar Processors

Connect thousands to tens of thousands of very simple processors.

The reason that there will be these three types stems from the trade-off between speed and integration in the current device technology. At present, the low degree parallel machines have more generality while, the higher parallel machines attain better cost performance if there is enough parallelism.

In the next decade, it will be necessary to make researches in each of the three areas, keeping in mind the suitability of each application area. In the next century, with the development of the device technology, we believe that the three areas will converge into one.

11.5 リアルワールド情報を実時間処理するための超並列プロセッサ

浅川 和雄

株式会社 富士通研究所

リアルワールド情報を処理し、知能機械、たとえばロボットを制御するには、曖昧情報を実時間で統合処理し、しかも環境の変化に自律的に適応する情報処理システムが必要である。これには、柔軟な推論、パターン認識、行動制御が実行できる知能情報処理システムが有用であって、高速処理を主眼とした従来の並列コンピュータでの実現は難しい。並列、分散、学習を特徴とするニューロコンピューティングを実時間処理する超並列のニューロコンピュータの開発が待たれている。

ニューロコンピュータでは、環境のモデルをプログラムで記述するのではなく、事例あるいは設定された評価をベースに環境のモデルを実時間で学習により獲得する。しかし、ニューロコンピュータがリアルワールドのモデルを実時間で学習獲得し、環境変化に自律的に適応するには、概算でも数十ギガFLOPS の処理速度を持つマシンが必要となる。こうした背景から、富士通研究所では、超並列のニューロコンピュータの開発を目指した研究開発を推進している。

Massively Parallel Computer for Real World and Real Time Information Processing

Kazuo Asakawa

Computer-Based Systems Laboratory
FUJITSU LABORATORIES LTD.
1015, Kamikodanaka, Nakahara-ku, Kawasaki 211, JAPAN

It is required for an intelligent robot working in real world to integrate flexible induction, real time pattern recognition for ambiguous sensory input, and adaptive behavioral control. These are not adequate for conventional parallel computers aiming to fast mathematical calculation. Neurocomputers with parallel distributed processing and learning are straightforward to massively parallel computers.

Neurocomputers are able to acquire environmental model by learning instead of tedious symbolic programming. More than ten Giga FLOPS processing speed will be required for realization autonomous adaptation to environmental changing. Therefore, we research and develop massively parallel neurocomputers to process real world information in real time.

11.6 並列計算のためのソフトウェア

梅谷 征雄（日立製作所中央研究所）

原子力などの分野で要求される高速性は今後主に並列計算によって達成される。共有記憶の並列計算機ではプログラミングの手法は現状のスーパーコンから大きく変わらないであろうが、分散記憶の場合は大きな変革が必要になろう。従来ソフトウェアのポートイングには自動並列化+対話的プログラム変換を用いるのが普通であろうが、残念ながらベクトル化等に較べて格段に手間と時間の掛かることを覚悟しなければならない。新規開発には、たとえば数値核融合炉のような分野ごとの専用ソフトウェアをユーザ・ベンダ共同で開発するのが早道ではなかろうか？この場合、視覚インターフェースとオブジェクト指向がキーワードになろう。この二つの技術により、ソフトウェアの検証・改良・拡張が非常に容易になるであろう。一方、ネットワーク分散環境のインフラを活かすために、異機種間でソフトウェアの相互乗り入れやポートイングを行う仕掛けが追求されねばならない。いずれにしてもソフトウェアにより多くの投資の必要な時代となる。

Software for Parallel Processing

Yukio Umetani(HCRL)

High performance of the computer as required in atomic energy field will mainly be accomplished by parallel processing from now on. The programming methodology for the shared-storage parallel processor will not change so much from current one. But for the distributed-storage parallel processor, drastic change will occur. I anticipate that the orthodox method combining automatic parallelization and interactive tuning will be used for porting existing software. But it will require tremendously vast amount of time and effort compared to those for vectorization. For the new development, more application-oriented approach like developing numerical fusion reactor by the collaborative effort of user and vendor seems to be feasible. In this case, "visual interface" and "object-oriented programming" are the two important points. Owing to these characteristics, I expect that the validation, improvement, and the enhancement of the developed software will become much easier. Another point to discuss is the necessity of the framework to enable the cross execution or the porting of software between different machines. Anyway, more investment will be required on software side in future.

Keywords (parallel processing, interactive tuning, visual interface, object-oriented programming, porting)

12. ポスターセッション

12.1 3次元炉心ヒートアップおよびFP放出源評価コード：SHAPE

芳賀 暢, 日高 昭秀 (原工試・安解析)

LOCA時の炉心露出条件下で、一次冷却系に放出される FP 量を現実ベース (BE) で評価する SHAPEコードの概要と、 TMI-2事故をモデルとする事例解析の結果を紹介する。このコードは、出力分布を考慮して3次元的に炉心ヒートアップを計算し、その結果を受けた燃料被覆管破損の時間・空間挙動から、時間経緯を追って FP 放出を定量的に評価する。本コードの機能は基本的に、(1) 空間分布を考慮した炉心内の FP 蓄積量の評価、(2) 空間分布を考慮した炉心ヒートアップ、被覆管酸化量、水素発生、被覆管破損の計算、(3) 炉心から一次冷却系への FP 放出量の計算、(4) カラー CRT表示を含む計算結果の出力、の4つのプロセスから構成される。これらの解析機能について、TMI-2 事故事例解析の CRT 表示例を基にポスター化して紹介する。なお、本件は「 ENS/ANS主催の原子炉安全性に係る国際会議、Avignon, France, Oct. 1988」で報告したものとの再掲示である。

3D Core Heatup and FP Source Evaluation Code : SHAPE

T. Haga and A. Hidaka (NUPEC/JINS)

The SHAPE code, to be used for the BE calculation of FP release into the primary loop under LOCA condition, is briefly described together with the result of demonstration calculation on the TMI-2 accident. This code realistically calculates core heatup considering 3D power distribution, and calculates space-time occurrence of cladding rupture, and then calculates successive FP release into the primary loop. The code functions basically consist of 4 steps, i.e., (1) evaluation of the core FP inventory with spatial distribution, (2) calculation of 3D core heatup, cladding oxidation, hydrogen generation, and cladding rupture, (3) calculation of successive FP release from the failed fuel into the primary loop, and (4) output of the calculated result including color CRT display. Those code functions are demonstrated by postars showing several examples of color CRT display on the TMI-2 accident. These results were also exhibited in the ENS/ANS Specialist Meeting on Thermal Reactor Safety, Avignon, France, Oct. 1988.

Keywords (LOCA, 3D, core-heatup, cladding, oxidation, rupture, FP-release)

THE SHAPE CODE FOR
CORE HEATUP AND FISSION PRODUCTS SOURCE EVALUATION

T. Haga and A. Hidaka (JINS, Japan)

A computer code SHAPE has been developed to evaluate fission products source in nuclear reactors under the accident conditions which are supposed to exist potentially. The code first calculates spatial core heatup in three dimensional mode, and then evaluates fission products(FP) release from fuels considering various aspects of fuel damage conditions. The SHAPE's calculational procedure basically consists of four individual steps, i.e., calculations of FP source inventory in a reactor, spatial core heatup, FP release from fuels, and lastly the CRT display of calculated results. Several test calculations have been made to study the code performance, with examples on the BWR and PWR core heatup and FP release under the typical LOCA core uncovering condition.

1. INTRODUCTION

To evaluate FP source and its release into the plant environment in the accident situation of nuclear power plants, a computer code SHAPE has been developed at Japan Institute of Nuclear Safety (JINS). This code calculates the core heatup and FP release from fuels in three dimensional mode, aiming to make a realistic evaluation considering detailed core status, i.e., fuel types and loading pattern, burnup and power distributions, etc. Accordingly, the code is designed to evaluate the following physical quantities with space-time behaviors under the accident situation. (Fig. 1)

- * Fuel temperature in steady state, and during transient.
- * Fuel cladding temperature in steady state, and during transient.
- * Zr-H₂O reaction during heatup transient condition.
- * Fuel cladding rupture during heatup transient condition.
- * FP release from pellet to gap under steady condition.
- * FP release from pellet to gap during transient heatup condition.
- * FP release from fuel to coolant by cladding rupture.
- * Distributions of important parameters obtained from spatial profile.

To study the SHAPE code performance, several example calculations have been made on BWR and PWR, each with a typical LOCA core uncovering condition. The case for BWR calculation considered the multiple failure accident with an ATWS condition, while the case for PWR took up the TMI-2 accident as an example. This paper will describe a brief outline of the SHAPE code analysis model, and then show the result of those example calculations together with the related technical discussions.

Since the SHAPE code evaluates the FP source in the fuel and its release into the coolant, it should then be followed by the evaluation of its transport in the primary coolant system, and also in the plant system within containment. Other computer codes, MACRES and WETBERAN, are provided for this purpose, and these three codes together constitute a JINS' code package of source term issues.

2. DESCRIPTION OF THE SHAPE CODE

2.1. Incore FP Inventory Data File

The incore FP source inventory is first calculated based on the status of each fuel by the ORIGEN-2 code(1) being incorporated in the SHAPE code system. The calculated FP inventories constitute a FP data file of the SHAPE system, comprising of 11 elements(41 nuclides) which are considered hazardous if released to the external environments. This data file is given for each fuel type, and in terms of 10 burnup steps up to 40 GWD/T, and elapsed time during 7 days after reactor shutdown. The FP distribution in the actual core shall be described in 3-dimensional mapping by making use of this data file, where any data coming between those burnup and time steps are given by interpolation. The total incore FP inventory is then calculated by integrating the distributed local inventories in the reactor. This eventually considers the fuel irradiation history in the reactor to be followed by the space-time heatup behaviors after the transient initiation, but it is virtually an off-line computation to the SHAPE system which is carried out as a preparatory procedure prior to the case run.

2.2. Core Heatup and FP Release from Fuels

The SHAPE's calculation actually starts from spatial core heatup under thermal transient condition. To describe an initial core status prior to the transient, the reactor is divided into a multiple number of regions having different initial conditions. These conditions are generally taken from two major data sources, one is the core status data including power and burnup mapping prior to initiation of the transient, and the other is the thermal-hydraulic conditions obtained either from measurements or from calculations given by RELAP5 code for example. Thus, the fuel heatup calculation is carried out in the following procedures. The basic heatup equations for fuel and coolant are then given by,

$$\frac{1}{r} \frac{\partial}{\partial r} (k_r r \frac{\partial T}{\partial r}) + \frac{\partial}{\partial z} (k_z \frac{\partial T}{\partial z}) + Q = \rho c \frac{\partial T}{\partial t} \quad \dots \dots \dots \quad (1)$$

$$-F_e C_e \frac{\partial T_e}{\partial Z} + Q_s = \rho_e C_e \frac{\partial T_e}{\partial t} \quad \dots \dots \dots \quad (2)$$

where T = fuel rod temperature in R/Z geometry,
 k_r, k_z = heat conduction coefficient for fuel pellet or cladding,
 Q = heat source in fuel pellet or in cladding,
 Q_s = heat transfer to coolant per unit time and unit volume,
 c, ρ = specific heat and density of fuel pellet or of cladding,
 C_e, ρ_e = specific heat and density of coolant,
 T_e = coolant temperature at height z ,
 F_e = coolant mass flow in unit area of the flow path.

The core inlet coolant temperature, inlet flow rate and quality, initial power level, reactor water level, etc., shall be assigned to each of the divided regions as boundary conditions. In addition to these basic equations, the important models are briefly summarized in the following.

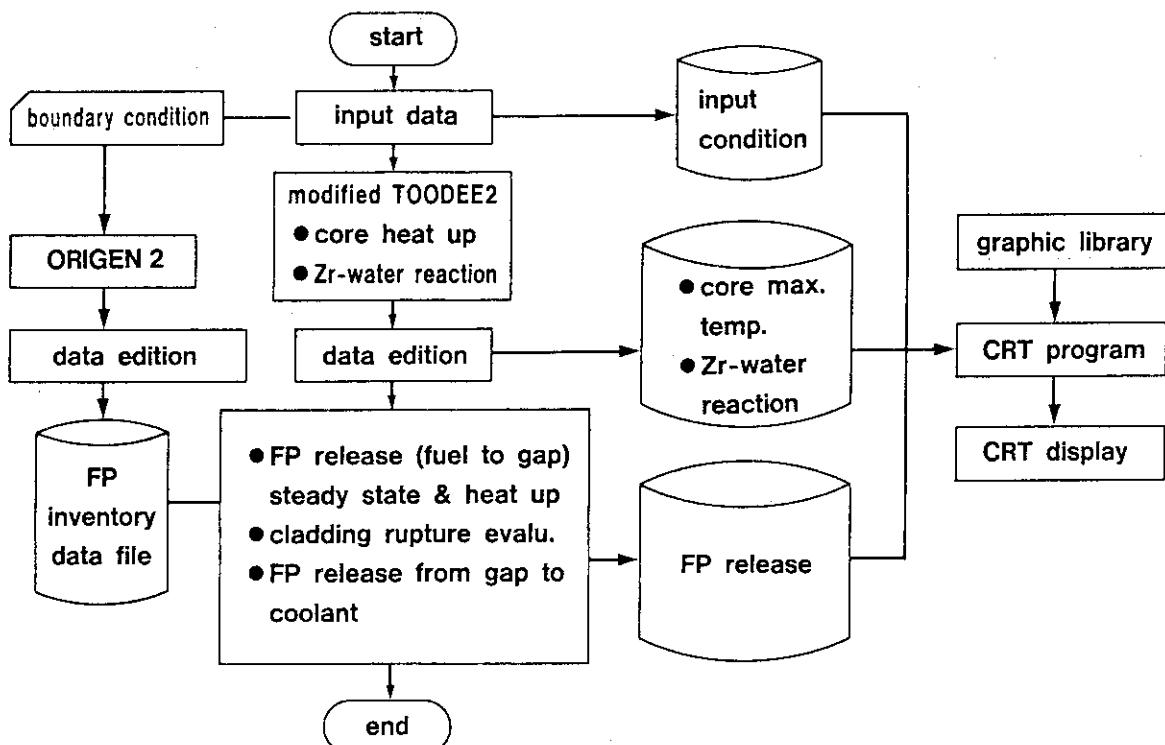


Fig. 1 SHAPE calculational procedure.

(1). Metal/water reaction model

The metal/water reaction (zirconium oxidation) during the fuel heating used to be evaluated by the Baker-Just equation(2), but this model is said to give too conservative result in the higher temperature region. To get a more reasonable result for the metal/water reaction, the SHAPE code employs a following parabolic rule which virtually comprises a number of evaluation models by choosing a different set of parameters to be used, i.e., a set of k and Q_0/R shall be selected optionally to represent a particular model.

$$d^2 = k_0 t \exp\left(-\frac{Q_0}{R T_c}\right) \quad \dots \dots \dots \quad (3)$$

where d = thickness of oxidized cladding layer,
 t = reaction time,
 k_0 = proportional constant to represent reaction rate,
 Q_0/R = activation energy / universal gas constant,
 T_c = cladding center temperature.

Some important considerations in making use of this equation are on the various causes restricting the metal/water reaction to continue. The first to be considered is the existence of thin oxidized layer on the cladding surface which would have already been formed in the previous reactor operation, and the second, perhaps more important, is the restraint of the metal/water reaction due to insufficient supply of steam, possibly because of hydrogen generation in the neighborhood. Also, it is noted

that the oxidation reaction would no longer continue with exhaustion of the cladding material, or with the temperature exceeding much beyond the cladding melting point. These restrictive effects to the metal/water reaction are considered by various physical models in the SHAPE code, which are to be used by optional selection depending on the physical phenomena supposed to be existing.

(2). Heat transfer correlation functions

A number of convective heat transfer correlation functions are provided in the SHAPE code to describe various modes of heat transfer from the cladding surface to the coolant. Most of these correlations functions are known to many people, and therefore their names are only quoted here so that many complicated descriptions might be avoided.

- (a) Single phase forced convection: Dittus & Boelter
- (b) Nucleate boiling: Thom
- (c) Two phase forced convection: Shorock & Grossman
- (d) Stable film boiling: Groeneveld
- (e) Low pressure film boiling: Dougall & Rohsenow

(3). Radiative heat transfer

When fuel heatup is considered at a very high temperature level, with core uncover condition, the effect of radiative heat transfer would become increasingly important. Thus, radiative heat transfer between fuel rods, and between fuel rods and other structural materials, is taken into account in the SHAPE heatup calculation. The heat source is given by Stephan-Boltzmann's equation as a function of the cladding surface temperature, where the geometrical shadowing effect is taken into account by a "view factor method" so as to describe heat exchange in the fuel lattice. The view factor is defined as a fraction of the solid angle from a point of interest, which within its scope overlooks other radiative heat sources in the surroundings. Generally, consideration of radiative heat transfer in the heatup calculation tends to give an effect of smoothing the local heating profile. At a periphery of the fuel lattice, the effect would then result in radiative cooling.

(4). Cladding rupture models

Two optional models are provided for cladding rupture in the fuel heatup process. One is an equivalent of the TOODEE's rupture model(3) that makes use of a pressure-temperature correlation as rupture conditions. These were prepared by ORNL from experimental data. The other is the Chapman's model(4) that gives rupture conditions as a function of temperature together with its increasing rate, and also related with circumferential stress in cladding tube.

In addition to the above cases considered in the fuel heatup process, another mode of cladding rupture is the case which is expected to occur by thermal shock at a time of quenching. If oxidation has proceeded to result in embrittlement of the cladding material prior to the quenching, it would be likely to have fuel fragmentation together with the cladding rupture, causing a large amount of FP release at one moment. In SHAPE, this type of cladding rupture is supposed to take place at quenching condition, particularly when β -phase layer thickness in the cladding would become less than 25% of the total thickness.

(5). FP release from fuels

The last objective of the SHAPE code calculation is to evaluate FP release from fuels and its transport into primary coolant. This is evaluated in two different phases, i.e., in the steady state during plant operation and in the transient state during accident. The former considers the transport of FP from the inner crystalline structure of the fuel pellet to the cladding gap, while the latter considers the transport from the pellet to the coolant in sequential steps, i.e., from the pellet to the gap, and then to the coolant. Thus, the FP release generally depends on the fuel heatup and cladding rupture modes, which is evaluated in the SHAPE code by several analytical models provided for various heating conditions. These models are shown below.

For the normal operation mode, the models provided in the SHAPE code are Manson & Reymann's model(5), modified Mason & Leymann's model, CESIOD model(5), and modified CESIOD's(6). The modified models as above consider the NUPEC (Nuclear Power Engineering Test Center, Japan) fuel irradiation test data, considering the effects of fuel burnup and irradiation history on the FP release rate. The modified CESIOD model also considers the result of CONTACT experiments(6) together with the NUPEC data.

For the heatup mode, three FP release models are provided, i.e., NUREG-0772 model(7), Kelly & Reynold's model(8), and CORSOR-M model(9). Then for the rupture mode, Lorenz & Collins' model(10) is considered for I and Cs, and WASH-1400 gap escape fraction(11) is considered for all other nuclides.

(6). CRT display and editing of calculated result

As for the last step, a provision is made to give the calculated result in a color CRT display system. This is an effective method showing the complicated space-time core heatup behaviors. The display system employs a convenient "menu-selection" method following usual conversation-type procedure. Frequency distributions of various important parameters as obtained from the spatial core behaviors, such as temperature, oxidation, amount of FP release, are also edited and shown for quick understanding.

3. TEST CALCULATIONS ON BWR PWR SAMPLE PROBLEMS

3.1. Calculation on BWR Sample Problem

To study SHAPE code performance, several example calculations have been carried out for the core heatup, each with a typical core uncover condition. For the case of BWR calculation, a multiple failure accident has been considered with an MSIV closure as an initiating event, and with an ATWS condition assuming the failures of all the safety system functions. This eventually results in lowering the reactor water level down to the core uncover condition, due to steam release through SRV. The thermal-hydraulic behaviors at this accident has first been calculated by using the RELAP5/MOD1/JINS/B code, and then the important parameters obtained from this result, such as reactor water level, pressure, coolant flow rate, etc., have been used to initiate the SHAPE heatup calculation. The former is a JINS's version of the RELAP5/MOD1 code, in which a jet pump model for BWR analysis has been incorporated.

An example of the calculated result on the BWR case is shown in Fig. 2, where the fuel temperature behavior at each axial node is shown as a function of time. In this result, the initial power (and temperature) rise is due to the effect of void decrease caused by the MSIV closure, and the fuel heatup actually starts rising at about 150 seconds after the initiation of the accident. The difference in the startup time of the fuel temperature indicates the beginning of the fuel uncoverage at each core height.

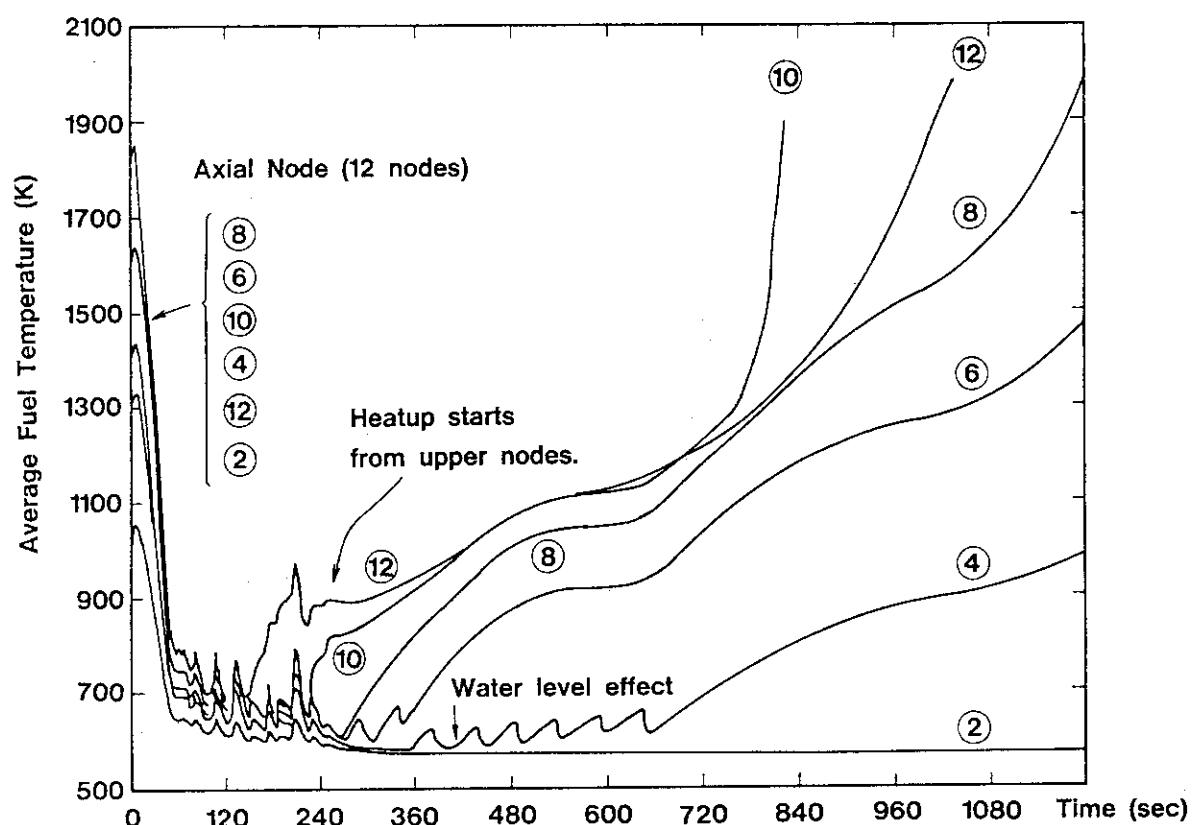


Fig. 2 Fuel temperature behavior (BWR).

3.2. Calculation on PWR Sample Problem

For the PWR calculation, the TMI-2 accident has been taken up as an example, and the result is briefly shown. The calculation covered a time interval of 100-174 minutes after the accident initiation, during which the core uncoverage was considered to have started and resulted in major fuel heatup in the initial phase. The important base data and necessary boundary conditions for the calculation were taken from the Standard Problem Package(12)(13), or prepared by the authors when not available. The maximum value of cladding temperature in the heatup calculation was limited up to 2,500 K, somewhat above the melting point, beyond which the cladding material would melt down and not stay at its original position. When cladding rupture occurred, the metal/water reaction at its internal surface was assumed to take place within 6cm of distance from the ruptured points. A few examples of the calculated result are briefly shown here.

Coolability of the fuel might have been maintained prior to the uncovery at each axial node, but the fuel temperature quickly starts rising after the core uncovery is attained as the mixture water level comes down. Hence, the temperature first picks up from Node 7 (top node), but it is soon exceeded by that of Node 6 having higher heat source. Then, Fig. 3 shows the cladding temperature behavior at Node 6 (hottest node), for each of 5 core zones. Because of the rapid increase in cladding temperature and internal pressure, the tubes would have swelled and ruptured at different time according to the spatial heatup behaviors. The calculated result indicates that the first fuel rupture would have occurred at Node 6, Zone-1 (hottest zone) at 138 minutes after the accident initiation. Except zone 5, the core periphery, the cladding rupture would have occurred at Node 6 axial location, the second node from the top. After the rupture, additional metal/water reaction would take place inside the tube near the rupture point, which then causes further increase of cladding temperature by the reaction heat. In the actual case(14), it has been reported that the first rupture might have occurred at 138 minutes, and the present calculation reasonably agrees with it.

As for FP release from fuels, an example is shown in Fig. 4 for the case of I-131. In this case, partitioned inventories in fuel, gap, and its release into primary coolant, are indicated as functions of time. The calculation used the modified CESIOD model for normal operation, the CORSOR-M model for heatup transient, and the Lorenz & Collins' model for rupture mode.

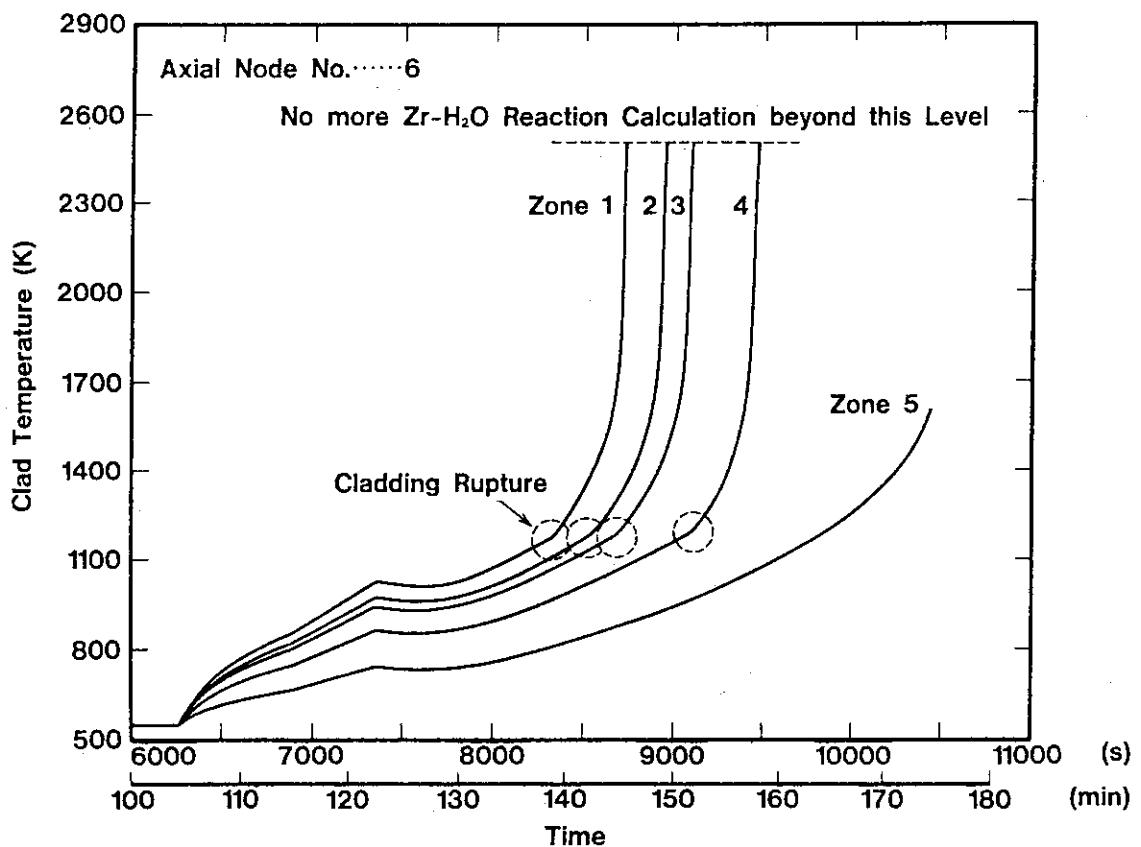


Fig. 3 Clad temperature v.s. time. (TMI - 2)

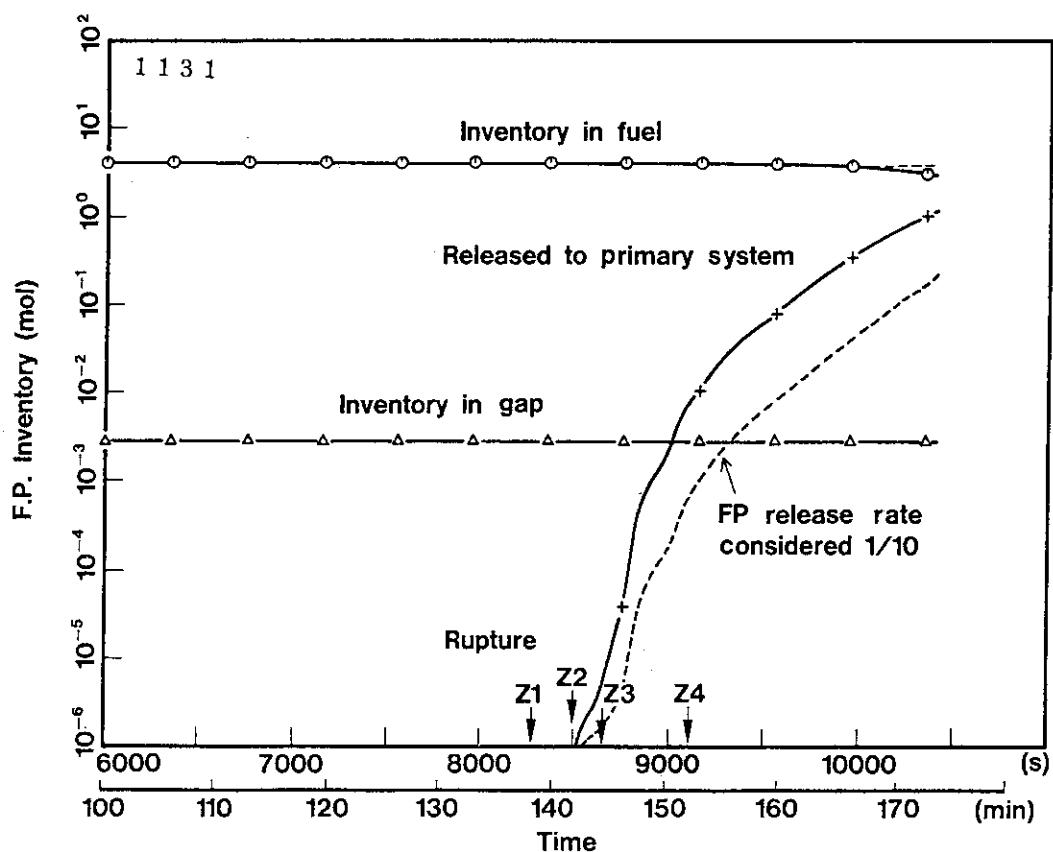


Fig. 4 I-131 partitioning between fuel, gap and RCS(TMI-2)

4. CONCLUDING REMARKS

An outline of the SHAPE code has been briefly described, together with the test calculations made on the BWR and PWR(TMI-2) as examples. The code is designed to calculate spatial core heatup and resulted FP release from the fuel in accident condition. From the restriction of the analytical models, the present code does not consider relocation or melt-down of fuel lattice structure in the course of fuel heatup, except swelling and ballooning. This was the major reason for not having the TMI-2 test calculation beyond 174 minutes, prior to which the regular lattice structure would have been roughly maintained in the actual accident. The future problem for the SHAPE code will have to be focused on this aspect, and studies are underway to expand the code capability to cope with the SCD of a greater scope.

Reference

- (1) A.G. Groff: "A User's Manual for the ORIGEN2 Computer Code". ORNL/TM-7175 (Jul. 1980)
- (2) L. Baker, Jr. and L. C. Just: "Studies of Metal-Water Reaction at High Temperature III. Experimental and Theoretical Studies of the Zirconium-Water Reaction", ANL-6548 (May 1962)
- (3) U.S. NRC: "TOODEE 2: A Two Dimensional Time Dependent Fuel Element Thermal Analysis Program", NUREG-75/057 (Jan. 1975)
- (4) D.A. Powers, R.O. Meyer: "Cladding Swelling and Rupture Models for LOCA Analysis", NUREG-0630 (Apr. 1980)
- (5) D.L. Hagrman, G.A. Reyman, R.E. Mayson: "MATPRO-VERSION 11 (REVISION 2) A Handbook of Materials Properties for use in the Analysis of Light Water Reactor Fuel Rod Behavior", NUREG/CR-0497, TREE-1280, REV. 2, R3 and R4 (Aug. 1981)
- (6) M. Charles, J.J. Abassin, D. Baron, M. Bruet, P. Melin: "Utilization of CONTACT Experiments to Improve the Fission Gas Release Knowledge in PWR Fuel Rods", Water Reactor Fuel Element Performance Computer Modelling, pp. 1-18(1983)
- (7) US.NRC: "Technical Bases for Estimating Fission Product Behavior during LWR Accidents", NUREG-0772(1981)
- (8) J.L. Kelly, A.B. Reynolds, M.E. McGown: "Temperature Dependence of Fission Product Release Rates"; Nucl.Sci. and Eng., 88,2, pp. 184-191(1984)
- (9) M.R. Kuhlman, D.J. Lehmicke, R.O. Meyer: CORSOR User's Manual", NUREG/CR-4173, BMI-2122 (May 1985)
- (10) R.A. Lorenz, J.L. Collins, A.P. Malinauskas: "Fission Product Source Terms for the Light Water Reactor Loss-of-Coolant Accident", Nucl. Technol. 46, pp. 404-410 (Aug. 14, 1979)
- (11) R.L. Ritzman, et al.: "Release of Radioactivity in Reactor Accidents", WASH-1400(NUREG-75/014) APPENDIX VII to Reactor Safety Study, U.S. Nuclear Regulatory Commission (Oct. 1975)
- (12) J.M. Broughton et al.: "Update Information on Standard Problem Package", JMB-33-87 (Jun. 8, 1987)
- (13) D.W. Golden et al.: "TMI-2 Standard Problem Package", EGG-MMI-7382 (Sep. 1986)
- (14) "Investigation into the March 28, 1979 Three Mile Island Accident by Office of Inspection and Enforcement", NUREG-0600, VOL. 1 (Aug. 1979)

12.2 LESによる乱流数値シミュレーション

佐竹 信一, 河村 洋 (東理大)

乱流は、原子力工学をはじめとする多くの分野において極めて重要な流れの形態である。他方、近年計算機の急速な進歩とともに、乱流を数値シミュレーションによって計算機上に再現することができるようになってきた。しかし、実際の工学機器等に現れるような形状が複雑でレイノルズ数の高い乱流については、直接数値計算 (Direct Numerical Simulation :DNS) は適用ができない。そこで、Large eddy simulation (LES) という手法が開発されている。LESは、大きな渦については実際にナビエストークスの方程式を解き、計算格子より小さいスケールの渦についてはモデル化（サブグリッドモデル）を行う。本研究では、十分発達した平行平板間乱流、二重円管内乱流を LES を用いて計算した。平行平板間乱流については、Kim らの DNS データと比較した。また、二重円管内乱流については Rehme の実験結果と比較し、非対称な流れ場が LES によって再現されることを示した。

Numerical simulation of a turbulent flow by LES

Shin-ichi Satake Hiroshi Kawamura (Sci. Univ. of Tokyo)

Turbulent flow is often encountered in the various engineering fields such as nuclear and mechanical engineering. On the other hand, present computer development enables numerical simulation of turbulence. Direct numerical simulation (DNS), however, cannot be applied to real engineering problems where turbulence of a high Reynolds number plays significant roles. Therefore a method called Large eddy simulation (LES) has been developed. The LES is a numerical technique in which large-scale turbulent structures are computed explicitly by the Navier-Stokes equation, and the small structures are modeled by a so-called Sub-Grid model. In this study, fully developed turbulent channel and annuli flows are analyzed numerically using LES. In case of the channel flow, present numerical results are compared with those of the direct numerical simulation reported by J.Kim et al. For the annuli flow, comparison with the experimental data by Rehme revealed that the asymmetrical velocity field is well reproduced by the present calculation.

Keywords (Turbulent flow, Large eddy simulations, Sub-Grid model, Channel, Annuli)

12.3 イオンビーム引き出しシミュレーションにおける並列化手法

久保田昌晴, 井門 俊治 (埼玉大学)

中性粒子入射加熱 (NBI) は磁気閉じ込め型プラズマの加熱法として重要である。ビームの軌道は引き出し電極の構造に影響を受ける。イオン源からのイオンビームの軌道を計算するための計算コードを開発し、このコードを計算時間を短縮するために並列化した。良く知られているように並列化はいくつかの場合には有効である。物理問題それぞれについて並列コンピューティングの効果を調べた。多数のトランスペュータを用いての並列計算の能力はイオンビーム引き出しシミュレーションにおいてかなり高い。並列化したイオンビーム引き出しシミュレーションについての計算時間について報告し、またここで実際の物理問題のシミュレーションにおける並列化手法と適応性について述べる。

Parallelization method for ion beam extraction simulations

Masaharu Kubota and Shunji Ido (Saitama University)

Neutral Beam Injection (NBI) is an important method for heating the magnetically confined plasmas. Beam optics depends on the configuration of ion beam extraction electrodes. Calculation code was developed to simulate the trajectories of ion beams from source plasmas. This code was parallelized to reduce the calculation time. As well known, parallelization can be useful in some cases. We examined effects of parallel computing in each of physical problems. Parallel performance by using several transputers was found to be much excellent in ion beam extraction simulations. Calculation time of parallel ion beam extraction simulation is reported, and the methods of parallelization and applicability of parallelism in the simulations of actual physical problems is shown here.

Keywords (parallel computing, transputer, ion beam extraction)

12.4 MIMD型並列計算機の物理シミュレーションへの応用

彦坂 聰, 井門 俊治 (埼玉大学)

物理シミュレーションの高速化のために並列化は有効な方法である。しかし、並列化のための自動コンパイラは十分ではなく、高並列化システムに対しては解くべき問題に応じた並列化技法とプログラミングが必要である。ここではM I M D型の並列計算システム中のマイクロプロセッサとしてトランスペュータT 8 0 0 及び i 8 0 8 6 0 を用いた。

本研究ではベンチマーク及び物理シミュレーションとして、

- (1) 通信時間ベンチマーク
- (2) 2次元熱伝導方程式シミュレーション (98%)
- (3) 2次元ランダムウォークシミュレーション (97%)
- (4) イオンビーム引出しシミュレーション (97%)

についてトランスペュータで()内に示した並列化率を得た。これらを iPSC/860 で計算を行なった結果を報告する。

Performance of Physical Simulation applied in MIMD type Parallel Processing Systems

Satoshi Hikosaka and Shunji Ido (Saitama University)

For the acceleration of physical simulations, the parallelization is very useful method. Because an automatic parallel compiler is not well developed, it is necessary to develop parallelization techniques and programms for solving problems in high parallelization system. Here we used transputer T800 and i80860 as microprocessor in MIMD type parallel calculation system.

In this report, we examined benchmark tests and physical simulations in following methods.

- (1) a program to transfer data,
- (2) a program to solve two-dimensional thermal conduction (98%),
- (3) a program to solve two-dimensional random walk problem (97%),
- (4) a program to simulate an ion beam extraction system two-dimensionally (97%),

In the calculation of transputer, we get rates of parallelization as shown in (). We report these results of calculation in iPSC/860.

Keywords (physical simulation, parallelization, transputer, iPSC/860)

Personal Super/Parallel Computing By Using Transputers and Intel 80860

Parallel Simulations By Using Transputer

Parallel C and Parallel FORTRAN were used to program parallel simulation codes, where OCCAM was not used because it was not familiar to users in the areas of numerical calculations.

Parallelism is typically useful to shorten the calculation time. But, in any case, the ratio of parallelized part must be close to unity infinitely, to increase the acceleration factor. The relation between the parallel ratios and acceleration can be easily shown in Andahl's law. Because the parallel ratio depends on each problem, we must estimate and examine it for each actual problem to be solved. Here the authors will show three types of problems.

The first problem is an initial value problem of a electron bunching simulation (EBS), where electron beam bunching will be calculated with time evolution. 73 groups of electrons those have different phase to the traveling wave are divided to a number of transputers and the motions are traced by them separately. At the end of each time step, the data of position of each electron group are gathered to one transputer and the space charge effect is calculated, when parallel scheme is broken during that calculation.

The second is a Monte Carlo Simulation (MCS). Where different series of random number are generated in each processor (T800) and a number of particle are traced independently. In the simulation, calculation on each particle can be done in parallel scheme. The parallel ratio is almost 99.9% and almost linear increase of acceleration can be obtained as shown in Fig.1.

Summary Transputer (T800) and 64 bit RISC Intel 80860 (1860) can be used as an accelerator added on a personal computer. When we use 32 bit T800 or 1860, scientific calculations will be carried out several ten times faster than in the case of using 32 bit personal computers or UNIX workstation. These added CPU can be used in a parallel connection system. The results of bench mark tests and examples of physical simulations using T800 and 1860 will be reported here.

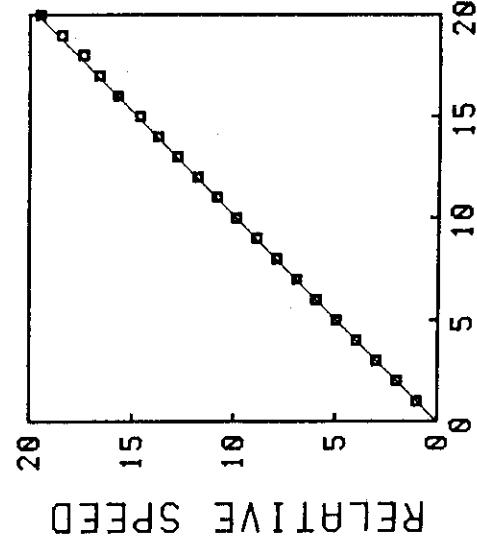
Personal Supercomputing and Parallel Computing

By using recent powerful microprocessors, we can carry out the large scale numerical simulations, for those supercomputers or mini supercomputers were needed a few years ago. Here the authors will report on the simulations by using transputers and 64 bit RISC (Intel 80860).

Transputers used here is T800 which is a 32 bit CPU including 64 bit FPU and 4 KB SRAM on a chip. Transputers were designed to be used in a parallel system, where each transputer is connected with others by serial links with the speed of 10 or 20 Mbps. By using these links, users can easily build a multi processor system of MIMD (Multi Instruction Stream and Multi Data Stream).

Intel 80860 (1860) is a 64 bit RISC, which is not compatible to the micro processor family of 8086 (8086, 80286, 80386, 80486) used in a personal computers, but it shows very powerful calculation performance.

Typically, transputer (T800) shows about 10 times higher speed than a 32 bit personal computers (80386/80387), and 1860 shows about 10 times higher speed of T800, then about 100 times higher speed of 80386/80387. In our bench mark test by using Linpack program, T800 (20MHz) shows the speed of about 0.4 MFLOPS and 1860 shows that of greater than 3 MFLOPS. Therefore, in a rough estimation, 20 transputers parallel system is almost similar to a large scale conventional computer such as Hitachi M680H or a non-vectorized supercomputer. And 1860 shows almost half to the above large conventional and expensive computer system. The authors will describe the actual cases of simulation carried out by using a parallel transputer system and an add-on CPU of 1860.



NUMBER OF TRANSPUTERS

Fig.1 Relative calculation speed as a function of the number of transputers

The third problem is a FEM (Finite Element Method) problem, which can be used in an electro-magnetic analysis. For this problem, the parallel ratio is relatively very low, almost 77%. Therefore, acceleration of the calculation speed is saturated when more than five parallel microprocessors (T800) were used.

Personal Supercomputing By Using 1860

Two types of 1860 systems were used in our actual simulations. In one system, 1860 is used a vector unit for 32 bit microprocessor, where MS-FORTRAN is used with a vector library. In another system, a cross compiler is supported by a personal computer. After being compiled and linked, an executable program is transferred to an 1860 board installed in a personal computer and executed by 1860.

In the first system, only when vector subroutines are called by a FORTRAN program on 80386/80387 CPU, 1860 will be used in a vector calculation. Therefore, the length of vector essentially decides the calculation speed. Figure 2 shows the calculation time as a function of the total size of 2-dimensional variables. 1860(1) means an algorithm, where one-dimensional elements of two-dimensional variables are processed by one vector calculation. In another case of 1860(2), the total variables of 2-dimensional elements are processed in one vectorized calculation. As shown in Fig.2, the increase of vector length results in the increase of acceleration.

S t e p = 5 0 0

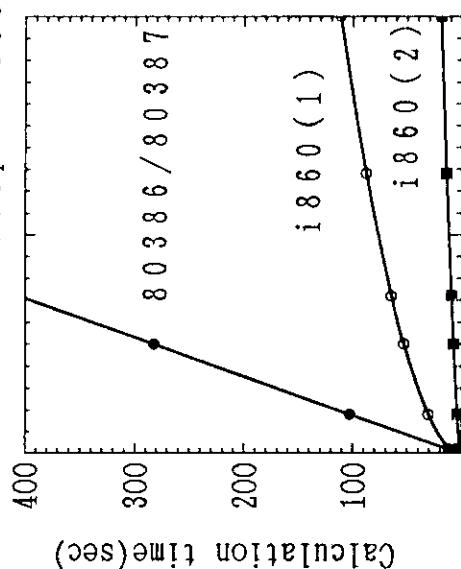


Fig.2 Calculation time to solve a 2-dimensional simulation as a function of the number of meshes

In the second system, a program on a 1860 board must be linked to a program executed by 80386/80387, where communication between the two programs must be written explicitly. However, a program developed on a personal computer can be easily rewritten with very few modification.

Results and Discussions

Calculation times is summarized in Table 1 for the three problems of EBS, MCS and FEM, where sequential program is carried out by a 32 bit personal computer, a 32 bit UNIX workstation, single T800 and 1860 of the above second system. The parallel ratio of EBS, MCS and FEM is 97%, 99.9%, 77% respectively. Therefore the acceleration performance of the calculation is different in these three problems, as shown in Table 1. The authors would like to summarize the results of Table 1 as follows.

- (1) 1860 is an excellent CPU for numerical calculations, which can be used in a personal system, and almost similar to the first Cray.
- (2) Transputers are very useful in a large scale simulation when a program was well parallelized. Sometimes, more than 10 transputers show the more powerful calculation potential than 1860.

Conclusions

Both a parallel microprocessor (T800) system and 1860 in a personal computer give us the calculation environment almost similar to a supercomputer a few years ago. Those systems are usefully and actually used in physical simulations to generate the meaningful results.

Type	Computer	CPU	Calculation Time [sec]		
			EBS	MCS	FEM
Sequential	PWR-70	30386/80387	2289	8354	51
	LUNA	6030/68881	1227	4239	24
	Transputer	T800-20	1050	397	2
Parallel	i860/PWR-70	180860	157	1349	5088
			1	701	2512
	Transputers	T800-20 x 6	252	853	10
			20	113	251
			22	103	

Table 1. Calculation time for three problems

12.5 標準遮蔽計算コードシステムINTEL-BERMUDAの開発

長谷川 明（日本原子力研究所）

放射線遮蔽計算を必要としている分野は、原子炉の設計のみならず原子力の全分野にわたっている。現在の遮蔽計算は、使用する計算コード及び核データについてのきわめて高度の知識が必要とされ、計算者により結果に一桁以上の差が出ることも希ではなく、一般の利用者が簡単に使える状況ではなくなりつつある。原研では、これに対応して、一般の利用者が簡単にかつ正確に複雑な遮蔽計算を行う事が出来るシステムとして、INTEL-BERMUDAの開発を現在進めている。本システムは、A/I, C AD, CG 等の最新の計算機技術を取り入れ、高精度遮蔽計算コードであるBERMUDAをはじめANISN, DOT, MCNPといった広く利用できる遮蔽計算コードを最新の核データを用いて簡便に行う事が出来る様にするものであり、エンジニアリング・ワーク・ステーション上で入力、出力の支援を行い、実際の大規模計算は大型機上で実行させる形を取っている。現在要素技術の開発が主であるが、最終的には統合化された遮蔽設計計算環境としての完成を目指している。本システムの開発の現状を紹介する。

Development of a Standardized Shielding Calculation Code System: INTEL-BERMUDA

Akira HASEGAWA (JAERI)

The area in need of shielding calculations covers all of the nuclear application fields in the nuclear industries as well as reactor design calculations. Shielding calculation requires highly advanced knowledge both of so sophisticated transport codes and deep insights of nuclear data. It is not so seldom to get large different results (in scales of order) using the same code depending on the selection of different group structure, problem modelings, proper choice of options, etc., for the users other than the specialists. These codes are becoming difficult to use to get good results for the users. In JAERI, we started a project to standardize and consolidate the shielding calculation codes to integrating the standard shielding calculation system named INTEL-BERMUDA, so as to use easily and accurately for all users. We targeted the following transport codes as to the integration, BERMUDA: high precision integral transport code, ANISN/DOT: most popular deterministic Sn code, MCNP: simulated point-wise energy Monte Carlo code. This system is constructed on the basis of EWS(Engineering Work Station Sun-4) and Main-Frame/Super computers (FACOM-M/780 & VP-2600). In EWS, pre/post-processing such as input preparation and output processing adopting most advanced computational technologies in today such as A/I (Artificial Intelligence), CAD (Computer Aided Design), CG (Computer Graphics) to extend the user friendliness. For heavy computation works such as transport calculation are performed mainly by main-frame/super computer due to the shortage of computation powers of EWS. The project is now going on, we are in the development stage of key components. Current development status is shown in this presentation.

keywords (shielding, transport code, BERMUDA, ANISN, DOT, MCNP, INTEL-BERMUDA)

12.6 知的原子力設計システム（IRDS）の開発

久語 輝彦, 土橋敬一郎, 中川 正幸, 森 貴正 (原研)

原子炉の概念設計を効率的に支援するための知的原子力設計システム（IRDS）の開発を行っている。本システムは主に炉心設計の分野を対象とし、核、熱水力及び燃料設計を含む。マンマシンインターフェースを充実させるため、ウインドウシステムや画像処理が容易に利用できるエンジニアリングワークステーション上で構築される。階層構造をもつ設計情報は”設計モデル”にオブジェクト指向的に記述した。本システムでは、”設計モデル”を中心にプロセスが稼働し、この”設計モデル”は参照や保存のために設計データベースに収容される。シミュレーションコードは、ワークステーションにモジュールとして移植された。また時間の要する計算は、メインフレームを利用する。試作段階ではあるが、効率的な設計支援のために、専門家の知識や経験の導入も行っている。

Development of Intellectual Reactor Design System(IRDS)

T.KUGO, K.TSUCHIHASHI, M.NAKAGAWA, T.MORI (JAERI)

The interectual reactor design system (IRDS) is developed to support reactor conceptual design studies. The system includes the fields covering the reactor core designs, such as neutronics, thermohydraulics, fuel design and so on. To fulfil man-machine interface, the system has been developed on an engineering workstation which can easily utilize a window system, graphics and so on. Design information which has a hierarchy structure is described in "design model" with object-oriented architecture. Design process is driven around the design model which is compiled into design data base to provide the reference data and save the data updated. Some simulation codes are converted into modules on a workstation. However, the simulation codes which need much computation time can be executed in the mainframe. To efficiently support the design, the knowledge and experience of experts are transferred into knowledge base which is now in testing stage.

Keywords: IRDS, Module, Neutronics, Thermohydraulics, Fuel Design, Engineering Workstation, Man-machine Interface

12.7 ベクトル化モンテカルロコードの開発と応用

森 貴正, 中川 正幸(原研), 佐々木 誠(日本総研)

汎用ベクトル化モンテカルロコードGMVP(多群)とMVP(連続エネルギー)を開発した。両コードはスタック駆動ゾーン選択法によってベクトル化されており、95%以上のベクトル化率が達成されている。幾何形状モデルは四方及び六方の多重格子と組合わせ形状(combinatorial geometry)によって表現される。格子形状表現は全炉心のような複雑形状の表現を容易にするばかりでなく、高いベクトル化効率の実現にも有用である。多群コードは P_1 展開型断面積とともに二重微分型断面積も使用でき、連続エネルギーコードでは評価済核データに与えられている全ての中性子反応を陽に取り扱うことによって精度の高い輸送計算を実現している。

ポスターでは、ベクトル化の方法、コードの機能と性能評価の結果、ベクトル化コードの異種スーパーコンピュータへの移植性、及び実用問題への応用例について発表する。

Development of Vectorized Monte Carlo Codes and Their Applications

T. MORI, M. NAKAGAWA (JAERI), M. SASAKI(Nihon soken)

Vectorized Monte Carlo codes GMVP (multi-group) and MVP (continuous energy) have been developed for general purpose use. The vectorization algorithm is based on the stack-driven zone-selection method, and a vectorization ratio of more than 95 % is achieved by both codes. A geometry model is described by the combinatorial geometry together with multi-level square or hexagonal lattices which are quite useful to obtain a high gain by vectorization as well as to treat easily complicated geometry such as whole reactor cores. GMVP can use both the P_1 expansion type and the double-differential form cross sections. MVP treats all neutron reactions given in an evaluated nuclear data library based on the continuous energy method.

In posters, we present the vectorization method, capabilities of the codes, the results of performance evaluation, the portability of vectorized codes between different supercomputers and some examples of practical uses.

Keyword: vectorization, Monte Carlo code, particle transport, multi-group, continuous energy, speed up, portability, application, general purpose use

13. 閉会の辞

炉物理研究委員会
原子炉システム専門部会長
石黒 幸雄

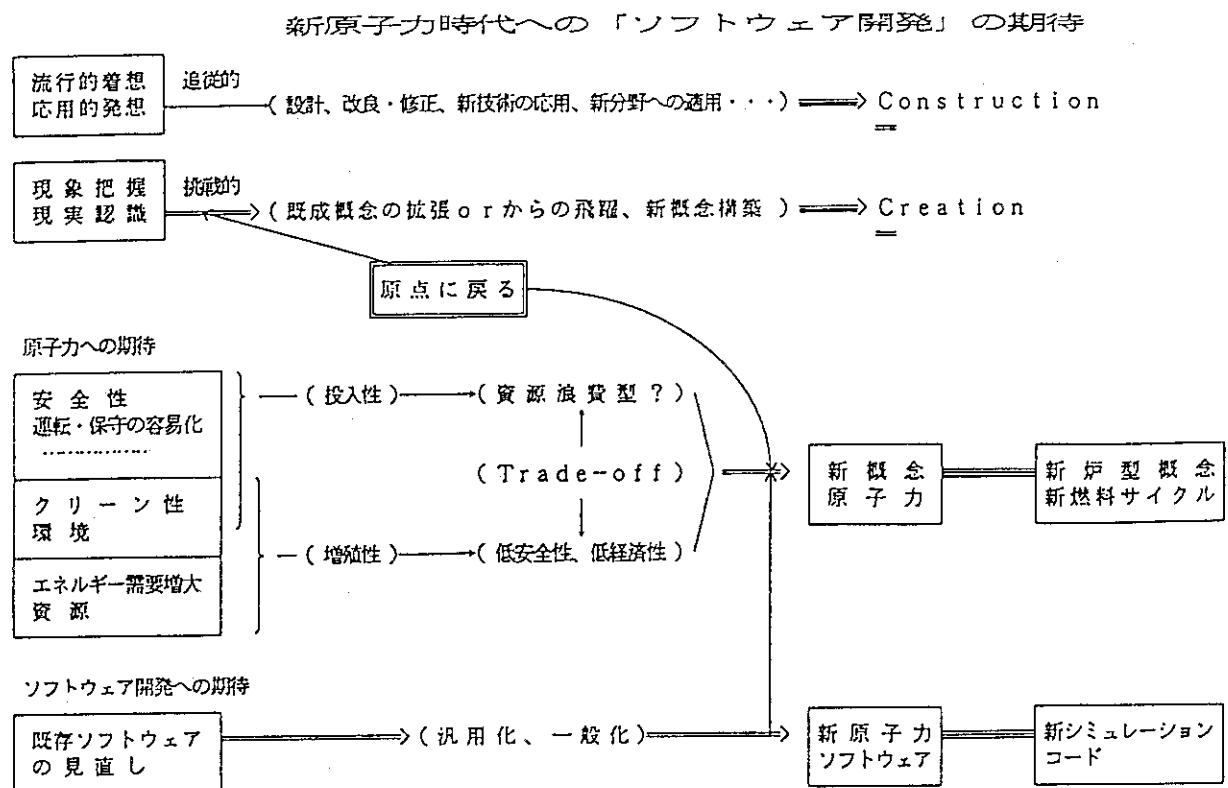
金子 義彦 炉物理研究委員会委員長が人事異動により10月1日をもって原研東京本部の企画室長に就任されました。本日、ご多忙とのことで、私が代わって挨拶させて頂きます。本来、ここでの挨拶のような場合には、客観的なことを申し上げるべきであります。しかし、最近の原子力の動向に関連しまして「ソフトウェア開発」に対する私自身の個人的な期待を述べさせて頂きまして、挨拶とさせて頂きます。

近年、TMIやチェルノブイル等の事故を契機として、安全性、運転・保守の容易化、クリーン性等が強く要望されており、世界的にも原子力の研究開発は「投入性向上」を目指しているようでございます。しかし、投入性のみを追求することは、長期的に見ますと、資源浪費型原子力システムに繋がりかねません。

一方、温室効果等地球環境問題からクリーンな電気に対する要望、さらに開発途上国のエネルギー需要が、延いては原子力に対する期待が明らかに高まりつつあります。これらの状況のもとでは、長期的に核燃料需要が逼迫する恐れがあります。このような問題に対処するために、「核燃料増殖システム」を実用化する開発努力も強く望まれております。

残念ながら、我々は過去の経験から、「増殖性」を追求すると常に「低安全性」、「低経済性」に導かれる事を知っています。従って、原子力が世界的に停滞しているこの時期こそ、原点に立ち戻って原子力開発の方策を見直すことが必要であると思われます。即ち、投入性と増殖性とをどのように調和させつつ炉型戦略を確立していくかを、再考すべき時期ではないかと考えられます。

このような目的のためには、核燃料サイクルも含めて幅広く炉型を検討することは不可欠で、一般性の一層高いソフトウェアの存在が前提とされます。この意味で、既存ソフトウェアの見直しを行って、これら的一般化、汎用化を進めることは、新原子力時代を切り開くには大変有意義ではないかと考えております。



付録 研究会出席者名簿

原子力コード研究委員会・炉物理研究委員会 合同研究会 所外出席者名簿(順不同)

氏名	会社名	氏名	会社名
Prof. Claudio Rebbi	Boston Univ.	宮下 精二	京都大学
植木 紘太郎	運輸省船舶技術研究所	錦織 賢夫	大阪大学
金城 勝哉	動力炉核燃料開発事業団	石黒 美佐子	茨城大学
芳賀 暢	原子炉工学試験センター	久保田 昌晴	埼玉大学
加藤 隆夫	高エネルギー物理学研究所	彦坂 聰	"
近藤 悟	動力炉核燃料開発事業団	河村 洋	東京理科大学
石川 眞	"	佐竹 信一	東京理科大学
植之原 雄二	(株)東芝	岩田 修一	東京大学
川合 将義	(株)東芝	安藤 孝二	"
瀧川 幸夫	(株)東芝	小寺 廉一	東京都渋谷区神山町4-14 F2P
小池 誠彦	日電	塩田 和則	東京都台東区今戸2-38-2
栗田 康男	I B M	能澤 正雄	(財)原子力データセンター
清水 茂則	I B M	藤村 統一郎	"
浅川 和雄	(株)富士通	飯島 勉	"
梅谷 征雄	日立	山崎 和彦	"
三竹 晋	原子力安全解析所	小澤 保孝	"
高橋 千里	(株)石川島播磨重工業	佐々木 健	"
山野 直樹	(株)住友原子力工業(株)	成井 正夫	"
奥田 基	(株)富士通	内藤 新司朗	"
小山 隆	(株)日本コンパックスコンピュータ		
山本 章夫	(株)原子燃料工業		
真木 紘一	(株)日立製作所エネルギー研究所		
内川 貞夫	(株)日立製作所エネルギー研究所		
宿谷 弘行	(株)日本総合研究所		
中村 寿	(株)日本原子力発電		
長谷川 正	(株)三菱原子力工業		
田原 義和	"		
中田 哲夫	(株)川崎重工業		
辻 政俊	(株)東洋エンジニアリング		

付録 研究会出席者名簿

原子力コード研究委員会・炉物理研究委員会 合同研究会 所外出席者名簿(順不同)

氏名	会社名	氏名	会社名
飯泉 仁	東海研究所	石黒 幸雄	原子炉工学部
佐野川好母	大洗研究所	山下 栄一	"
浅井 清	企画室	柴田 勝成	"
横田 光史	固体物理第2研究室	中原 康明	"
別役 広	固体物理第3研究室	篠原 慶邦	炉制御研究室
谷 啓二	臨界プラズマ第2実験室	八木 秀之	"
竹田 辰興	理論解析研究室	篠原 慶邦	原子炉制御研究室
内藤 敦孝	核燃料施設安全評価研究室	島崎 潤也	"
奥野 浩	"	藤井 義雄	"
高野 誠	"	石川 信行	"
増川 史洋	"	前川 洋	核融合炉物理研究室
中島 豊	核データセンター	小迫 和明	"
市原 晃	"	前川 藤夫	"
本間 功三	反応度安全研究室	土橋 敬一郎	原子炉システム研究室
花島 進	加速器管理室	森 貴正	"
小川 益郎	高温熱工学研究室	中川 正幸	"
山本 浩康	同位体分離研究室	久語 輝彦	"
水本 元治	加速器工学研究室	奥村 啓介	"
横堀 仁	"	森本 裕一	"
小栗 英知	"	長谷川 明	遮蔽研究室
迫川 邦俊	"	鈴木 友雄	"
長谷川 和男	"	田中 俊一	"
美濃 浩志	"	小手川 洋	"
村田 裕彦	"	笹本 宣雄	"
白石 忠男	材料試験炉部 計画課	秋濃 藤義	炉物理実験研究室
土田 升	"	西田 雄彦	"
富樫 喜博	W A S T E F 課	大崎 浩	再処理研究室
阿部 豊	伝熱流動研究室	後藤 賴男	炉修研究室(嘱託)
栗原 研一	核融合装置試験部 J T - 60	伊勢 武治	むつ 原子力船技術部
有賀 武夫	照射解析研究室		

付録 研究会出席者名簿

原子力コード研究委員会・炉物理研究委員会 合同研究会 所外出席者名簿（順不同）