

J A E R I - M
94-026

ワークステーション版
REFLA/TRACコードの開発

1994年3月

大貫 晃・秋本 肇・村尾 良夫

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。
入手の問合わせは、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、
お申し込みください。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, Tokaimura, Naka-gun, Ibaraki-
ken 319-11, Japan.

© Japan Atomic Energy Research Institute, 1994

編集兼発行 日本原子力研究所
印 刷 ニッセイエプロ株式会社

ワークステーション版 REFLA / TRAC コードの開発

日本原子力研究所東海研究所原子炉工学部

大貫 晃・秋本 肇・村尾 良夫

(1994年1月31日受理)

REFLA / TRAC コードは、軽水炉の安全評価解析コードの検定並びに事故解析、新型軽水炉の事故解析や設計に用いることを目的とした最適予測コードである。そのため、高い予測精度が要求され、予測精度を左右する各種物理モデルの評価が重要となる。REFLA / TRAC コードの3次元モデルの評価を行う場合、大型計算機により行っているが、出力結果を得るまでのターンアラウンド時間が長く、3次元モデルの評価を効率よく行うことが困難であった。そこで、大型計算機と同程度の高速な計算が可能で、多次元の図形出力機能に優れたエンジニアリングワークステーション(EWS)上で稼働するREFLA / TRAC コードを開発した。併せて、時系列プロット機能の整備、及び2次元図形表示機能の開発を行い、3次元計算の解析を容易に行えるようにした。今後、EWSの計算速度の高速化、図形表示機能の拡充により、より一層評価効率が向上するものと期待できる。

本報告書では、EWS版REFLA / TRAC コード、時系列プロット機能及び2次元図形表示機能の各概要について述べる。

Development of REFLA/TRAC Code for
Engineering Work Station

Akira OHNUKI, Hajime AKIMOTO and Yoshio MURAO

Department of Reactor Engineering
Tokai Research Establishment
Japan Atomic Energy Research Institute
Tokai-mura, Naka-gun, Ibaraki-ken

(Received January 31, 1994)

The REFLA/TRAC code is a best-estimate code which is expected to check reactor safety analysis codes for light water reactors (LWRs) and to perform accident analyses for LWRs and also for an advanced LWR. Therefore, a high predictive capability is required and the assessment of each physical model becomes important because the models govern the predictive capability. In the case of the assessment of three-dimensional models in REFLA/TRAC code, a conventional large computer is being used and it is difficult to perform the assessment efficiently because the turnaround time for the calculation and the analysis is long. Then, a REFLA/TRAC code which can run on an engineering work station (EWS) was developed. Computational speed of the current EWS is the same order as that of large computers and the EWS has an excellent function for multidimensional graphical drawings. Besides, the plotting processors for X-Y drawing and for two-dimensional graphical drawing were developed in order to perform efficient analyses for three-dimensional calculations. In future, we can expect that the assessment of three dimensional models becomes more efficient by introducing an EWS with higher calculational speed and with improved graphical drawings.

In this report, each outline for the following three programs is described:

- (1) EWS version of REFLA/TRAC code,
- (2) Plot processor for X-Y drawing and
- (3) Plot processor for two-dimensional graphical drawing.

Keywords: Reactor Safety, TRAC Code, EWS, Numerical Simulation, Graphics, Two-phase Flow, Heat Transfer

目 次

1. 序	1
2. FACOM版 REFLA/TRACコードのEWS版への変換	2
2.1 EWS版作成に必要な変更内容	2
2.2 EWS版 REFLA/TRACコードのファイル構成	4
2.3 チェック計算	4
3. FACOM版時系列プロット機能のEWS版への変換	10
3.1 グラフィックファイル変換プログラム	10
3.2 時系列プロットプログラム	10
4. 2次元図形表示機能の開発	13
4.1 ポストプロセッサとしての2次元図形表示プログラムの開発	13
4.2 計算と同期して2次元図形表示を行うプログラムの開発	16
5. 結 言	27
謝 辞	27
参考文献	28
付 録 A FACOM版からEWS版へ変換する際の変更内容	29
付 録 B EWS版 REFLA/TRACコード操作マニュアル	35
付 録 C 2次元図形表示システム操作マニュアル	41

Contents

1. Introduction	1
2. Conversion of REFLA/TRAC Code from FACOM Version to EWS Version	2
2.1 Items to be Changed for EWS Version	2
2.2 File Configuration of REFLA/TRAC Code EWS Version	4
2.3 Check Calculation	4
3. Conversion of Plot Processor for X-Y Drawing from FACOM Version to EWS Version	10
3.1 Graphic File Conversion Program	10
3.2 Plot Program for X-Y Drawing	10
4. Development of Plot Processor for Two-dimensional Graphical Drawing	13
4.1 Development of Plot Program for Two-dimensional Graphical Drawing as Post-processor	13
4.2 Development of Cooperative Two-dimensional Graphical Drawing with Calculation	16
5. Conclusions	27
Acknowledgments	27
References	28
Appendix A Items to be Changed for Conversion from FACOM Version to EWS Version	29
Appendix B Usage Manual of REFLA/TRAC Code EWS Version	35
Appendix C Usage Manual of Plot System for Two-dimensional Graphic Drawing	41

1. 序

REFLA/TRACコードは、米国で開発された原子炉過渡解析3次元コードTRAC-PF1⁽¹⁾を骨組みとし、再冠水時の炉心熱水力モデル、凝縮モデル、圧力損失計算モデル等の物理モデルを改良した最適予測コードである。⁽²⁾⁻⁽⁴⁾ 同コードは、軽水炉の安全評価解析コードの検定、事故解析、事故シミュレーション、新型軽水炉の事故解析や設計に用いることを目的としている。そのため、高い予測精度が要求され、予測精度を左右する各種物理モデルの評価が重要となる。

REFLA/TRACコードの3次元ベッセルコンポーネントの評価を行う場合、大型計算機でのバッチジョブによる計算、及び大型計算機で利用可能な多次元の図形出力により行っている。両作業とも多数のユーザーと並行して行われるため、出力結果を得るまでのターンアラウンド時間が長く、コンポーネントの不具合箇所の修正及びモデルの評価を行う効率が悪い。近年、エンジニアリングワークステーション(EWS)の進歩は著しく、大型計算機と同程度の高速な計算ができるようになってきている。そのため、マシンを占有した場合、大型計算機よりもターンアラウンド時間を短縮できるものと考えられる。また、EWSは多次元の図形出力機能も優れており、解析面でのターンアラウンド時間の短縮もはかれる。REFLA/TRACコードの3次元ベッセルコンポーネントの評価作業の効率を上げるためには、EWS上でREFLA/TRACコードを利用できるようにする必要がある。

近年、TRAC-PF1やRELAP5コード等のEWSバージョンが開発され、プラントアナライザー(NPA)として活用されているが、1次元のシステム計算を対象としたものがほとんどであり、3次元モデルの評価を効率的に行うための多次元図形出力機能を完備したものは少ない。

このような背景から、EWS上で稼働するREFLA/TRACコードを開発し、併せて、時系列プロット機能の整備、及び2次元図形表示機能の開発を行い、3次元計算の解析を容易に行えるようにした。なお、EWSとしてはSUN製SPARCstation 2 (SunOS4.1.2, 日本語Open Windows 2.0.1 Rev.C)を使用した。

本報告書では、EWS版REFLA/TRACコード、時系列プロット機能及び2次元図形表示機能の各概要について述べる。また、付録には日本原子力研究所大型計算機(FACOM M-780, 以下 FACOM)上で稼働しているREFLA/TRACコード及び時系列プロット機能をEWS版へと変換する際の変更点、及びEWS版REFLA/TRACコードの操作法(時系列プロット機能の操作法を含む)、並びに2次元図形表示機能の操作法に関するマニュアルを添付した。

2. FACOM版REFLA/TRACコードのEWS版への変換

2.1 EWS版作成に必要な変更内容

EWSとしては下記のシステムを使用した：

SUN SPARC station 2 (32MB main memory)

OS; Sun OS 4.1.2, Window; Open Windows 2.0.1 Rev. C,

FORTRAN Compiler; Sun FORTRAN 1.4

上記の計算機環境で正常な実行を可能にするため、FACOM版REFLA/TRACコードのいくつかのルーチンでソースレベルでの変更（それに伴うルーチン追加も含む）が必要であった。以下に変更箇所の概要を述べる。いくつかのものは2.3節で述べるチェック計算を行う過程で不具合が見つかり、修正したものである。変更点の詳細については付録Aにまとめた。

(1) 特種文字、16進表記の変更

EWSの文字表示が、ASCIIコードであるため、¥は無条件に\$に変換されてしまう。FACOM版REFLA/TRACコードでは¥、したがってEWSでは\$から始まるサブルーチン、コモン、変数が多数存在する。この記述はEWS上のFORTRAN77では許されないため、ソースファイルすべてに関して、\$から始まるサブルーチン名、コモン名、変数名について、先頭の\$をzに変更した。また、ファイル名では\$は使用できないため、ファイル名上のすべての\$を#に置き換えた。

FACOMとEWSでは16進表記法が異なるため、16進数を与えているデータ文に対し変更を行った。

(2) インクルード文の変更

REFLA/TRACコードの全てのソースファイルに関して、インクルード文をEWS用FORTRAN77に準拠する記述に変更した。

(3) ファイルアロケートの変更

EWS版REFLA/TRACコードでは、ファイルアロケートの方法に、EWS標準の事前結合のない手法であるOPEN/CLOSE文によるダイナミックアロケーションを用いた。

OPEN文において、装置番号とファイル名が1対1で対応するように、EWS版REFLA/TRACコード実行時の入出力ファイル名をソースプログラム上で固定とした。

(4) FACOMサービスサブルーチンの変更

FACOM版REFLA/TRACコードではFACOMサービスサブルーチンのDATE, TIME, CLOCKM, EXITを使用している。ここでDATE, TIME, CLOCKMに対して新たにdatex, timex, clockmなるEWS版サブルーチンを作成した。EXITに対しては、STOP文に置き換えた。

またFACOM版REFLA/TRACで使用するアセンブラルーチン¥¥LOCFは、z\$locfなるC言語記述の関数へ置き換えた。

(5) サブルーチン引き数の修正

親ルーチンと子ルーチンの、引数の長さ(バイト数)がサブルーチン間で統一する様、親ルーチン側 CALL文の引数を修正した。

(6) EQUIVALENCE文の修正

8バイトREALと4バイトINTEGERとのEQUIVALENCE文で、インクルードファイルに対して、先頭アドレスが規定出来るよう修正を行った。

(7) エントリー引数の修正

CALL文とENTRY文との間の引数の個数を合わせた。

(8) データ文の修正

ラベルドコモンCHGALPについて、コモン変数の初期化を、VOIDCK, NEWDLTサブルーチンの2箇所で行っていた。この記述はEWS上のFORTRAN77ではコンパイル時にエラーとなってしまう。従ってこれを改めて(コメント化)、VOIDCKのみで行うよう修正した。

(9) サブルーチンPACKITの修正

FACOM版REFLA/TRACコードのソースプログラムのサブルーチンPACKITの中で、新しい8バイト変数の前後4バイトずつに、異なる4バイト変数を代入する処理を行っている。EWS版FORTRAN77で識別できるよう修正した。

(10) サブルーチンREADI, READRの修正

サブルーチンREADI, READRに対するCALL文の引数の数は、読み込み変数の個数によって異なるため、CALL文の引数の数とサブルーチンREADI, READRの引数の数が異なるものが多く存在していた。そこで、CALL文とSUBROUTINE文引数の個数が常に同じになるように、引数の個数ごとに複数のサブルーチンを作成した。

(11) サブルーチンINPUT内におけるサブルーチンCLEARのCALL文の修正

サブルーチンINPUTでCLEARをCALLする場合、FACOM版REFLA/TRACコードのソースプログラムのコーディングのままでは、常に実行エラーが起きてしまった。そこで実行エラーを取り除くため、以下に示すような修正を行った：

修正前 CALL CLEAR(0, IA(1, LIJN)IT) → 修正後 CALL CLEAR4(0, IA(1, LIJN)IT*2)

(12) コモン文の修正

ラベルドコモンDCMDLを使用するサブルーチンについて、DCMDLをインクルード文として参照できるように変更した。

2. 2 EWS版REFLA/TRACコードのファイル構成

本節では、2. 1の変更により作成されたEWS版REFLA/TRACコードのEWSディスク内のディレクトリ構造、及び入出力ファイル構成について示す。

図1にEWSディスク内のディレクトリ構造を示す。EWS上のREFLA/TRACコード本体に関するファイルは、すべてディレクトリtrac内にある。

表1にEWS版REFLA/TRACコードの入出力ファイルの一覧を示す。各ファイル名は固定とした。実行方法の詳細については付録Bにまとめた。

2. 3 チェック計算

FACOM版からEWS版への変換が妥当に行われたか否かを調べるため、EWS上で計算を行い、得られた計算結果（リスト出力またはグラフ出力）をFACOM計算機による既存の計算結果と比較してチェックした。チェック計算は10ケース行った。各チェック計算の概要を表2に示す。

チェック計算の結果の概要を以下に述べる。

(1) Edwards Pipeの実験

FACOM版とEWS版の計算結果の相違は、以下のように無視できるほど小さかった。

- (a) 有効数字6桁まで出力される圧力の6桁目の数字が1だけ異なる場合が多い。
- (b) 有効数字4桁まで出力されるボイド率、気相温度、蒸気密度、液相流速、気相流速、の4桁目が希に1だけ異なる場合がある。

(2) CISE実験

FACOM版とEWS版の計算結果の変数値には相違がなかった。

(3) 東工大凝縮実験

冷水が注入された直後の振動する時間帯では、タイムステップサイズが異なり変数値に違いがみられた。振動をする計算では各計算機のシステムに依存する丸め誤差等の違いが影響し、変数値の違いが現れたものと考えられる。ほぼ定常状態が達成された時間帯では、圧力の有効数字5桁目が異なることと液相流速の有効数字3桁目が1だけ異なることを除いて一致していた。

(4) CORE単体（高温壁からの除熱計算）

FACOM版とEWS版の計算結果の変数値には相違がなかった。

(5) CORE単体（再冠水条件での計算）

100秒までのヒートアップ計算では、FACOM版とEWS版の計算結果の変数値には相違がなかった。再冠水開始後は各物理量が振動し、タイムステップ幅が両計算で異なったため、変数値に違いがみられた。振動をする計算では各計算機のシステムに依存する丸め誤差等の違いが影響し、変数値の違いが現れたものと考えられる。模擬燃料ロッドがクエンチした後は振動が小さくなり、両計算結果の違いは小さくなった。

(6) CORE単体（PWR液单相定常計算）

FACOM版とEWS版の計算結果の変数値には、0.005秒の炉出力の有効数字4桁目が1異なることを除いて相違はなかった。

(7) VESSEL単体（高温壁からの除熱計算）

再冠水モデルのコーディングの丸め誤差の処理により、I F文判定の分岐先がF ACOM版とEWS版とで異なることがわかった。I F文判定を修正することにより、F ACOM版とEWS版との間の違いはなくなった。

(8) VESSEL 単体 (再冠水条件での計算)

注水開始前は違いはなかった。再冠水開始後は各物理量が振動し、タイムステップ幅が両計算で異なったため、変数値に違いがみられた。振動をする計算では各計算機のシステムに依存する丸め誤差等の違いが影響し、変数値の違いが現れたものと考えられる。模擬燃料ロッドがクエンチした後は振動が小さくなり、両計算結果の違いは小さくなった。

(9) VESSEL 単体 (平板炉心再冠水試験)

EWS版での計算は正常に終了することを確認したが、リスト出力が膨大なため、リストでの比較は行わず、3章で述べる時系列プロット機能によりF ACOM版との比較を行った。

約150秒までのヒートアップ計算では、F ACOM版とEWS版の計算結果に大きな相違はなかった。再冠水開始後は各物理量が振動し、タイムステップ幅が両計算で異なったため、変数値に違いがみられた。ただし各物理量の平均値はほぼ一致した。振動をする計算では各計算機のシステムに依存する丸め誤差等の違いが影響し、変数値の違いが現れたものと考えられる。

(10) 1次元モデルによる再冠水システム計算 (円筒炉心再冠水試験)

EWS版での計算は正常に終了することを確認したが、リスト出力が膨大なため、リストでの比較は行わず、3章で述べる時系列プロット機能によりF ACOM版との比較を行った。

ヒートアップ計算中は、F ACOM版とEWS版の計算結果に大きな相違はなかった。再冠水開始後は各物理量が振動し、タイムステップ幅が両計算で異なったため、変数値に違いがみられた。ただし各物理量の平均値はほぼ一致した。振動をする計算では各計算機のシステムに依存する丸め誤差等の違いが影響し、変数値の違いが現れたものと考えられる。図2に比較例として炉心入口での水温(TL・・・)及び飽和温度(TS・・・)の比較を示す。上図がF ACOM版での結果であり、下図がEWS版でのものである。約90秒までがヒートアップ期間である。再冠水開始後約130秒までの時間帯では水温の変動が大きく、値にやや違いがみられるが、平均的には大きな違いはない。またそれ以降についても平均的な挙動は同一である。

いままで述べてきた各チェック計算の結果より、振動の激しい場合は各計算機のシステムに依存した違いが現れるものの、振動が小さく定常に近い計算ではF ACOM版とEWS版との違いはほとんど無いことがわかる。また振動している場合でも、その平均値に大きな違いはなかった。このことより、2.1節で述べた変更を施すことにより、F ACOM版REFLA/TRACコードはEWS版に妥当に交換されたものと考えられる。

表1 EWS版REFLA/TRACの入出力ファイル一覧

装置番号	ファイル名	ファイル形式	内 容	備 考
1	TRACIN	カードイメージ	入力データファイル	必 須
2	TRCOUT	カードイメージ	プリンタ出力ファイル	リスト出力
5	TRCINP	カードイメージ	入力ワークファイル	ワーク
7	TRCMSG	カードイメージ	メッセージファイル	メッセージ出力
11	TRCGRF	バイナリー	グラフィックファイル	グラフィック用出力
12	TRCDMP	バイナリー	計算結果ダンプファイル	ダンプ出力
13	TRCRST	バイナリー	リスタート計算用入力ファイル	オプション(TRCDMPをRENAMEして作る)
15	TRCSO5	カードイメージ	SO Edit*用入力ファイル	オプション
16	TRCSO6	カードイメージ	SO Edit用出力ファイル	オプション
61	TRCDBG	カードイメージ	デバッグ用ファイル	デバッグ時の出力ファイル
90	TRCF90	カードイメージ	凝縮モデル用入力ファイル	オプション
99	TRCF99	カードイメージ	計算時間制御用ファイル	オプション
6	-----		TTY用ファイル	モニター上に出力される

* : 任意の物理量を出力するためのユーティリティ

表2 チェック計算の概要

コンポーネント単体のチェック	コンポーネント名	計算の内容	チェック対象となる主な物理パラメータ			
			壁面せん断力	界面せん断力	壁面熱伝達	界面熱伝達
	PIPE	Edwards Pipe Blowdown実験 ⁽⁶⁾ (セル数200、0.5秒の過渡計算)	○	○		○
		CISE実験 ⁽⁶⁾ (断熱蒸気・水2相流、セル数20)	○	○		
	TEE	東工大凝縮実験 ⁽⁷⁾ (セル数15、20秒の過渡計算)		○		○
	CORE	高温壁からの除熱計算、2ステップのみの計算			○	
		再冠水条件での計算 (100秒ヒートアップ、100-200秒注水)		○	○	
		PWR定格運転時の液単相定常計算 (60秒の解析)	○		○	
	VESSEL	高温壁からの除熱計算(1次元)、2ステップのみの計算			○	
		再冠水条件での計算(1次元) (初期高温壁、1-10秒注水)		○	○	
		平板炉心再冠水試験(2次元) ⁽⁸⁾		○	○	
システム計算のチェック	CORE, TEE PIPE, SG PLENUM	円筒炉心再冠水試験 (1次元コンポーネントによるシステム計算) ⁽⁹⁾	○	○	○	○

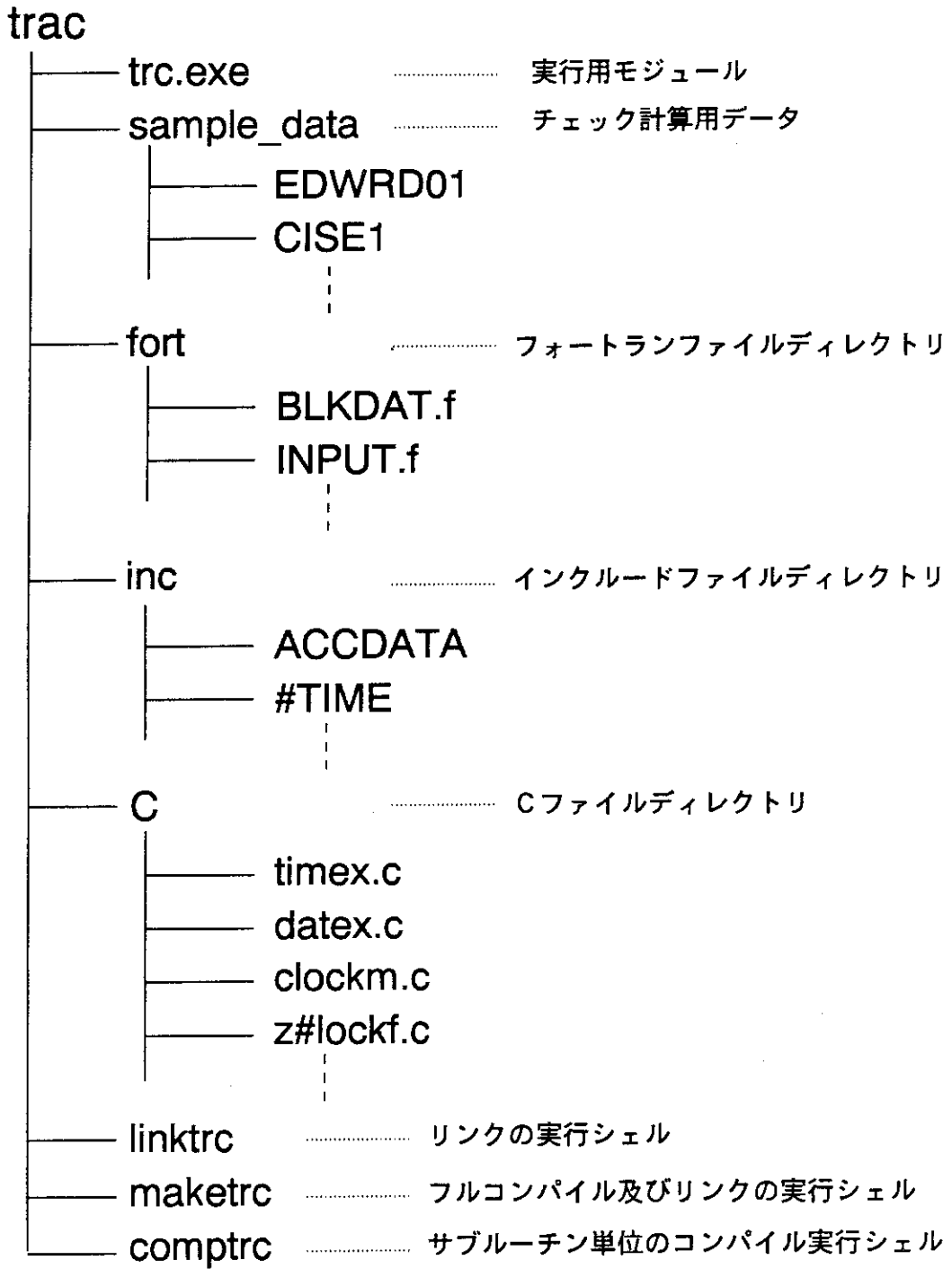
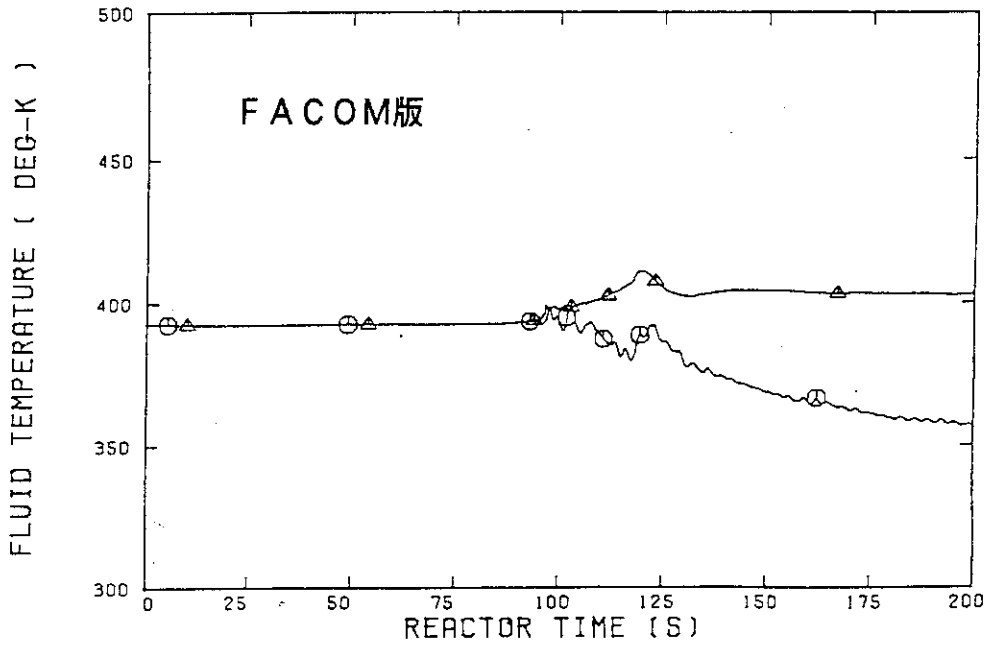


図1 EWS版REFLA/TRACコードのディレクトリ構造

CORE FLUID TEMPERATURE (INLET) CCTF62C2

○---(TIME (62) - TL1102 (62)) △---(TIME (62) - TS1102 (62))



CORE FLUID TEMPERATURE (INLET) CCTF62C2

•---(TIME (1) - TL1102 (1)) +---(TIME (1) - TS1102 (1))

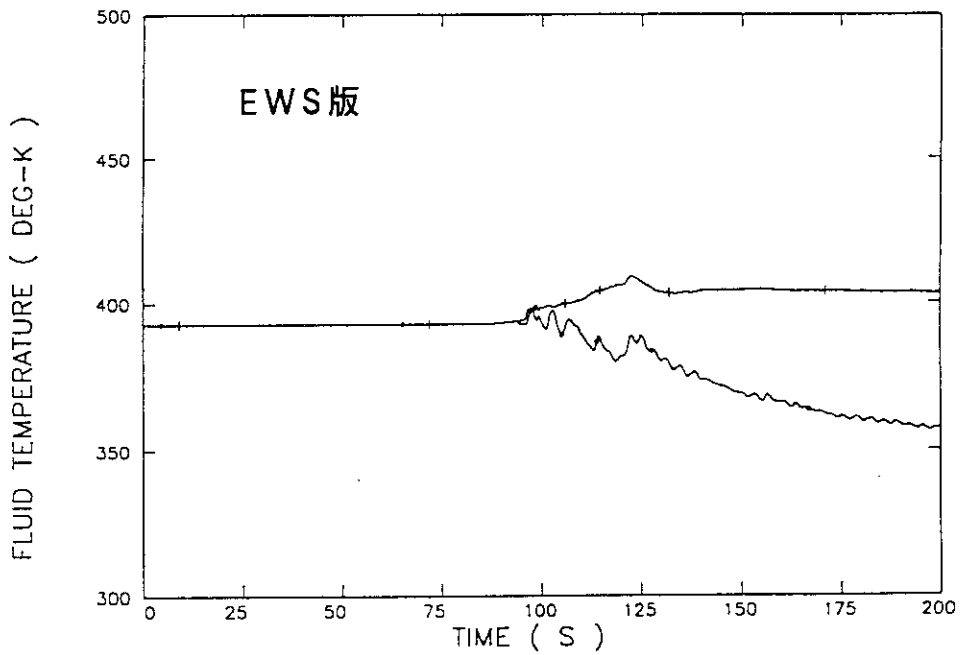


図2 炉心入口での水温及び飽和温度の比較

3. FACOM版時系列プロット機能のEWS版への変換

本章では、EWS版REFLA/TRACコードでの計算結果の時系列プロット図を作成するため、FACOM上にある既存のプロットプログラムを、EWS上で使用できるように変換した内容について述べる。このプロットプログラムは、REFLA/TRACコードの特殊なグラフィックファイル形式を変換するプログラム(CONV)と、これにより変換したファイルを入力データとして作図を行うプログラム(I PLOT)からなる。以下、3. 1節ではEWS版CONVを作成した際の変更点の概要を、3. 2節ではEWS版I PLOTを作成した際の変更点の概要を述べる。各変更内容の詳細については付録Aにまとめた。また、各機能の実行方法については付録Bにまとめた。

3. 1 グラフィックファイル変換プログラム

REFLA/TRACコードが出力するグラフィックファイルは、特殊な圧縮形式を採用しており、そのままではFACOM上の既存の図形出力プログラムは利用できなかった。そこで、FACOM上のシステムではこのグラフィックファイルの変換を行うツールを整備した。EWS版REFLA/TRACコードにおいてもこのツールは必要であり、EWS上への移植を行った。

移植にあたり大きく2つの変更を行った。以下にその概要を述べる。

(1) ファイルの入出力に対する修正

対話形式(インタラクティブ)に各種ファイル名を入力できるように、入出力ルーチンの修正を行った。対話形式で打ち込むファイルは以下の4種類である：

- (a) *Message Dump FILE*； 計算された各物理量の識別記号等が書かれたファイル、
- (b) *Input Data FILE*； 変換用の入力データ(このデータにより変換する内容(コンポーネント、時間領域、被覆管温度の高さ等)が選定できる)、
- (c) *TRCGRF FILE*； 計算により出力されたグラフィックファイル、
- (d) *Plot FILE for I PLOT*； 変換後の出力を格納するファイル。

(2) サブルーチンRDATAの修正

タイムエディットデータ(主に各種物理量が格納されている領域)を読み込む際のデータ変換に対して、FACOM版では8バイト変数の始めの4バイトを取り出して4バイト変数で使用しようとしていた。EWSでは不具合が生じるためサブルーチンRDATAの修正を行った。

3. 2 時系列プロットプログラム

FACOM上のシステムでは3. 1節でのファイル変換を行った後、REFLA/TRACコード用に整備したI PLOTシステムで時系列データの図形出力を行っている。EWS上においても変換されたプロットファイルを用いて、時系列データの作図が行えるようにした。作図プログラムとしては、FACOM版のI PLOTシステムをEWS上に移植したものを使うこととした。FACOM版のI PLOTシステムは、ファイルハンドリング関連のルーチンがアセンブラで記述されているが、それ

らの機能の書き換えを行った。また、FACOM版のI PLOTシステムは、FACOM用カルコンプルーチンを使用して、プロッター出力をしている。EWS版では、FACOMカルコンプルーチンの代わりに、既存のプログラムFORCAL/G⁽¹⁰⁾のライブラリーを使用した。

FORCAL/Gは、カルコンプ互換のライブラリーを持っており、このライブラリーをリンクすることにより、EWS上でX-Windowシステム対応のディスプレイへの出力、並びにPostScript形式に代表される各種図形処理プリンター言語のファイルを出力することができる。2章の図2下図はPostScript形式で出力した例である。

以下に、移植にあたり変更した内容の概要を述べる。

(1) サブルーチンのダミー化

FACOMアセンブラルーチン、特殊なカルコンプルーチン、算術ライブラリー等、不用ないしはFORCAL/Gにその機能の無いものについてダミールーチンとした。

(2) 特殊文字、16進表記の修正

2.1節(1)と同様の修正を行った。ここでは、\$から始まるサブルーチン名、コモン名、変数名について、先頭の\$の前にxを付けた。また、ファイル名でのすべての\$は_に置き換えた。

(3) オープン文の修正、追加

EWSでは、OPEN文においてACTION指定子、及びSTATUS指定子の'SHR'は使用できない。ここで、STATUS='SHR'は、STATUS='UNKNOWN'に変更し、ACTION指定子は、すべて削除した。

また、FACOM版ではサブルーチンOPENの中で、ファイルのアロケートをアセンブラルーチン呼び出して行っていたが、EWS版では、ここに新たにOPEN文を付け加えて、ファイルのアロケートを行えるようにした。

(4) FACOMサービスサブルーチンの作成

FACOMサービスサブルーチンである、TIME、DATEに相当するサブルーチンTIME__a、DATE__aを新たに作成した。

(5) 原研FACOM特有のライブラリ呼び出しに対する修正

原研FACOM特有の図化ライブラリを使用している箇所はダミー化、FORCAL/Gの相当する機能への置き換え等を行った。

EWS版へ変換されたプロットプログラムのディレクトリ構造を図3に示す。ファイル変換プログラムと時系列プロットプログラムは共にディレクトリiplot内にある。

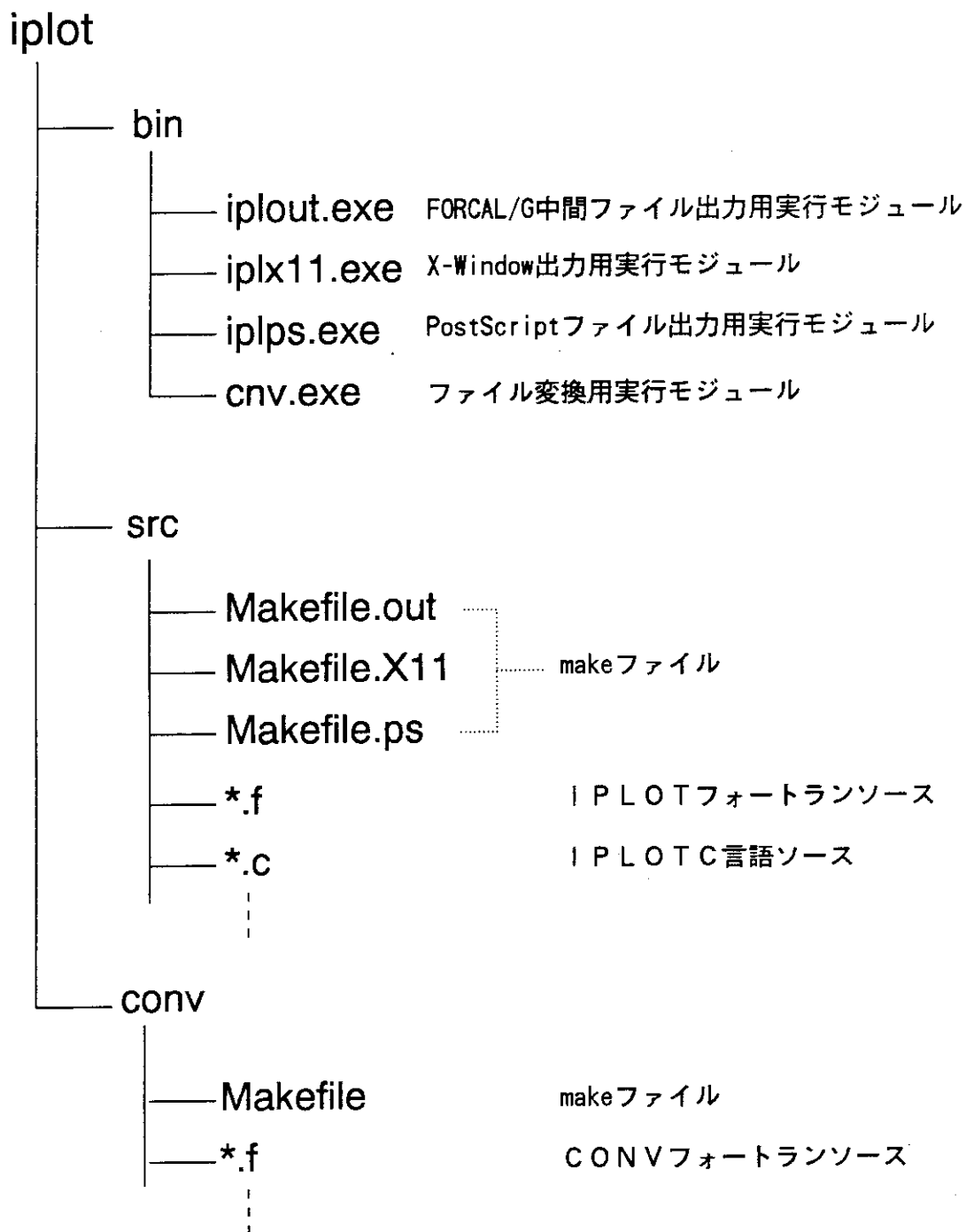


図3 プロットプログラムのディレクトリ構造

4. 2次元図形表示機能の開発

本章では、3次元ベッセルコンポーネントの評価作業を効率的に進めるために開発した2次元図形表示機能の概要について述べる。本機能は以下に示す2種類の機能を有している。ディスプレイ上に表示された画像データはX Windowのユーティリティプログラムであるxwd及びxwudを使用することによりセーブ及び再表示が可能である：

- (1) I P L O T用グラフィックファイルに保存されている計算結果または実験結果の2次元図形表示を行う機能、
- (2) EWS版REFLA/TRACコードの計算と同期して2次元図形表示を行う機能。

図4にEWSへ導入した2次元図形表示機能のファイル構成を示す。本機能に関するファイルはすべてディレクトリvtracの中においた。上記(1)の機能に関する実行モジュールはgraexeであり、ソース類はディレクトリgraphiの中においた。また、上記(2)の機能に関する実行モジュールはvtrexであり、ソース類はディレクトリvtracの中においた。

4. 1節において(1)の機能を実現するために開発したプログラムの概要を、4. 2節において(2)の機能を実現するために開発したプログラムの概要を述べる。

4. 1 ポストプロセッサとしての2次元図形表示プログラムの開発

3. 1節で開発したグラフィックファイル変換プログラムconvから出力されるI P L O T用計算結果ファイル、またはI P L O T用実験結果ファイルを読み込み、2次元の図形表示を行うプログラムgraexeを開発した。図5にgraexeのメインルーチンのフローシートを示す。

処理の流れとしては、まず計算結果の表示かまたは実験データの表示かをキー入力し、次に表示の速度をキー入力する。ここからプログラムがユーザーの作成したカードイメージ入力データ、及びI P L O T用計算結果グラフィックファイルもしくはI P L O T用実験結果ファイルを読み込む。次にタイトル、カラーバーを作成し、その後データの各タイムステップごとに逐次被覆管表面温度、ボイド率、流体温度等の2次元コンター図を表示する。

以下に、本プログラムのインクルードファイルの概要、サブルーチンの概要及び実行例を示す。本プログラムの操作方法については付録Cにまとめた。

(1) インクルードファイルの概要

本プログラムのコモンブロックはすべてFORTRANインクルードファイル(inc_graphi, inc_plot, inc_xll)の中においた。また、プロットデータの配列要素の個数をパラメータ文とし、それをインクルードファイル(inc_para)の中に置いた。以下にインクルードファイルの概要を述べる。

(a) inc_graphi

ユーザが記述したカードイメージの入力データから読み込んだ値を格納するコモンブロックをまとめたインクルードファイルである。

(b) inc_para

時間データ及びプロットデータの各個数を設定するパラメータ文を記述するインクルードファイ

ルである。実際に本パラメータで設定する配列変数はinc_plotに格納してある。

- (c) inc_plot
I P L O T用グラフィックファイルから読み込んだデータを格納するコモン変数をまとめたインクルードファイルである。
- (d) inc_x11
E W Sの図形表示機能に受け渡すデータを格納したコモン変数をまとめたインクルードファイルである。

(2) サブルーチンの概要

以下に個々のサブルーチンの概要を述べる。いくつかのサブルーチンは図4のフローシート上でその位置が確認できる。

- (a) charcut
文字変数を読み込み、文字を左づめにして文字数を調べる。
- (b) colorx
データ値とカラーテーブルの最小値及び最大値を読み込み、本データ値のカラー番号を設定する。
- (c) cread_exp
カードイメージの実験データ表示用入力データを読み込み、各変数に値を格納する。
- (d) cread_iplot
カードイメージの計算結果表示用入力データを読み込み、各変数に値を格納する。
- (e) draw_fl5
フラグiiiを受け取り、iii=1ならば計算結果表示用となり、iii≠1ならば実験データ表示用となり、それぞれに固有な形式でX11グラフィックウインドウに再表示要求を出す。
- (f) draw_timestep
計算結果の表示において、時間データをウインドウに書き込む。
- (g) draw_timestep_exp
実験データの表示において、時間データをウインドウに書き込む。
- (h) draw_title
フラグiiiを受け取り、iii=1ならば計算結果表示用となり、iii≠1ならば実験データ表示用となり、タイトルをX11グラフィックウインドウに書き込む。
- (i) draw_win1
計算結果表示用。ある時間ステップでの被覆管表面温度（固定）を1つめのウインドウにプロットする。
- (j) draw_win2
計算結果表示用。ある時間ステップでの物理量（任意、被覆管表面温度を除く）を2つめのウインドウにプロットする。
- (k) draw_win3
計算結果表示用。ある時間ステップでの物理量（任意、被覆管表面温度を除く）を3つめのウインドウにプロットする。

- (l) draw_win_exp
実験データ表示用。ある時間ステップでの物理量（任意）をウインドウにプロットする。
- (m) fread_exp
実験データから時間データとプロットするデータを配列変数に書き込む。
- (n) fread_iplot
計算結果のファイルを読み込み、配列変数に書き込む。
- (o) graphi_xpich
プロットする図のX軸方向のピッチ幅を設定する。
- (p) graphi_ypich
プロットする図のY軸方向のピッチ幅を設定する。
- (q) main
メインルーチン ①カードデータの読み込み
②プロットデータの読み込み
③X, Yピッチの設定、タイトルの表示
④時間、プロットの表示
の流れで処理が行われる。
- (r) openwin_exp
実験データ表示用ウインドウの作成要求を示す。
- (s) openwin_iplot
計算結果表示用ウインドウの作成要求を示す。
- (t) pause10
キー入力した遅延時間オプションに従って表示時間間隔をコントロールする。
- (u) pregraphi_exp
実験データ表示用。カラーバーを作成する。
- (v) pregraphi_iplot
計算結果表示用。カラーバーを作成する。
- (w) waitx
実行中リターンキーを入力すると、実行が一旦停止する機能を持つルーチン。再度リターンキーを入力すると実行が再開される。
- (x) その他のX11用サブルーチン
X11グラフィックルーチンをcallするインターフェースルーチンであり、ルーチン名の最後の文字が1の時はウインドウ1用、2の時はウインドウ2用、3の時はウインドウ3用となる。下表に個々のルーチンの内容を示す。また、これらのルーチンはすべてx11pac1sun.c, x11pac2sun.c, x11pac3sun.cに記述されている。

ルーチン名	内容	ルーチン名	内容
x11opn	開始時の処理	x11fill	多角形の塗りつぶし
x11igd	ウインドウの設定1	x11era	未使用

xllwin	ウインドウの設定2	xllend	終了時の処理
xlldrs	未使用	xllcur	未使用
xlldrw	線分の書き込み	xllctb	未使用
xllfls	再表示	xllcol	色の設定
xllchi	文字サイズの設定	xlldsh	未使用
xllchr	文字の書き込み	xllclp	未使用
xllera	未使用	xllrec	矩形の塗りつぶし
xllresize	未使用	xllpcr	未使用

(3) 実行例

図6に計算結果に対する実行例を示す。この計算は大型再冠水平板炉心試験⁽⁸⁾に対するものでVESSELコンポーネントを2次元にノーディングして計算した。上左の図が被覆管温度、上右がボイド率、下右の図が軸方向の液相流量に対する2次元コンター図である。各図とも炉心部分についてのみ表示した。炉心は横方向8体のバンドルからなり、バンドル3、4の出力がもっとも高く、バンドル7、8の出力がもっとも低い。出力分布に対応し、出力が高いほど被覆管温度が高い。一方、炉心内のボイド率分布はほぼ平坦である。液相の流量は出力の高いバンドルで高く、出力の低いバンドルでは炉心中央付近から下では下降流となっている領域がある。このような各物理量の解析は従来多数の時系列プロット等により行っていたが、開発した2次元図形表示機能により比較的簡便に行えるようになった。

4. 2 計算と同期して2次元図形表示を行うプログラムの開発

2章で作成したEWS版REFLA/TRACコードの3次元VESSELコンポーネント計算について、2次元の図形表示機能をつけ加えたプログラムvtrexexを開発した。このvtrexexを実行することによりVESSELコンポーネント内の被覆管表面温度、ボイド率、流体温度等の計算結果を、計算実行中にモニタリングする事が可能である。

本プログラムの図形処理の流れは、まず既存サブルーチンINITの時点で新たに必要となる図形処理用入力データTRCGDATを読み込む。この図形処理用入力データの中には、表示させたい場所(ノード)のデータが含まれる。次に、その指定したノードについてREFLA/TRACコードのダイナミックエリア内のポインターを計算する。そして、実際の物理量を計算するルーチンが始まり、各タイムステップごとにポインターで指定した物理量をプロットする。図7に計算と同期して2次元図形表示を行うために新たに追加したルーチンのツリー図を示す。

以下に、EWS版REFLA/TRACコードから修正したルーチン及び新たに付け加えたルーチンの概要及び本プログラムの実行例を示す。本プログラムの操作方法については付録Cにまとめた。

(1) 変更及び追加箇所の概要

EWS版REFLA/TRACコードに計算と同期して2次元図形表示を行わせるため、いくつかのルーチンでソースレベルでの変更、追加を行った。

(a) ショートエディット機能の修正

図形表示機能のベースとなるショートエディット機能で、VESSELコンポーネントの部分の不具合を修正した。

(b) 図形表示機能用ルーチンの呼び出しのための修正

図形表示機能用ルーチンの呼び出しのために、既存のサブルーチンINIT,TRANSに修正（新しいサブルーチンのCALL文を追加）を行った。

(c) 図形表示機能用ファイルの入出力

図形表示機能用ファイルについて、macdep.f内のサブルーチンz\$openz\$closeにてopen/close文を加えた。ここで、OPEN文では2章でのプログラムと同様、装置番号とファイル名が1対1で対応するように、本機能のファイル名をソースプログラム上で固定とした。

(d) コモン変数の追加

新たにy_h_com12, y_h_fin, y_graphi, y_plot, y_xwinなるインクルードファイルを作成し、これらの中に図形表示機能用のコモン変数を記述した。y_h_com12, y_h_finには、必要な物理量ポインター及びポインター検索に必要な変数等を格納した。y_graphiには、図形表示に必要なデータ（図タイトル、表示の最大値、最小値、表示物理量の個数等）を格納した。y_plotには、入力データで指定したノードの物理量が入るコモン変数を格納した。また、y_xwinには、実際に図形表示するための座標等を格納した。

(e) グラフィック表示用追加サブルーチン

グラフィック表示用に下表に示すようなサブルーチンを新たに作成した。

サブルーチン名	内 容	サブルーチン名	内 容
y_inp	TRCGDATを読み込む	y_fls	図の再表示要求を出す
y_gxpich	x方向のピッチ幅を求める	y_drawwin1	被覆管表面温度図を作成する
y_gypich	y方向のピッチ幅を求める	y_drawwin2	2つめの図を作成する
y_openwin	ウインドウを作成する	y_drawwin3	3つめの図を作成する
y_pregraphi	カラーバーを作成する	y_drawtimestep	時間、タイムステップの出力
y_gtitle	タイトルを作成する		

(f) ポインター検索用サブルーチンの修正追加

REFLA/TRACコードのショートエディットでは、ユーザーの必要とする物理量をプリン

トアウトするため、ダイナミックエリア内のポインターを計算し、そのポインターの示す場所の値を抜き出す処理を行っている。今回の図形表示機能についても任意の場所の値を必要とするため、ショートエディットのポインター検索部分と同様な処理をする必要がある。そのため、ショートエディットのポインター検索用サブルーチンを全てy_を付けてコピーした。コピー後、これらのサブルーチン内のポインターなどを示す変数（インクルードファイルy_h_com12, y_h_finの中に記述するコモン変数）を書き換えた。

(g) ショートエディット用サブルーチンHCOMP, HMEDITの修正

REFLA/TRACコードにおけるショートエディットの機能の入出力をコントロールするルーチンとしてHCOMP, HMEDITなるサブルーチンがある。これらを、y_hcomp, y_hmeditなるサブルーチン名でコピーし、今回の図形表示機能用に修正した。サブルーチンy_hcompでは、入力データの読み込みルーチン、x y軸のピッチ幅の計算ルーチン、タイトル及びカラーバーの設定ルーチン、そして、入力データで指定したノードのポインターを計算するルーチン呼び出している。サブルーチンy_hmeditは各タイムステップごとに呼び出される。但し、タイムステップが入力で指定した表示間隔でない場合には何もせず、親ルーチンに戻る。ここでは、時間、タイムステップを画面に表示するルーチン、及び求めたポインター物理量を実際に画面に表示するルーチン呼び出している。

(2) 実行例

図8にvtrex起動時の実行例を示す。この計算は4. 1節の実行例と同様に大型再冠水平板炉心試験に対するものでVESSELコンポーネントを2次元にノーディングして計算した。上左の図が被覆管温度、上右がボイド率、下右の図が気液各相の流体温度に対する2次元コンター図である。本実行例では、このVESSELコンポーネントの部分について図形表示を行いながら計算を実行した。

被覆管表面温度については、炉心領域のデータであるため下部プレナム、上部プレナム、バッフル、ダウンカマを除いた部分の温度を表示した。ボイド率については、下部プレナム、上部プレナム、バッフル、ダウンカマを含む全ての値を表示させた（右から1、2列目がダウンカマ領域、3列目がバッフル領域。但し、解析例ではダウンカマ下端及びバッフル下端は閉じられている。）。流体温度は下部プレナム、上部プレナムは含むが、バッフル、ダウンカマを含めず、各レベルでの気相（蒸気）温度及び液相温度を交互に表示した。

4. 1節での実行例と同様に、出力分布に対応した挙動が識別できる。本図形表示機能により計算実行時から計算の内容がチェックでき、VESSELコンポーネントのモデルの評価が簡便に行えるようになった。

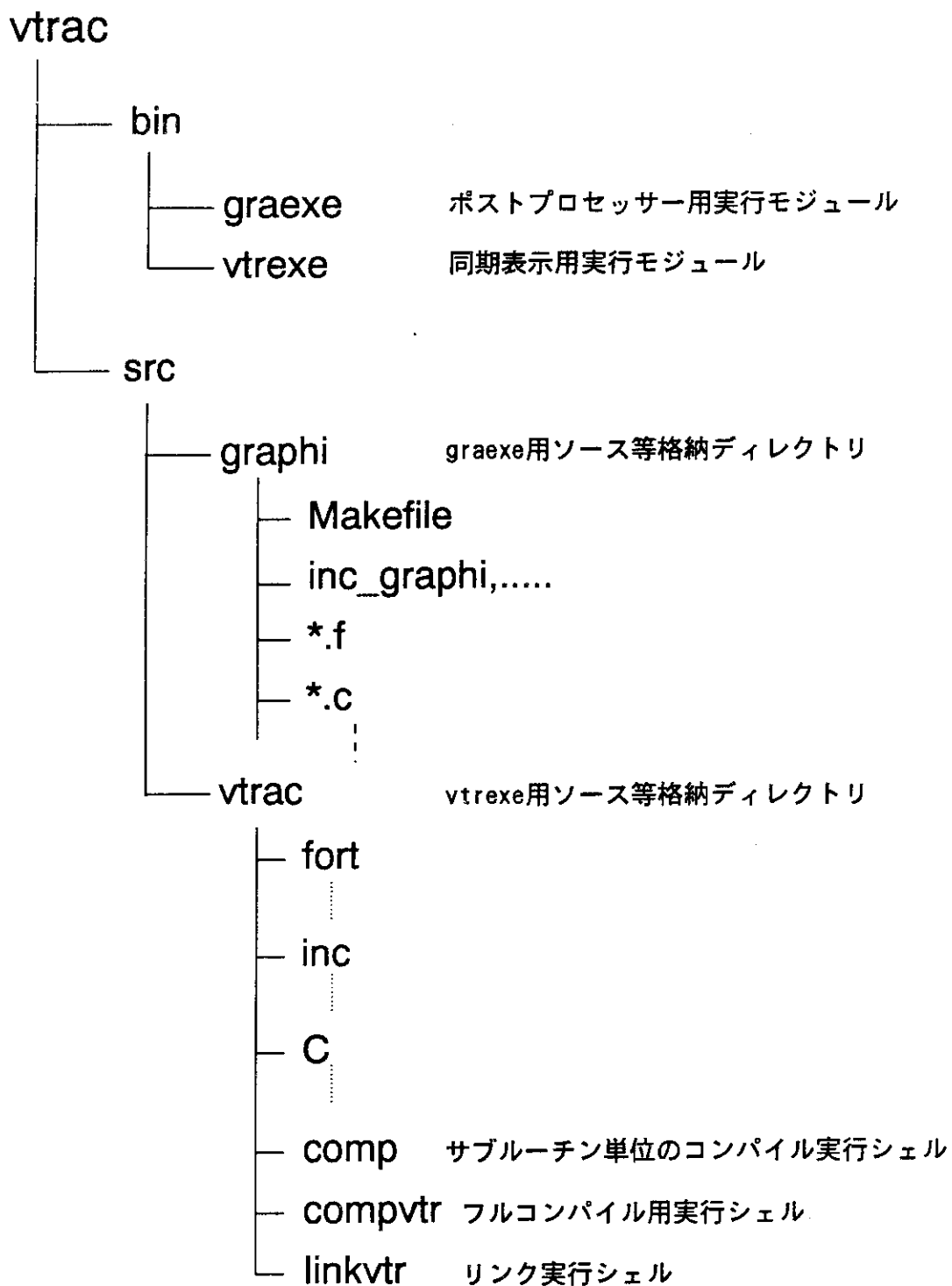


図4 2次元図形表示用プログラムディレクトリ構造

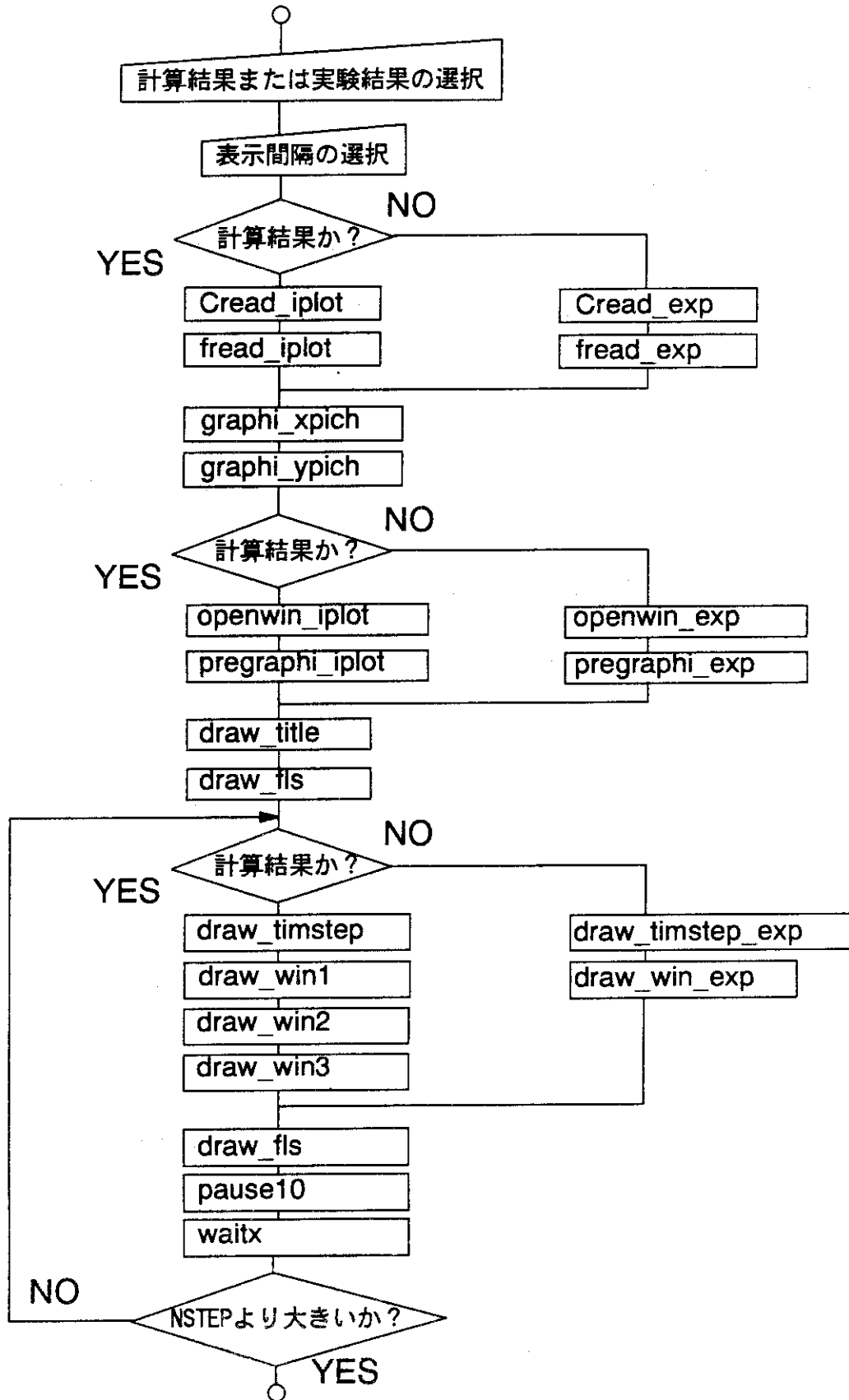
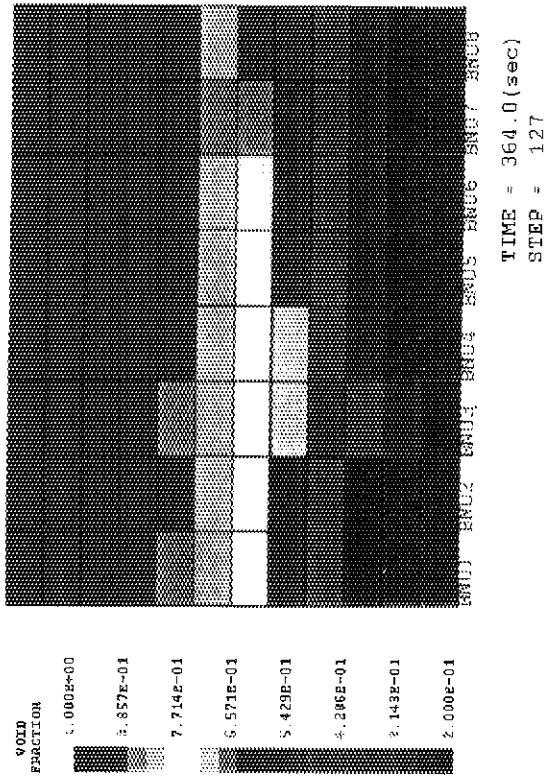
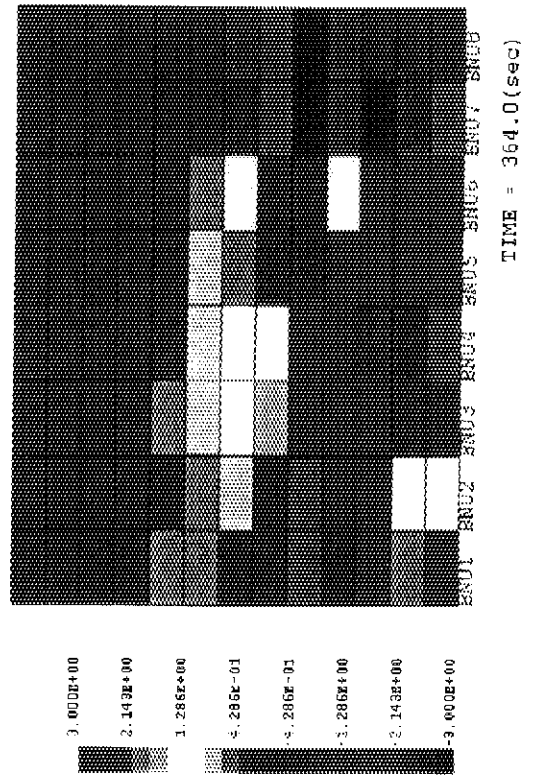


図5 メインルーチンの流れ図

SCTF(621) by R-TRAC Void fraction



SCTF(621) by R-TRAC Axial ML (kg/s)



SCTF(621) by R-TRAC Clad temperature

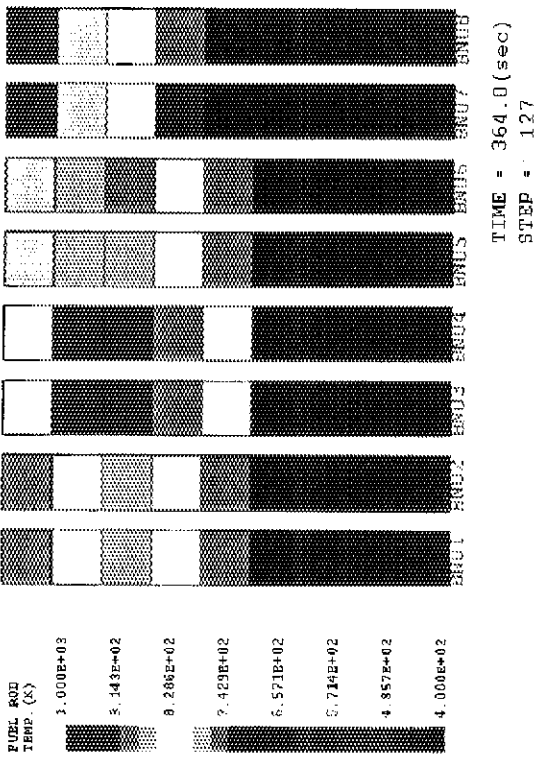


図6 ポストプロセッサとして計算結果を2次元図形表示した例

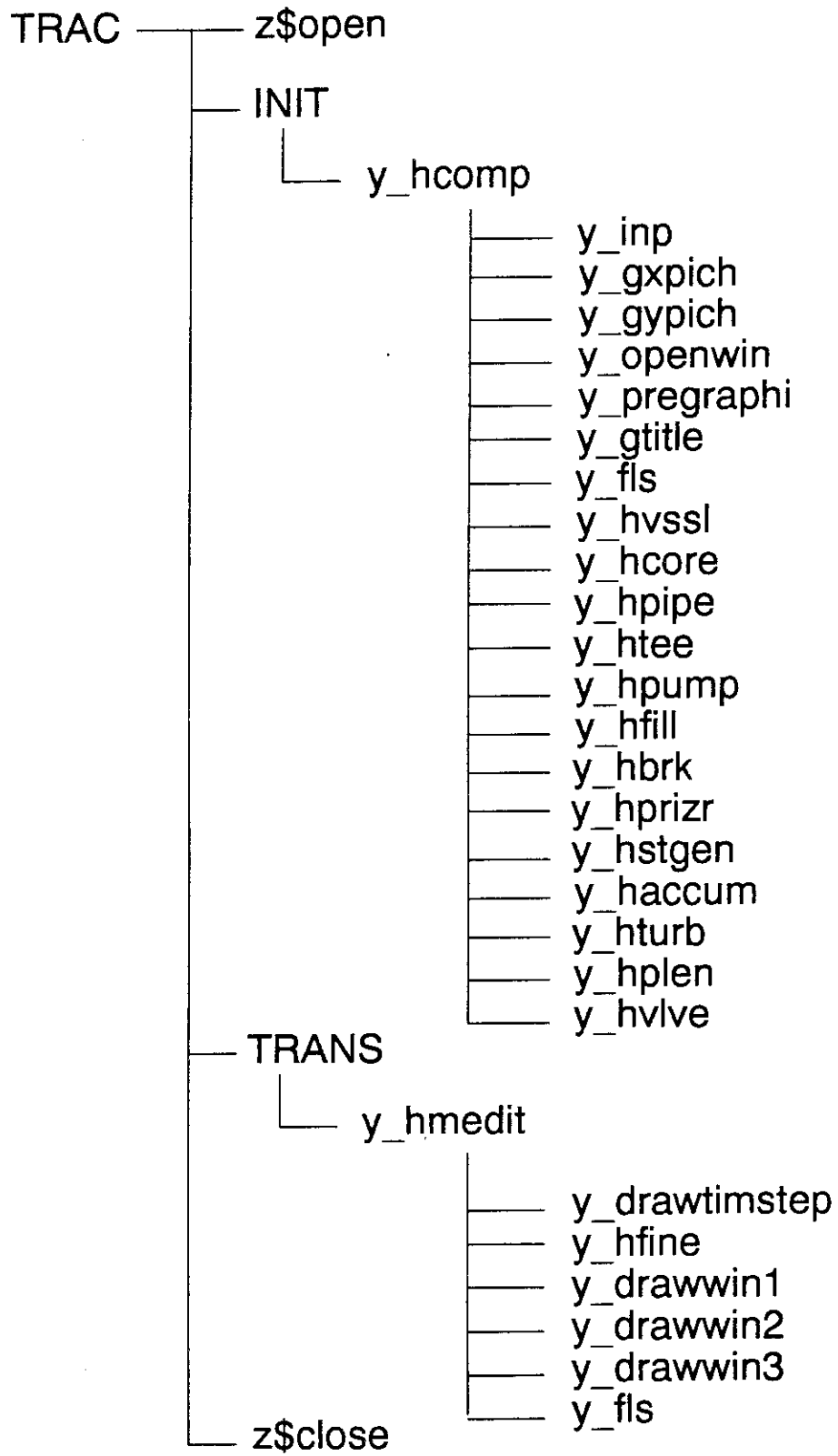


図7 計算と同期して2次元図形表示を行うために追加したプログラムのツリー

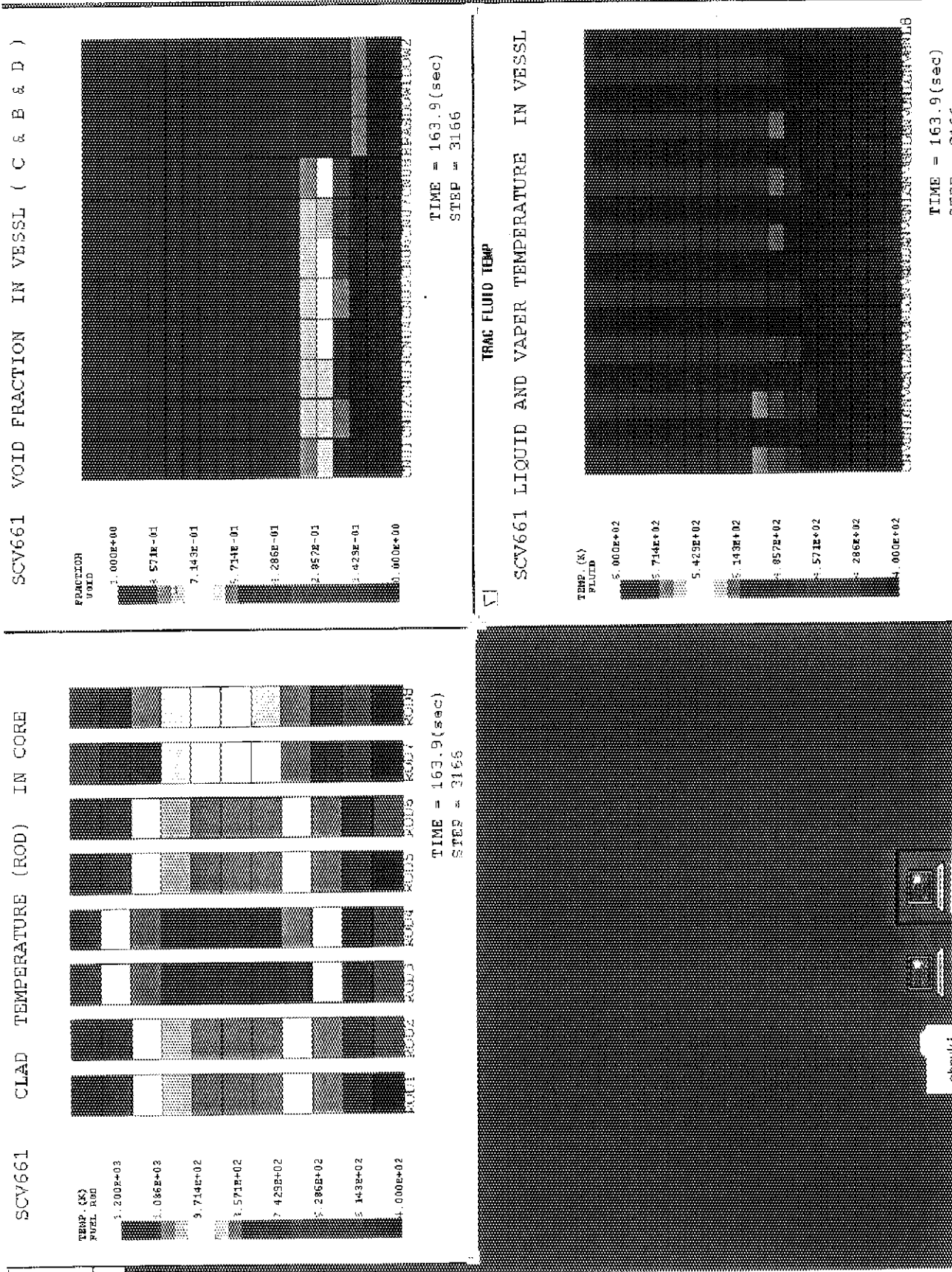


図8 計算と同期して計算結果を2次元図形表示した例

5. 結 言

REFLA/TRACコードの3次元モデルの評価作業の効率を上げるため、大型計算機と同程度の高速な計算が可能で、多次元の図形出力機能に優れたEWS上で稼働するEWS版REFLA/TRACコードを開発した。併せて、時系列プロット機能の整備、及び2次元図形表示機能の開発を行い、3次元計算の解析を容易に行えるようにした。特に、新たに開発した2次元図形表示機能は、ポストプロセッサとしての機能だけではなく、計算と同期して図形表示を行うこともでき、3次元計算の膨大な出力結果の解析効率が格段に向上するものと期待できる。

今後、より計算速度の高速なEWSへの導入、並びに流速や流量のベクトル図表示、3次元図形による表示等、図形表示機能の拡充により、より一層便利なシステムになるものとする。

謝 辞

本プログラムの開発・整備にあたり、㈱CRC総合研究所の新田芳和氏並びに鈴木俊之氏から多大な助力を得ました。ここに謝意を表します。

5. 結 言

REFLA/TRACコードの3次元モデルの評価作業の効率を上げるため、大型計算機と同程度の高速な計算が可能で、多次元の図形出力機能に優れたEWS上で稼働するEWS版REFLA/TRACコードを開発した。併せて、時系列プロット機能の整備、及び2次元図形表示機能の開発を行い、3次元計算の解析を容易に行えるようにした。特に、新たに開発した2次元図形表示機能は、ポストプロセッサとしての機能だけではなく、計算と同期して図形表示を行うこともでき、3次元計算の膨大な出力結果の解析効率が格段に向上するものと期待できる。

今後、より計算速度の高速なEWSへの導入、並びに流速や流量のベクトル図表示、3次元図形による表示等、図形表示機能の拡充により、より一層便利なシステムになるものとする。

謝 辞

本プログラムの開発・整備にあたり、㈱CRC総合研究所の新田芳和氏並びに鈴木俊之氏から多大な助力を得ました。ここに謝意を表します。

参考文献

- (1) Liles D. R., et al.: "TRAC-PF1/MOD1: An advanced best-estimate computer program for pressurized water reactor thermal-hydraulic analysis", NUREG/CR-3858, LA-10157-MS (1986).
- (2) Akimoto H. and Murao Y.: "Development of reflood model for two fluid model code based on physical models used in REFLA code", J. Nucl. Sci. Technol. 29[7], 642 (1992).
- (3) Akimoto H., et al.: "Assessments of REFLA/TRAC code for various postulated accidents in PWR", Fifth Int. Topical Meeting on Reactor Thermal Hydraulics (NURETH-5) Vol. VI, 1797 (1992).
- (4) 秋本 肇他2名: "REFLA/TRACコード1次元再冠水モデルの予測性能評価", JAERI-M 93-240 (1993).
- (5) Edwards A. R. and O'Brien T. P.: "Studies of phenomena connected with the depressurization of water reactors", J. Br. Nucl. Energy Soc., 9 (1970).
- (6) Agostini G., et al.: "Density measurements of steam-water mixtures flowing in a tubular channel under adiabatic and heated conditions", CISE-R-291 (1969).
- (7) Akimoto H., et al.: "Analysis of direct contact condensation of flowing steam onto injected water with multifluid model of two-phase flow", J. Nucl. Sci. Technol. 20[12], 1006 (1983).
- (8) Ohnuki A., et al.: "Study on ECC injection mode in reflood tests with SCTF core II -Comparison between gravity and forced feeds-", JAERI-M 91-001 (1991).
- (9) Okubo T., et al.: "Evaluation report on CCTF core-II reflood test C2-4 (Run 62) Investigation of Reproducibility", JAERI-M 85-026 (1985).
- (10) FORCAL/G EWS版 説明書、株式会社CRC総合研究所.

付録A FACOM版からEWS版へ変換する際の変更内容

A. 1 REFLA/TRACコードをFACOM版からEWS版へ変換する際の変更内容

ここでは、2. 1節で述べた変更内容を具体的に述べる。

(1) 特種文字、16進表記の変更

EWSの文字表示が、ASCIIコードであるため、¥は無条件に\$に変換されてしまう。FACOM版REFLA/TRACコードでは¥、したがってEWSでは\$から始まるサブルーチン、コモン、変数が多数存在する。この記述はEWS上のFORTRAN77では許されないため、ソースファイルすべてに関して、\$から始まるサブルーチン名、コモン名、変数名について、先頭の\$をzに変更した。

FACOM版	SUBROUTINE ¥BSFTL
EWS版	SUBROUTINE zBSFTL

また、ファイル名では、\$は使用できないため、ファイル名上のすべての\$を#に置き換えた。

FACOMとEWSでは16進表記法が異なるため、16進数を与えているデータ文に対し変更を行った。

FACOM版	DATA A/Z0000FFFF/
EWS版	DATA A/Z'0000FFFF'/

(2) インクルード文の変更

REFLA/TRACコードの全てのソースファイルに関して、インクルード文を以下に示すように、EWS用FORTRAN77に準拠する記述に変更した。

	----*----1----*----2----*----3
FACOM版	*INCLUDE 'filename'
EWS版	INCLUDE 'filename'

(3) ファイルアロケートの変更

EWS版REFLA/TRACでは、ファイルアロケートの方法に、EWS標準の事前結合のない手法であるOPEN/CLOSE文によるダイナミックアロケーションを用いた。この処理を一括して行うため、メインルーチンから呼ばれるz\$open, z\$closeなるサブルーチンを新たに追加した。

ただし、SOEditで使用する装置番号15, 16のファイルに関しては、SOEdit用入力ファイルの有無を問い合わせるように、サブルーチンP\$PIMP (FACOM版ではP¥PIMP) と、ここで使用するコモンP\$IO (FACOM版ではP¥IO) を変更した。これによりSOEdit用入力ファイルが存在しない場合、またはその1レコード目が想定されるデータ ('MAJOR', '1D', '3D', 'TIMER', 'END'のいずれか) でない場合には、SOEditの出力ファイルを作成しないといった機能をもたせた。

サブルーチン z\$open, z\$closeはファイルmacdep.fの中に記述し、../trac/fortの中に置いた。

OPEN文において、装置番号とファイル名が1対1で対応するように、EWS版REFLA/TRACコード実行時の入出力ファイル名をソースプログラム上で固定とした。

(4) FACOMサービスサブルーチンの変更

FACOM版REFLA/TRACではFACOMサービスサブルーチンのDATE, TIME, CLOCKM, EXITを使用している。ここでDATE, TIME, CLOCKMに対して新たにdatex, timex, clockmなるEWS版サブルーチンを作成し (EWSではdate, timeはサブルーチン名として使用できなかった)、これらサービスサブルーチンを呼び出すCALL文のサブルーチン名を変更した。datex, timex, clockmは、C言語記述のプログラムである。EXITに対しては、STOP文に置き換えた。

またFACOM版REFLA/TRACで使用するアセンブラルーチン $\$LOCF$ は、 $z\$locf$ なるC言語記述の関数へ置き換えた。そこでこれらのサービスサブルーチン、アセンブラルーチンは、 $datex$, $timex$, $clockm$, $z\$locf$ なるファイル名でディレクトリC内に格納した。

(5) サブルーチン引き数の修正

親ルーチンと子ルーチンの、引数の長さ(バイト数)がサブルーチン間で統一するように親ルーチン側CALL文の引数を修正した。たとえば、子ルーチンの引数が8バイトホレリス型で受け取る場合で、FACOM版ソースプログラムの親ルーチンでの引数の中身が8バイトに満たない場合には、親ルーチンのCALL文のホレリス型パラメータの後ろに空白をつけ、8バイトにしておく。このような修正を行った子ルーチン名は、EOVLY, GRFPUT, IOVLY, LOCPMP, LOCTEE, LOCTRB, LOCVLV, WIARR, WARRAYである。例として修正前と修正後のサブルーチンWARRAYのCALL文部分のリストを示す。

修正前	CALL WARRAY(5HSOLID, A(BUMP+LSN), NCELLS)
修正後	CALL WARRAY(8HSOLID , A(BUMP+LSN), NCELLS)

(6) EQUIVALENCE文の修正

8バイトREALと4バイトINTEGERとのEQUIVALENCE文で、インクルードファイルに対して、先頭アドレスが規定出来るよう修正を行った。すなわち次のような修正である。

修正前	EQUIVALENCE(A(1), IA(1))
修正後	EQUIVALENCE(A(1), IA(1, 1))

以下に、この修正を行ったサブルーチンおよびインクルードファイルを示す。

<サブルーチン>	BDPLEN, SVSET, SVSETI
<インクルードファイル>	ACCUMVLT, BLANKCOM, BREAKVLT, COREVLT, FILLVLT, FIXVLT FIXEDL01, FIXEDLT, GENVLT, LCMSPACE, PLENDATA, PRIZEVLT PUMPVLT, P\$IOPT(PYIOPT), STGENVLT, TEEVLT, TURBNVLT VALVEVLT, VSSELVLT

(7) エントリー引数の修正

FACOM版REFLA/TRACのサブルーチンCONDENの中にある、エントリーCNDINT, CNDCAL, CNDCA2は、それぞれ引数の個数が異なっている。FACOMのフォートランでは普通各ルーチンを静的に結合するが、EWSの場合は動的に結合する。従ってエントリーが呼ばれた時、引数でない変数に対しては値が保証されない。そこで、EWS版REFLA/TRACのサブルーチンエントリーCNDINT, CNDCAL, CNDCA2についてCALL文とENTRY文の引数の個数をサブルーチンCONDENのSUBROUTINE文と合わせた。以下に修正後のリストを示す。

CNDINTについて

修正前	CONDEN ENTRY CNDINT(TIME, MSC, NCELLS, JBG, JEND, ICALC, SUBENG)
	HTIF CALL CNDINT(TOT, MSC, NCELLS, JBG, JEND, ICALC, SUBENG)
修正後	CONDEN ENTRY CNDINT(TIME, MSC, NCELLS, i1, JBG, JEND, ICALC, SUBENG, r1, r2)
	HTIF CALL CNDINT(TOT, MSC, NCELLS, 0, JBG, JEND, ICALC, SUBENG, 0., 0.)

CNDCALについて

修正前	CONDEN ENTRY CNDCAL(J, JEND, SUBENG, ET0)
	ETACAL CALL CNDCAL(JJJ, JEND, SUBENG, ET0)

修正後 CONDEN ENTRY CNDCAL(r1, i1, i2, J, JBG, JEND, ICALC, SUBENG, ETO, r2)
 ETACAL CALL CNDINT(0., 0, 0, JJJ, JBG, JEND, ICALC, SUBENG, ETO, 0.)

CNDCA2について

修正前 CONDEN ENTRY CNDCAL(JBG, JEND, SUBENG, ETO, DDMIN)

ETACAL CALL CNDCAL(JBG, JEND, SUBENG, ETO, DDMIN)

修正後 CONDEN ENTRY CNDCAL(r1, i1, i2, i3, JBG, JEND, i4, SUBENG, ETO, DDMIN)

ETACAL CALL CNDCAL(0., 0, 0, 0, JBG, JEND, 0, SUBENG, ETO, DDMIN)

(8) データ文の修正

ラベルドコモンCHGALPについて、コモン変数の初期化を、VOIDCK, NEWDLTサブルーチンの2箇所で行っていた。この記述はEWS上のFORTRAN77ではコンパイル時にエラーとなってしまう。従ってこれを改めて（コメント化）、VOIDCKのみで行うよう修正した。

(9) サブルーチンPACKITの修正

FACOM版REFLA/TRACコードのソースプログラムのサブルーチンPACKITの中で、新しい8バイト変数の前後4バイトずつに、異なる4バイト変数を代入する処理を行っている。ここで、サブルーチンPACKITの中に新たにEQUIVALENCEを作り、基の8バイト変数を4バイト変数に代入する部分、ならびにここでできた4バイト変数を2つずつの特殊な形式にして格納する部分について修正を行った。

(10) サブルーチンREADI, READRの修正

サブルーチンREADI, READRに対するCALL文の引数の数は、読み込み変数の個数によって異なるため、CALL文の引数の数とサブルーチンREADI, READRの引数の数が異なるものが多く存在していた。そこで、CALL文とSUBROUTINE文の引数の個数が常に同じになるように、引数の個数に応じてサブルーチンを分け、READI1~READI5, READR1~READR5を作成した。そして、READI, READRに対するCALL文を修正した。(READI及びREADRは削除した。)

(11) サブルーチンINPUT内におけるサブルーチンCLEARのCALL文の修正

サブルーチンINPUTの762行目と764行目の、CLEARをCALLする場合、FACOM版REFLA/TRACコードのソースプログラムのコーディングのままでは、常に実行エラーが起きてしまった。そこで実行エラーを取り除くため、以下の修正を行った。

762行目 修正前 CALL CLEAR(0, IA(1, LIJN)IT)
 修正後 CALL CLEAR4(0, IA(1, LIJN)IT*2)
 764行目 修正前 CALL CLEAR(0, IA(1, LOU)IT)
 修正後 CALL CLEAR4(0, IA(1, LOU)IT*2)

サブルーチンCLEARは、他のいくつかのルーチンでCALLされているが、EWSの実行時にエラーとはならないので修正は行っていない。

(12) コモン文の修正

ラベルドコモンDCMDLの中のCONDATなるコモン変数は、ブロックデータBLOCKCにて2000要素の配列として初期設定を行っている。しかし、実際使用する際には、先頭アドレスのみをコモンとして指定している。ここでラベルドコモンDCMDLを使用するサブルーチンについて、DCMDLをインクルード文として参照できるように変更した。新たに作成したインクルードDCMDLは、ディレクトリtrac/incの中へ置きCONDATの

配列要素は2000個とった。このインクルードDCMDLを使用するサブルーチンは、CNDMAX, CONDEN, ETACAL, HTIFである。

A. 2 グラフィックファイル変換プログラムをFACOM版からEWS版へ変換する際の変更内容

ここでは、3. 1節で述べた変更内容を具体的に述べる。

(1) ファイルの入出力に対するオープン文の追加、修正

対話形式（インタラクティブ）に各種ファイル名を入力できるよう、入出力ルーチンの修正、追加を行った。

装置番号5番は、標準入力用としてファイルはアロケートしないようにした。修正したサブルーチンは、READ5で、EWSの装置番号55番からCONVのデータを読み込むよう修正した。CONVの実行時メッセージ（FACOMでは標準出力）をファイルに落とせるように既存ソースプログラムすべてに対して、装置番号6番への書き込みを装置番号66番に変更した。

また新たに追加したサブルーチンは、macdepで、TRCGRFのオープン及びI PLOTのデータファイルのオープンを行う。ここで、TRCGRF形式のファイルのオープンする装置番号は、11番から始まりCONVの入力データで指定したファイル数まで、12, 13, ... と増加する。I PLOT用のファイルは装置番号2番に出力する。

(2) サブルーチンRDATAの修正

タイムエディットデータ（主に各種物理量が格納されている領域）を読み込む際のデータ変換に対して、サブルーチンRDATAの修正を行った。タイムエディットデータの始めの数ワードは、特殊な圧縮形式ではなく、1ワードに8バイトの変数が格納されている。ここでサブルーチンRDATAの文番号30の後のCALL BFIN (・・・)で、4バイト変数でデータを受け取り、8バイト変数の始め4バイトを取り出して4バイト変数で使用しようとしていた。しかしEWSでは8バイト変数の始め4バイトを抜き出しても4バイト変数とはならないので不具合が生じてしまう。ここで、次のような修正を加えた。

- (a) 1ワードに8バイトの変数がはいつているワード数を調べる。
- (b) BFINから新たに追加した8バイトローカル変数でまず読み込む。
- (c) (a)で調べたワード数までは8バイト変数を4バイトの本来代入すべき変数に代入する。
- (d) それ以外のデータは特殊な圧縮形式のデータであるため、このローカルの8バイト変数とEQUIVALENCEをとった4バイト変数を代入すべき変数に代入する。

A. 3 時系列プロットプログラムをFACOM版からEWS版へ変換する際の変更内容

ここでは、3. 2節で述べた変更内容を具体的に述べる。

(1) サブルーチンのダミー化、サブルーチン名の変更

次に示すサブルーチンについては、ダミーサブルーチンとした。

NLIN, LESQ1, LINEGL, PROMPT, SPLV, SHIFT, SVC99, TSSCMD

これらは、FACOMアセンブラルーチン、特殊なカルコンルーチン、算術ライブラリー等である。

サブルーチンipplot内でCALLされているALLOCMについてはアロケート時のメッセージに関するサブルー

チンでありEWS版では使用しないと考えられたためCALL文をコメントにした。また、サブルーチンTREADのサブルーチン名をTREADxに変更した。

(2) 特殊文字、16進表記の修正

A. 1(1)と同様の修正を行った。ここでは、\$から始まるサブルーチン名、コモン名、変数名について、先頭の\$の前にxを付けた。

FACOM版	COMMON/\$USER2/\$USER(30)
EWS版	COMMON/x\$USER2/x\$USER(30)

また、ファイル名では、\$は使用できないために、ファイル名上のすべての\$を_に置き換えた。

FACOMとEWSでは16進表記法が異なるため16進数を与えているデータ文に対し以下のような変更を行った。

FACOM版	DATA A/Z0000FFFF/
EWS版	DATA A/Z' 0000FFFF' /

(3) オープン文の修正、追加

EWSでは、OPEN文においてACTION指定子、及びSTATUS指定子の'SHR'は使用できない。ここで、STATUS='SHR'は、STATUS='UNKNOWN'に変更し、ACTION指定子は、すべて削除した。

また、FACOM版ではサブルーチンOPENの中で、ファイルのアロケートをアセンブラルーチン呼び出しで行っていたが、EWS版では、ここに新たにOPEN文を付け加えて、ファイルのアロケートを行えるようにした。

(4) FACOMサービスサブルーチンの作成

サブルーチンSAVE, CDATEでは、FACOMサービスサブルーチンであるTIME, DATEを呼び出している。ここで、TIME及びDATEと同機能のサブルーチンTIME__a, DATE__aを新たに作成して、新しくmacdep.fというファイル内においた。

(5) 原研FACOM特有のライブラリ呼び出しに対する修正

原研FACOMは、特有の図化ライブラリをもっており、FACOM版のI PLOTではこのルーチンを使用している。ここで原研FACOM特有のライブラリ呼び出しに対して、次のような修正を加えた。

LINEGL, RTPLOT	ダミーサブルーチンとする
LETTER	SYMBOLへ変更
PLOTの第3パラメータ	444, 666, 888は-3へ変更

ここで、SYMBOL及びPLOTはFORCAL/G (EWS用カルコンプ互換ライブラリ) のライブラリである。

付録B EWS版REFLA/TRACコード操作マニュアル

B. 1 EWS版REFLA/TRACコードの構成

2次元図形表示機能を除くEWS版REFLA/TRACコードは、REFLA/TRACコード本体、グラフィックファイル変換プログラム、及びプロットプログラムの3つのモジュールから成り立っている。各モジュールの実行の流れを図B. 1に示す。REFLA/TRACコード本体の実行では本文2. 2節表1に示した各入出力ファイルが関係する。グラフィックファイル(TRCGRF)の変換では変換の仕方を指定するのにデータファイルが必要となる。プロットプログラムでは、ディスプレイへの出力、PostScriptファイルへの出力、及びカルコン互換のライブラリーFORCAL/G用の中間ファイルへの出力の3種類の出力ができるになっている。B. 2節で各モジュールの使用法について述べる。また、B. 3節では各プログラムをソースレベルから修正する方法について述べる。

B. 2 使用法

(1) 環境設定

EWS版REFLA/TRACコードを使用する際、ユーザーは、個人のEWS環境に対して、いくつかの環境設定が必要となる。REFLA/TRACコードのユーザーは、ユーザーホームディレクトリの.cshrcファイルに以下のコマンドを付け加えると実行が簡便になる。

```
setenv      TRACHOME /.../..  ../trac
set path = ($path $TRACHOME)
set path = ($path /.../..  ../iplot/bin)
set path = ($path /.../..  ../forcal/bin)
```

1行目2行目は、REFLA/TRAC本体(trac)のあるディレクトリを、3行目はプロット用プログラム(conv. iplot)の実行モジュールが格納されているディレクトリを、4行目はFORCAL/Gの実行モジュールが格納されているディレクトリを指定している。これで、各実行モジュールにコマンドパスが設定される。

(2) EWS版REFLA/TRACコードの実行

(a) REFLA/TRACコード本体の実行

カレントディレクトリ(このディレクトリは、実行する問題名や各ケース名ごとに作成しておくことと便利である)に、実行用入力データを作成します。この時、入力データのファイル名はTRACINというファイル名にしておくこと(固定)。また、簡易凝縮モデルを使用する場合は、凝縮モデル用の入力データも必要となる。このファイル名はTRCF90に固定して用意すること。必要であるなら、TRCF99というファイル名で計算時間制御用ファイルを、TRCS05というファイル名で任意の物理量のリスト出力が得られるSOEditが使用できる。入力フォーマット等の詳細は、FACOM版REFLA/TRACコードと同一である。

ここでtrc.exeというプログラムを実行する。

```
% trc.exe
```

以下に示すファイルが出力される：

```
TRCGRF          グラフィックファイル
```

TRCDMP	計算結果ダンプファイル
TRCMSG	メッセージファイル
TRCOUT	リスト出力ファイル

リスタート計算をする場合、前回の計算結果ダンプファイル (TRCDMP) を TRCRST というファイル名にし、リスタート用の入力データファイルを TRACIN として計算を実行すればよい。

カレントディレクトリで複数回 `trc.exe` を実行した場合、その都度、上記のファイルが、オーバーライトしてしまうことに注意が必要である。

EWS版 REFLA/TRACコードにおいて、標準出力 (実行時のディスプレイ出力) は FACOM版の TTYへ出力するものを表示した。また、FACOM版ではデバッグライトを装置番号 61 に出力している。EWS版でも、このデバッグライトを TRCDBG というファイル名で出力できるようにした。但し現在の実行モジュールでは実行終了時に TRCDBG を削除するようになっている。

(b) グラフィックファイル変換プログラムの実行

カレントディレクトリで `cnv.exe` というプログラムを実行する。`cnv.exe` を実行すると、必要なファイルをキー入力するようメッセージが出る。このメッセージに従って入力ファイル名及び出力ファイル名をキー入力する。

例えば、LANL_FILE1 と LANL_FILE2 というファイル名の 2 つのグラフィックファイルから、IPLOT_DATA というファイル名で、IPLOT の入力ファイルであるプロットファイルを出力したいとする。このプログラムに入力データは CNVD というファイルに記述してあるとする。入力データは、FACOM版の実行において装置番号 5 番に割り当てられていたデータである。また、`cnv.exe` 実行時の出力メッセージ (FACOM版では装置番号 6 番) を CONVMSG というファイルにダンプする。

<実行例>

```
%cnv.exe
  (実行開始)
Please input Message Dump FILE <UNIT=6>
CONVMSG
Please input your DATA FILE <UNIT=5>
CNVD
Please input your TRCGRF No. 1/2 ..... 入力データCNVDで指定しているファイル数ま
LANL_FILE1                               で聞いてくる。
Please input your TRCGRF No. 2/2
LANL_FILE2
Please input your PLOT-FILE for IPLOT <UNIT=2>
IPLOT_DATA
  (実行終了)
%
```

ここで、ゴシック体の文字がユーザーのキーインプット、斜体文字が `cnv.exe` のメッセージです。

これらの入出力ファイル名の入力、ユーザーが `cnv.exe` を実行する際、必ずキー入力しなくてはなりません。もし、キー入力時にファイル名を間違えて入力してしまった場合は、`cntrlC` キーを押して実行を終了してから、もう一度同じ手順で、実行を再開すること。

(c) プロットプログラムの実行

ユーザーは、あらかじめエディターを使用して、カレントディレクトリにI P L O T用の入力データを作成しておく。ファイル名は任意でよい。入力データのフォーマット、コマンド等は、F A C O M版プロットプログラムと同一である。

このカレントディレクトリでプロットプログラムを実行する。プロットプログラムは、ユーザーの用途に合わせて、3つのプログラムを用意した。まず、iplout.exeという中間ファイルを作成するプログラム、iplx11.exeというX-Windowに直接出力するプログラム、iplps.exeというPostScript言語のファイルを出力するプログラムである。このPostScript言語のファイルはUNIXのlpコマンド(BSD系EWSではlprコマンド)を使用して、ポストスクリプト対応のプリンターに送るとプロット図がA4用紙に出力される。中間ファイルとは、FORCAL/Gの変換フィルタープログラム用のファイルである。FORCAL/Gの変換フィルタープログラムによって、この中間ファイルを、各デバイス(X-Window, PostScript言語のファイル等)に出力できる。この方法を使用すると、結果をX-Windowで確認したり、プリンターに送るファイルに出力したりすることができ、再度I P L O Tを実行しなくてすむ。

以下に実行例を示す。I P L O Tの実行結果リストは、装置番号66番(Sun-FORTRANであればfort.66というファイル)に出力される。

a) X-Windowに出力する場合

```
% iplx11.exe < I P L O Tデータ
      (ディスプレイ上にグラフィック用のWindowを新たに作成します)
```

b) PostScript言語のファイルに出力する場合

```
% iplps.exe < I P L O Tデータ > PostScriptファイル
      (PostScriptファイルのファイル名はユーザーが任意に指定)
```

c) FORCAL/G中間ファイルに出力する場合

```
% iplout.exe < I P L O Tデータ
```

c) で作成される中間ファイル名は、ユーザーのホームディレクトリにあるFORCAL/G環境設定ファイル.fcalrcで設定することができる。(デフォルトはforcalg.dat)

中間ファイルを作成した後、次のプログラムを実行することによりX-WindowまたはPostScript言語のファイルへ出力できる。

```
・X-Window          % plot_X11    <   中間ファイル
・PostScript言語のファイル % plot_ps    <   中間ファイル    > PostScriptファイル
```

B. 3 EWS版REFLA/TRACコードの修正

(1) REFLA/TRACコード本体の修正

REFLA/TRAC本体を新たに修正する場合は次の手順で行うこと。

- (a) カレントディレクトリに\$TRACHOMEからcomptrc、linktrcをコピーしてくる。
- (b) カレントディレクトリに修正したいソースプログラムをコピーし、修正する。
- (c) シェルコマンドのcomptrcを使用してソースプログラムをコンパイルする。
- (d) いまコンパイルしたカレントディレクトリのオブジェクトをリンクするようlinktrcの該当部分を修正する。
- (e) linktrcを実行する。

<例>

REFLA/TRACコードのディレクトリは/usr/trac、カレントディレクトリは/home/crcだとする。いま、

REFLA/TRACコードのメインルーチンを修正して新たに実行モジュール trctest.exeを作成するとする。
ユーザーの環境設定から、\$TRACHOMEは /usr/tracになっている。

```
% cp $TRACHOME/comptrc .
% cp $TRACHOME/linktrc .
% cp $TRACHOME/fort/main.f .
```

ここで、エディターを使用してソースファイルの修正を行う。

```
% comptrc main.f
```

カレントディレクトリにmain.oができる。次にlinktrcを修正する。

エディターでlinktrcを開き、\$TRACHOME/fort/main.oを/home/crc/main.oに修正し、更に最終行のファイル名trc.exeをtrctest.exeに修正してエディターを終了する。

次にlinktrcを実行する。

```
% linktrc
```

カレントディレクトリに実行モジュールtrctest.exeができる。

(2) LANL形式グラフィックファイル変換プログラム及び時系列プロットプログラムの修正

最初にIPLOTのディレクトリに移動する。この中にconv, src, binというディレクトリがある。グラフィックファイル変換プログラムを新たに修正したい場合、convの中へ移動する。

ここで、ソースファイルの修正を行い、makeを実行する。

```
% make
```

これにより新しいconv.exeが./binの下に作成される。

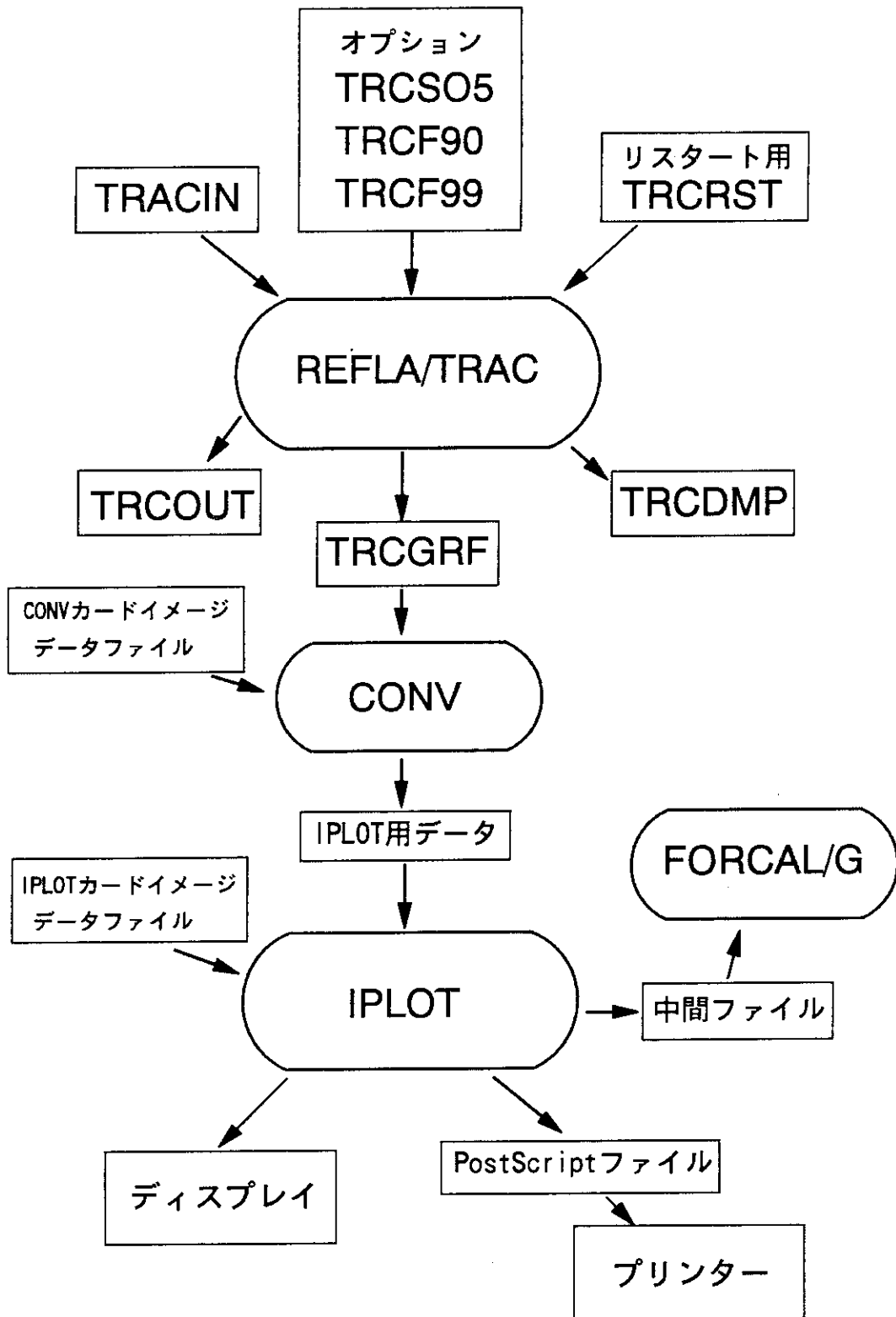
プロットプログラムを新たに修正したい場合、上記のsrcの中へ移動する。ここで、ソースファイルの修正を行い、makeを実行する。出力するデバイスに応じて3つのMakefileがある。

```
X-Window 出力の場合          % make -f Makefile.X11
```

```
PostScriptファイル出力の場合 % make -f Makefile.ps
```

```
中間ファイル出力の場合      % make -f Makefile.out
```

これで、./binの下に、それぞれiplx11.exe, iplps.exe, iplout.exeが作成される。



図B. 1 EWS版REFLA/TRACコードの処理の流れ

付録C 2次元図形表示システム操作マニュアル

C. 1 2次元図形表示システムの概要

本図形表示システムは目的に応じ、次に示す2つの異なるモジュールから成り立っている：

(1) graexe

I P L O T用計算結果グラフィックファイルもしくは実験結果ファイルを読み込み被覆管表面温度、ポイド率、流体温度等を表示するモジュール

(2) vtrexex

3次元VESSELコンポーネントの計算実行時に被覆管表面温度、ポイド率、流体温度等をモニタリングする機能をつけ加えたEWS版REFLA/TRACコード

本付録では、EWS版REFLA/TRACコードの解析結果を2次元図形表示システムにより、EWS上ディスプレイに2次元コンター図を表示するための入力データのフォーマット、各モジュールの実行方法、及び各モジュールの修正方法について述べる。

C. 2 使用法

(1) 環境設定及び図形表示用入力データ

本システム (graexe, vtrexex) を使用する際、ユーザーは個人のEWS環境に対して、いくつかの環境設定をする必要がある。ユーザーは、ユーザーホームディレクトリのシェル環境設定ファイル.cshrcファイルにつきのコマンドを付け加える。

```
set path = ( $path /**//* */*/vtrac/bin )
```

このコマンドは、本システムのロードモジュール格納ディレクトリに対するパスの設定を行うものである。

本システム (graexeとvtrexex) の図形表示用入力データは、一部の固定フォーマット形式のものを除いて全てフリーフォーマット形式を採用している。また、入力データのファイル名は固定とした。入力データファイル名を下表に示す。また、各入力データの構成及びフォーマットについては(2)及び(3)節の中で述べる。

モジュール	入力ファイル名	備考
graexe	gdata_iplot	計算結果グラフィックファイルの作図
	gdata_exp	実験結果ファイルの作図
vtrexex	TRCGDAT	計算と同期して作図する。

(2) graexeの実行

実行の際、データファイルはすべてカレントディレクトリから読み込まれる。したがって、gdata_iplotもしくはgdata_exp、及び計算結果グラフィックファイルもしくは各種の実験結果ファイルは、カレントディレクトリ内に格納しておく。

ここでgraexeを実行する。

```
% graexe
```

次に計算結果グラフィックファイル (I P L O T DATA) か、実験結果の作図かを入力するメッセージが出力されるのでキー入力する。

```
Please Input your data type (1 or 2)
```

```
1...I P L O T DATA
```

```
2...EXPERIMENT DATA
```

次に表示時間の遅延時間を設定するオプションを入力するようメッセージが出力されるので、キー入力する。

```
Please input delay time 1/2/3
```

```
1...quick          not delay
```

```
2...default        3 sec
```

```
3...slow           10 sec
```

ここで1か2か3をキー入力すると新たにグラフィック用のウィンドーが作成され実行を開始する。

本プログラムでは、実行開始後、実行中にリターンキーを入力すると実行を一端停止することができる。次にもう一度リターンキーを入力すると再度実行を始める。

(a) ポスト処理用入力データgdata iplot (計算結果用) の構成及びフォーマット

下表にgdata_iplotの構成を示す。

レコード番号	内 容
# 1	I P L O T標準形式ファイル名 (計算結果)
# 2	被覆管温度図タイトル
# 3	表示する燃料棒の本数
# 4	燃料棒1本当りの分割数
# 5	被覆管表面温度最小値、最大値
# 6	燃料棒名称 (任意)、I P L O T標準形式データタグID
# 7	燃料棒表示ポイント (高さ)
# 8	2つめの図のタイトル
# 9	2つめの図の表示するチャンネル数
# 10	2つめの図の各チャンネルの分割数
# 11	2つめの図の最小値、最大値
# 12	2 つ め の 図 の 各 チ ャ ン ネ ル 名 称 (任 意)

# 1 3	2つめの図の I P L O T 標準形式データタグ I D
# 1 4	3つめの図のタイトル
# 1 5	3つめの図の表示するチャンネル数
# 1 6	3つめの図の各チャンネルの分割数
# 1 7	3つめの図の最小値、最大値
# 1 8	3つめの図の各チャンネル名称 (任意)
# 1 9	3つめの図の I P L O T 標準形式データタグ I D

以下に、各レコードのフォーマットを示す。

#1	iplot_filename	最大8文字	I P L O T 標準形式データファイル名 (計算結果)
#2	titlfg1	最大45文字	被覆管表面温度図のタイトル
#3	num_rod	整数	表示する燃料棒の本数 (最大20本*)
#4	icells_rod	整数	表示する燃料棒1本当りの分割数 (最大40分割*)
#5	vrmin, vrmax vrmin vrmax	実数 実数	被覆管表面温度最小値 被覆管表面温度最大値
#6	rod_tit, rod_tag rod_tit rod_tag	4文字 8文字	表示する燃料棒の名称 (任意) I P L O T 標準形式データのタグ I D
(注)	本カードのフォーマットは (A4, 1X, A8) である。 本カードはnum_rod回繰り返す。		
#7	rod_hight	実数	燃料棒表示ポイント (高さ)
(注)	本カードは空白をデリミターとしたフリーフォーマットである。 rod_hightはicells_rod個ある。		
#8	titlfg2	最大45文字	2つめの図のタイトル
#9	num_void	整数	2つめの図で表示するチャンネル数の合計 (最大20*)
#10	icells_void	整数	2つめの図で表示する各チャンネル当たりの分割数 (最大40分割*)

#11	vvmin, vvmax vvmin vvmax	実数 実数	2つめの図の最小値 2つめの図の最大値
#12	void_tit	最大4文字	2つめの図で表示する各チャンネルの名称 (任意)
#13	void_tag (注)	8文字	2つめの図のタグID 本カードはブランクをデリミターとしたフリーフォーマットである。 void_tagはicells_void個ある。 #12, #13はnum_void回繰り返す。
#14	titlfg3	最大45文字	3つめの図のタイトル
#15	num_fluid	整数	3つめの図で表示するチャンネル数の合計 (最大20*)
#16	icells_fluid	整数	3つめの図で表示する各チャンネル当たりの分割数 (最大40分割*)
#17	vfmin, vfmax vfmin vfmax	実数 実数	3つめの図の最小値 3つめの図の最大値
#18	fluid_tit	最大4文字	3つめの図で表示する各チャンネルの名称 (任意)
#19	fluid_tag (注)	8文字	3つめの図のタグID 本カードはブランクをデリミターとしたフリーフォーマットである。 fluid_tagはicells_fluid個ある。 #18, #19はnum_fluid回繰り返す。

* - この制限は本文4. 1 (1) (c) のインクルードinc_plot内の値を変更し、フルコンパイルすれば変更できる。

以下にgdata_iplotのファイルの例を示す。この例に示すファイルを使い本文の図6が出力された。

```

IPLDAT
SCIF(621) by R-TRAC Clad temperature
8
9
400. 1000.
BN01 R*010107
BN02 R*010207
BN03 R*010307
BN04 R*010407
BN05 R*010507
BN06 R*010607
BN07 R*010707
BN08 R*010807

```

0.110 0.520 0.950 1.380 1.735 1.905 2.330 2.760 3.190
 SCTF(621) by R-TRAC Void fraction

8

12

0.2 1.0

BN01

AL010401 AL010501 AL010601 AL010701 AL010801
 AL010901 AL011001 AL011101 AL011201 AL011301
 AL011401 AL011501

BN02

AL010402 AL010502 AL010602 AL010702 AL010802
 AL010902 AL011002 AL011102 AL011202 AL011302
 AL011402 AL011502

BN03

AL010403 AL010503 AL010603 AL010703 AL010803
 AL010903 AL011003 AL011103 AL011203 AL011303
 AL011403 AL011503

BN04

AL010404 AL010504 AL010604 AL010704 AL010804
 AL010904 AL011004 AL011104 AL011204 AL011304
 AL011404 AL011504

BN05

AL010405 AL010505 AL010605 AL010705 AL010805
 AL010905 AL011005 AL011105 AL011205 AL011305
 AL011405 AL011505

BN06

AL010406 AL010506 AL010606 AL010706 AL010806
 AL010906 AL011006 AL011106 AL011206 AL011306
 AL011406 AL011506

BN07

AL010407 AL010507 AL010607 AL010707 AL010807
 AL010907 AL011007 AL011107 AL011207 AL011307
 AL011407 AL011507

BN08

AL010408 AL010508 AL010608 AL010708 AL010808
 AL010908 AL011008 AL011108 AL011208 AL011308
 AL011408 AL011508

SCTF(621) by R-TRAC Axial MI (kg/s)

8

13

-3.0 3.0

BN01

WZ010301 WZ010401 WZ010501 WZ010601 WZ010701
 WZ010801 WZ010901 WZ011001 WZ011101 WZ011201
 WZ011301 WZ011401 WZ011501

BN02

WZ010302 WZ010402 WZ010502 WZ010602 WZ010702
 WZ010802 WZ010902 WZ011002 WZ011102 WZ011202
 WZ011302 WZ011402 WZ011502

BN03

WZ010303 WZ010403 WZ010503 WZ010603 WZ010703
 WZ010803 WZ010903 WZ011003 WZ011103 WZ011203
 WZ011303 WZ011403 WZ011503

BN04

WZ010304 WZ010404 WZ010504 WZ010604 WZ010704
 WZ010804 WZ010904 WZ011004 WZ011104 WZ011204
 WZ011304 WZ011404 WZ011504

BN05

WZ010305 WZ010405 WZ010505 WZ010605 WZ010705
 WZ010805 WZ010905 WZ011005 WZ011105 WZ011205
 WZ011305 WZ011405 WZ011505
 BN06
 WZ010306 WZ010406 WZ010506 WZ010606 WZ010706
 WZ010806 WZ010906 WZ011006 WZ011106 WZ011206
 WZ011306 WZ011406 WZ011506
 BN07
 WZ010307 WZ010407 WZ010507 WZ010607 WZ010707
 WZ010807 WZ010907 WZ011007 WZ011107 WZ011207
 WZ011307 WZ011407 WZ011507
 BN08
 WZ010308 WZ010408 WZ010508 WZ010608 WZ010708
 WZ010808 WZ010908 WZ011008 WZ011108 WZ011208
 WZ011308 WZ011408 WZ011508

(b) ポスト処理用入力データgdata_exp (実験データ用) の構成及びフォーマット

下表にgdata_expの構成を示す。

レコード番号	内 容
# 1	実験データファイル名
# 2	実験データのタイプ
# 3	実験データ図タイトル
# 4	実験データ横方向分割数
# 5	実験データ縦方向分割数
# 6	実験値の最小値、最大値
# 7	実験データプロット名称 (任意)
# 8	実験データのタグID

以下に、各レコードのフォーマットを示す。

#1	exp_filename	最大20文字	実験データファイル名 (実験データファイルの先頭には時間データを格納しておくこと)
#2	exp_type	整数	実験データのタイプ (現在は以下のものに固定) =1 : 被覆管表面温度図 =2 : ボイド率図 =3 : 流体温度図
#3	titlfg4	最大45文字	実験データ図タイトル
#4	num_exp	整数	実験データ横方向分割数 (本数)

#5	icells_exp	整数	実験データ縦方向分割数
#6	vemin, vemax		
	vemin	実数	実験データ最小値
	vemax	実数	実験データ最大値
#7	exp_tit	最大4文字	実験データプロット名称(任意)
#8	exp_tag	文字	実験データのタグID
	(注)	本カードはブランクをデリミターとしたフリーフォーマットである。 exp_tagはicells_exp個ある。 #7, #8はnum_exp回繰り返す。	

以下にgdata_expのファイルの例を示す。

```
J3624.EWS.TEST.DATA
1
SCTF  EXPERIMENT  DATA  PLOT
1
10
400.0 500.0
EXP1
TE0141C TE0241C TE0341C TE0441C TE0541C TE0641C TE0741C
TE0841C TE0941C TE1041C
```

(3) vtrexの実行

実行方法は、ディレクトリvtrac/binの中のvtrexを実行する。ここで、ユーザーは最低限計算用入力データTRACIN、図形表示用入力データTRCGDATをカレントディレクトリに用意しておかなければならない。また、必要に応じ、TRCF90等のデータファイルを用意する。

ここでvtrexを実行する。

```
% vtrex
```

以下に示すファイルが出力される：

```
TRCGRF      グラフィックファイル
TRCDMP      計算結果ダンプファイル
TRCMSG      メッセージファイル
TRCOUT      リスト出力ファイル
```

リスタート計算をする場合、前回の計算結果ダンプファイル（TRCDMP）をTRCRSTというファイル名にし、リスタート用の入力データファイルをTRACINとして計算を実行すればよい。

カレントディレクトリで複数回vtrexを実行した場合、その都度、上記のファイルが、オーバーライトしてしまうことに注意が必要である。

下表にvtrexの入出力ファイルの一覧を示す。

装置番号	ファイル名	ファイル形式	内容	備考
1	TRACIN	カードイメージ	入力データファイル	必須
2	TRCOUT	カードイメージ	プリンタ出力ファイル	リスト出力
5	TRCINP	カードイメージ	入力ワークファイル	ワーク
7	TRCMSG	カードイメージ	メッセージファイル	メッセージ出力
11	TRCGRF	バイナリー	グラフィックファイル	グラフィック用出力
12	TRCDMP	バイナリー	計算結果ダンプファイル	ダンプ出力
13	TRCRST	バイナリー	リスタート計算用入力ファイル	オプション(TRCDMPをRENAMEして作る)
15	TRCSO5	カードイメージ	SO Edit*用入力ファイル	オプション
16	TRCSO6	カードイメージ	SO Edit用出力ファイル	オプション
41	TRCGDAT	カードイメージ	画像表示用入力データ	必須
42	TRCGRSLT	カードイメージ	デバッグ用ファイル	デバッグ時の出力
61	TRCDBG	カードイメージ	デバッグ用ファイル	デバッグ時の出力
90	TRCF90	カードイメージ	凝縮モデル用入力ファイル	オプション
99	TRCF99	カードイメージ	計算時間制御用ファイル	オプション
6	-----		TTY用ファイル	モニター上に出力される

*：任意の物理量を出力するためのユーティリティ

現在の実行モジュールでは、TRCDBG及びTRCGRSLTについて実行終了時に削除されるようになって
いる。

(a) 同期処理用入力データTRCGDATの構成及びフォーマット

下表にTRCGDATの構成を示す。

レコード番号	内 容
# 1	出力頻度 (何ステップおきに表示するか)
# 2	被覆管温度図タイトル
# 3	表示する燃料棒の本数
# 4	燃料棒1本当りの分割数
# 5	被覆管表面温度最小値、最大値
# 6	燃料棒名称 (任意)
# 7	被覆管表面温度のコンポーネント指定データ
# 8	2つめの図のタイトル
# 9	2つめの図の表示するチャンネル数
# 10	2つめの図の各チャンネルの分割数
# 11	2つめの図の最小値、最大値
# 12	2つめの図の各チャンネル名称 (任意)
# 13	2つめの図のコンポーネント指定データ
# 14	3つめの図のタイトル
# 15	3つめの図の表示するチャンネル数
# 16	3つめの図の各チャンネルの分割数
# 17	3つめの図の最小値、最大値
# 18	3つめの図の各チャンネル名称 (任意)
# 19	3つめの図のコンポーネント指定データ

以下に、各レコードのフォーマットを示す。

#1	interval_draw	整数	何ステップおきに表示するか
#2	titlfgl	最大45文字	被覆管表面温度図のタイトル
#3	num_rod	整数	表示する燃料棒の本数 (最大20本*)

#4	icells_rod	整数	表示する燃料棒1本当たりの分割数(最大40分割*)
#5	vrmin, vrmax vrmin vrmax	実数 実数	被覆管表面温度最小値 被覆管表面温度最大値
#6	rod_tit	最大4文字	表示する燃料棒の名称(任意)
#7	被覆管表面温度のコンポーネント指定データ** (注) 本カードはicells_rod回繰り返す。 #6, #7はnum_rod回繰り返す。		
#8	titlfg2	最大45文字	2つめの図のタイトル
#9	num_void	整数	2つめの図で表示するチャンネル数の合計(最大20*)
#10	icells_void	整数	2つめの図で表示する各チャンネル当たりの分割数 (最大40分割*)
#11	vvmin, vvmax vvmin vvmax	実数 実数	2つめの図の最小値 2つめの図の最大値
#12	void_tit	最大4文字	2つめの図で表示する各チャンネルの名称(任意)
#13	2つめの図のコンポーネント指定データ** (注) 本カードはicells_void回繰り返す。 #12, #13はnum_void回繰り返す。		
#14	titlfg3	最大45文字	3つめの図のタイトル
#15	num_fluid	整数	3つめの図で表示するチャンネル数の合計(最大20*)
#16	icells_fluid	整数	3つめの図で表示する各チャンネル当たりの分割数 (最大40分割*)
#17	vfmin, vfmax vfmin vfmax	実数 実数	3つめの図の最小値 3つめの図の最大値
#18	fluid_tit	最大4文字	3つめの図で表示する各チャンネルの名称(任意)

#19 3つめの図のコンポーネント指定データ**

(注) 本カードはicells_fluid回繰り返す。

#12, #13はnum_fluid回繰り返す。

*- この制限は/vtrac/src/vtrac/inc内のインクルードy_graphi及びy_plot内の値を変更し、フルコンパイルすれば変更できる。

**- ショートエディットのフォーマット、以下にその概要を述べる。

コンポーネント指定は、次の変数を読み込むことにより行われる。

#* KIND, NUM, NCEL, LEV, NROD

個々の変数の内容を下表に示す。

変数名	フォーマット	内 容
KIND	A 6	出力する変数名 (REFLA/TRACコードのコモンラベル/PTAB/内の先頭の「L」を除いて指定する。)
NUM	I 6	コンポーネント番号
NCEL	I 6	KINDで指定した変数の配列要素
LEV	I 6	KINDで指定した変数のレベル番号 (VESSELのレベルデータの場合、またはファインメッシュデータの場合)
NROD	I 6	KINDで指定した変数のロッド番号 (VESSEL, COREのロッドデータの場合)

KINDに指定する主なショートエディット変数を表C. 1-C. 4に示す。

VESSELまたはCOREのロッドデータの内、HRFL, HRFV, IDHT, IHTF, RFT, RFTN, TCHFF, ZHTのファインメッシュデータを指定した場合、LEV には出力したい高さのレベル番号 (COREの時はセル番号) を指定すること。

また、この内 RFT, RFTNの時はNCELに径方向のノード番号を指定する。他のファインメッシュデータのNCELの指定は無視される。

[NCELの指定方法]

通常の変数 (ファインメッシュ以外のデータ) の場合、ショートエディットで出力したい変数が1次元配列の時は、変数の配列要素をそのままNCELとして指定すればよい。

例: セル番号2のポイド率を出力

KIND=ALPN

NCEL=2

出力したい変数が2次元配列の場合、NCELには1次元配列にしたときの配列要素を指定する。表C. 1-C. 4で示したデータの内、2次元配列はディメンジョンを(N1, N2)の形で表している。出力したい2次元配列要素を(K1, K2)とすると、1次元配列に直した時の配列要素NCELは以下の式になる。

$$NCEL = N1 * (K2 - 1) + K1 \quad (\text{但し, } N1 \geq K1, N2 \geq K2)$$

例: セル番号3 (K2=3), ノード番号2 (K1=2) の壁温を出力。但し、壁の熱伝達ノード数 (NODES) を5とする (N1=5)。

KIND = TWN

NCEL = N1 * (K2 - 1) + K1

$$= 5 * (3 - 1) + 2$$

$$= 12$$

以下にTRCGDATのファイルの例を示す。この例に示すファイルを使い本文の図8が出力された。

SCV661 CLAD TEMPERATURE (ROD) IN CORE

8

11

400. 1200.

ROD1

RFIN	1	7	4	1
RFIN	1	7	5	1
RFIN	1	7	6	1
RFIN	1	7	7	1
RFIN	1	7	8	1
RFIN	1	7	9	1
RFIN	1	7	10	1
RFIN	1	7	11	1
RFIN	1	7	12	1
RFIN	1	7	13	1
RFIN	1	7	14	1

ROD2

RFIN	1	7	4	2
RFIN	1	7	5	2
RFIN	1	7	6	2
RFIN	1	7	7	2
RFIN	1	7	8	2
RFIN	1	7	9	2
RFIN	1	7	10	2
RFIN	1	7	11	2
RFIN	1	7	12	2
RFIN	1	7	13	2
RFIN	1	7	14	2

以下、NRODの値が1ずつ増えてROD3からROD8に対する入力を行う。

SCV661 VOID FRACTION IN VESSEL (C & B & D)

11

19

0.0 1.0

CN01

ALPN	1	1	1	0
ALPN	1	1	2	0
ALPN	1	1	3	0
ALPN	1	1	4	0
ALPN	1	1	5	0
ALPN	1	1	6	0
ALPN	1	1	7	0
ALPN	1	1	8	0
ALPN	1	1	9	0
ALPN	1	1	10	0
ALPN	1	1	11	0
ALPN	1	1	12	0
ALPN	1	1	13	0
ALPN	1	1	14	0

ALPN	1	1	15	0
ALPN	1	1	16	0
ALPN	1	1	17	0
ALPN	1	1	18	0
ALPN	1	1	19	0
CNO2				
ALPN	1	2	1	0
ALPN	1	2	2	0
ALPN	1	2	3	0
ALPN	1	2	4	0
ALPN	1	2	5	0
ALPN	1	2	6	0
ALPN	1	2	7	0
ALPN	1	2	8	0
ALPN	1	2	9	0
ALPN	1	2	10	0
ALPN	1	2	11	0
ALPN	1	2	12	0
ALPN	1	2	13	0
ALPN	1	2	14	0
ALPN	1	2	15	0
ALPN	1	2	16	0
ALPN	1	2	17	0
ALPN	1	2	18	0
ALPN	1	2	19	0

以下、NCELの値が1ずつ増えてCN03からCN08,BPAS,DOW1,DOW2に対する入力を行う。

SCV661 LIQUID AND VAPER TEMPERATURE IN VESSEL.

16

19

400. 600.

CNV1

TVN	1	1	1	0
TVN	1	1	2	0
TVN	1	1	3	0
TVN	1	1	4	0
TVN	1	1	5	0
TVN	1	1	6	0
TVN	1	1	7	0
TVN	1	1	8	0
TVN	1	1	9	0
TVN	1	1	10	0
TVN	1	1	11	0
TVN	1	1	12	0
TVN	1	1	13	0
TVN	1	1	14	0
TVN	1	1	15	0
TVN	1	1	16	0
TVN	1	1	17	0
TVN	1	1	18	0
TVN	1	1	19	0

CNL1

TLN	1	1	1	0
TLN	1	1	2	0
TLN	1	1	3	0
TLN	1	1	4	0
TLN	1	1	5	0

TLN	1	1	6	0
TLN	1	1	7	0
TLN	1	1	8	0
TLN	1	1	9	0
TLN	1	1	10	0
TLN	1	1	11	0
TLN	1	1	12	0
TLN	1	1	13	0
TLN	1	1	14	0
TLN	1	1	15	0
TLN	1	1	16	0
TLN	1	1	17	0
TLN	1	1	18	0
TLN	1	1	19	0
CNV2				
TVN	1	2	1	0
TVN	1	2	2	0
TVN	1	2	3	0
TVN	1	2	4	0
TVN	1	2	5	0
TVN	1	2	6	0
TVN	1	2	7	0
TVN	1	2	8	0
TVN	1	2	9	0
TVN	1	2	10	0
TVN	1	2	11	0
TVN	1	2	12	0
TVN	1	2	13	0
TVN	1	2	14	0
TVN	1	2	15	0
TVN	1	2	16	0
TVN	1	2	17	0
TVN	1	2	18	0
TVN	1	2	19	0
CNL2				
TLN	1	2	1	0
TLN	1	2	2	0
TLN	1	2	3	0
TLN	1	2	4	0
TLN	1	2	5	0
TLN	1	2	6	0
TLN	1	2	7	0
TLN	1	2	8	0
TLN	1	2	9	0
TLN	1	2	10	0
TLN	1	2	11	0
TLN	1	2	12	0
TLN	1	2	13	0
TLN	1	2	14	0
TLN	1	2	15	0
TLN	1	2	16	0
TLN	1	2	17	0
TLN	1	2	18	0
TLN	1	2	19	0

以下、NCELの値が1ずつ増えてCNV3,CNL3からCNV8,CNL8に対する入力を行う。

C. 3 2次元図形表示システムの修正

(1) graexeの修正

graexeのソースファイルを修正して、新たにgraexeを作成する場合には、まずgraexeのソースプログラムが入っているディレクトリへ移動し(../vtrac/src/graphi)、ここでソースファイルの修正を行い、makeを実行する。

```
% make
```

ソースファイル名の変更、ロードモジュール名の変更等行いたい時は、本ディレクトリ内のMakefileを修正する必要がある。

(2) vtrexeの修正

vtrexeのフォートランソースファイルを修正して、新たにvtrexeを作成する場合次の手順で行う。

まず、vtrexeのソースプログラムの入っているディレクトリ(../vtrac/src/vtrac)に対してパスを設定する。

```
% setenv VTRACHOME ../vtrac/src/vtrac
```

この場合のディレクトリは、絶対パスで指定する。次にフォートランソースファイル格納ディレクトリへ移動し、フォートランソースファイルのコンパイルを行う。

```
% ../comp *****.f
```

一つ上位のディレクトリ、即ちVTRACHOME へ移動しリンクを行う。

```
% cd VTRACHOME
```

```
% linkvtr
```

これにより、../vtrac/binの下に新しいvtrexeが作成できる。もし実行モジュール名を変更して作成したい場合は、linkvtr の最終行のファイル名vtrexeを変更したい名前にする。また、フルコンパイルを行いたい場合はVTRACHOMEにおいて../cmpvtrを実行する。

表C. 1 ショートエディット変数 (主な形状データ)

変数名	ディメンション	内 容
DX	NCELLS	セル流路長
FA	NCELLS + 1	流路面積
FAVOL	NCELLS + 1	臨界流モデルで使用するセルの流路面積
FRIC	NCELLS + 1	圧力損失係数
GRAV	NCELLS + 1	重力項
HD	NCELLS + 1	水力学的直径
VLVOL	NCELLS + 1	臨界流モデルで使用する液相速度
VVVOL	NCELLS + 1	臨界流モデルで使用する蒸気速度
VOL	NCELLS	セル体積
WA	NCELLS	壁面積

表C. 2 ショートエディット変数 (主な物性値データ)

変数名	ディメンション	内 容
ROAN	NCELLS	空 気 密 度
ROLN	NCELLS	液 相 密 度
ROVN	NCELLS	蒸 気 密 度
ROM	NCELLS	混 合 密 度
CL	NCELLS	液 相 熱 伝 導 率
CV	NCELLS	蒸 気 熱 伝 導 率
CPL	NCELLS	液 相 定 圧 比 熱
CPV	NCELLS	蒸 気 定 圧 比 熱
HFG	NCELLS	蒸 発 潜 熱
SIG	NCELLS	表 面 張 力
TSAT	NCELLS	飽 和 温 度
TSSN	NCELLS	蒸 気 圧 力 の 飽 和 温 度
VISL	NCELLS	液 相 粘 性 係 数
VISV	NCELLS	蒸 気 粘 性 係 数
CPW	(NODES-1) , NCELLS	壁 の 比 熱
CW	(NODES-1) , NCELLS	壁 の 熱 伝 導 率
EMIS	NCELLS	壁 の 熱 放 射 率
ROW	(NODES-1) , NCELLS	壁 の 密 度

表C. 3 ショートエディット変数 (主な物理量)

変数名	ディメンション	内 容
ALPN	NCELLS	ボイド率
ALVN	NCELLS	液相界面熱伝達率
HLVN	NCELLS	蒸気界面熱伝達率
CIFN	2. (NCELLS+1)	界面剪断応力係数 $\left[\begin{array}{l} (1, *) : \text{Annular} \\ (2, *) : \text{Bubbly} \end{array} \right]$
CONCN	NCELLS	ボロンの冷却材に対する質量比 (ISOLUT=1のとき)
EAN	NCELLS	空気内部エネルギー
ELN	NCELLS	液相内部エネルギー
EVN	NCELLS	蒸気内部エネルギー
GAMN	NCELLS	蒸気生成率
HIL	NCELLS	液相壁面熱伝達率
HIV	NCELLS	蒸気壁面熱伝達率
PAN	NCELLS	空気の分圧
PN	NCELLS	圧 力
SN	NCELLS	構造材に析出したボロン質量 (ISOLUT=1のとき)
TLN	NCELLS	液相温度
TVN	NCELLS	蒸気温度
TWN	MODES. NCELLS	壁 温
VLN	NCELLS+1	液相速度
VVN	NCELLS+1	蒸気速度
VMN	NCELLS+1	混合速度
VR	NCELLS+1	相対速度
WFL	NCELLS+1	液相の壁面剪断応力係数

表C. 3 ショートエディット変数 (主な物理量) (続き)

変数名	ディメンション	内 容
WFV	NCELLS+1	蒸気の壁面剪断応力係数
HOL	NCELLS	外壁と液相の熱伝達率
HOV	NCELLS	外壁と蒸気の熱伝達率
QPPC	NCELLS	臨界熱流束
TOL	NCELLS	外壁の液相温度
TOV	NCELLS	外壁の蒸気温度

表C. 4 ショートエディット変数 (ファインメッシュデータ)

変数名	ディメンション	内 容
HRFL	NZMAX	ファインメッシュの液相熱伝達率
HRFV	NZMAX	ファインメッシュの蒸気熱伝達率
IDHT	NZMAX	熱伝達セルのID番号
IHTF	NZMAX	ファインメッシュの熱伝達形態フラグ
RFT	NODES, NZMAX	ファインメッシュのロッド温度 (前タイムステップ)
RFTN	NODES, NZMAX	ファインメッシュのロッド温度 (新タイムステップ)
TCHFF	NZMAX	ファインメッシュのCHF点での壁温
ZHT	NZMAX	熱伝達ノードの軸方向位置