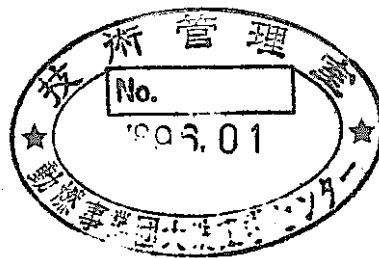


新流体力学コードAFDMの整備



1989年3月

富士通エフアイピー株式会社

複製又はこの資料の入手については、下記にお問い合わせ下さい。

〒311-13 茨城県東茨城郡大洗町成田町4002

動力炉・核燃料開発事業団 大洗工学センター

システム開発推進部 技術管理室

Inquiries about copyright and reproduction should be addressed to:
Technology Management Section, O-arai Engineering Center, Power Reactor
and Nuclear Fuel Development Corporation 4002, Narita O-arai-machi Higashi-
Ibaraki-gun, Ibaraki, 311-14, Japan

動力炉・核燃料開発事業団 (Power Reactor and Nuclear Fuel Development
Corporation)

新流体力学コードAFDMの整備*

小寺慶一**

要 旨

米国ロスアラモス国立研究所(LANL)で各国研究機関(米国, 西ドイツ, イタリア, フランス及び日本)の国際協力の下, 新流体力学コードAFDM(Advanced Fluid-Dynamics Model)が開発された。このAFDMコードは, 高速炉安全研究のために, 流体力学計算の精度向上, 適用分野の拡大を目指して開発されたものである。

本報告書は, このAFDMコードの動燃大洗工学センター大型計算機への導入・整備作業をまとめたものである。動燃のFACOM機への導入に当っては, LANLで開発されたCRAY版のソース・プログラムを基に, ソース変換作業を行った。

この結果, AFDM最新版が動燃計算機システム上において正常に作動するようになった。動燃におけるテスト計算の結果は, LANLにおけるものと比較して満足のゆくものであった。本報告書はまた, この導入作業に平行して実施したベクトル計算機FACOM-VP100上での効率化作業についても記述している。

* 本報告書は動燃炉・核燃料開発事業団との契約の下に実施された研究の成果である。

** 富士通エフ・アイ・ピー株式会社

March, 1989

Installation of AFDM, An Advanced Fluid-Dynamics Model*

Keiichi Koderu **

Abstract

At Los Alamos National Laboratory (LANL), An AFDM (Advanced Fluid-Dynamics Model) code was developed under international cooperation of the research institutes in the USA, West Germany, Italy, France and Japan. For the safety analysis of FBRs, it was developed to increase accuracy in fluid-dynamics calculations and to extend its applicability.

This report summarizes the installation of the AFDM code on the computer system at the Oarai Engineering Center, Power Reactor and Nuclear Fuel Development Corporation (PNC). In order to install it on the FACOM computer system at PNC, it was necessary to correct various parts of the source program from the CRAY version developed at LANL.

The latest version of AFDM has been successfully installed on the computer system at PNC. The results of the sample calculations performed at PNC were satisfactory when they were compared with those at LANL.

This report also describes the optimization of the code on the vector processor FACOM-VP100 at PNC made in the process of the installation.

* Work performed under contract with the Power Reactor and Nuclear Fuel Development Corporation.

** Fujitsu FACOM Information processing Co.

目 次

要 旨	i
Abstract	ii
目 次	iii
図 目 次	v
表 目 次	vi
1. 序 論	1
2. AFDMの概要	2
3. AFDM整備作業概要	5
3.1 動燃導入の経緯	5
3.2 Y8804版	5
3.2.1 HISTORIAN ライブラリの作成	5
3.2.2 コード内容の調査	6
3.2.3 FORTRAN プログラムの変換	6
3.2.4 システムライブラリの変換・作成	6
3.2.5 サンプル問題のテスト計算	7
3.3 Y8808版	9
3.4 Y8811版	9
3.5 Y8901版及びY8903版	10
3.6 ベクトル計算機用効率化	10
3.6.1 ベクトル化ディレクティブ	11
3.6.2 CFTライブラリ	11
3.6.3 AFDM動的分析	12
3.6.4 SRCH3Dルーチンの効率化	12
4. テスト計算結果	19
4.1 ロードモジュール作成	19
4.1.1 前処理コード	19
4.1.2 HISTORIAN オプション	19
4.1.3 PNC用AFDMアッドファイル	20
4.1.4 PNC用AFDMコレクションセット	20
4.1.5 ロードモジュール作成法	21
4.2 テスト計算	22
4.2.1 テスト計算結果	22
4.2.2 結果比較	23
4.2.3 実行環境	24

5. 結 論	62
謝 辞	63
参考文献	64
Appendix A SBSAME ファイルコーティリティの導入	65
Appendix B PULライブラリの整備	69

目 次

図 3.1	A F D M プログラム変換例 (1)	16
図 3.2	A F D M プログラム変換例 (2)	16
図 3.3	A F D M プログラム変換例 (3)	16
図 3.4	A F D M プログラム変換例 (4)	16
図 3.5	A F D M プログラム変換例 (5)	17
図 3.6	A F D M プログラム変換例 (6)	17
図 3.7	A F D M プログラム変換例 (7)	17
図 3.8	A F D M プログラム変換例 (8)	17
図 3.9	S R C H 3 D ソースリスト	18
図 3.10	I C V M G T の簡素化 (in SRCH3D)	18
図 3.11	G A T H E R ルーチンの展開 (in SRCH3D)	18
図 4.1	P N C 用 A F D M アッドファイル	26
図 4.2	P N C 用 A F D M コレクションセット	27
図 4.3	A F D M ロードモジュール作成 J C L 例 (scalar 版)	32
図 4.4	A F D M ロードモジュール作成 J C L 例 (V P 版)	33
図 4.5	A F D M 出力例 (Y 8 8 1 1 版 M 1 N P A 1 ケース)	34
図 4.6	A F D M 出力例 (Y 8 8 1 1 版 M 1 P A 1 ケース)	36
図 4.7	A F D M 出力例 (Y 8 8 1 1 版 M 2 1 A 1 ケース)	38
図 4.8	A F D M 出力例 (Y 8 8 1 1 版 M 1 N P S 1 ケース)	40
図 4.9	A F D M 出力例 (Y 8 8 1 1 版 M 1 P S 1 ケース)	42
図 4.10	A F D M 出力例 (Y 8 8 1 1 版 M 2 1 S 1 ケース)	44
図 4.11	A F D M 出力例 (Y 8 8 1 1 版 K 1 N P S 1 ケース)	46
図 4.12	A F D M 出力例 (Y 8 8 1 1 版 K 1 P S 1 ケース)	48
図 4.13	A F D M 出力例 (Y 8 8 1 1 版 K 2 1 S 1 ケース)	50
図 4.14	A F D M 出力例 (Y 8 9 0 1 版 M 2 1 S 1 ケース)	52
図 4.15	A F D M 出力例 (Y 8 9 0 1 版 M 1 P S 2 ケース)	54
図 4.16	A F D M 出力例 (Y 8 9 0 1 版 M 2 1 A 3 ケース)	56
図 4.17	A F D M 出力例 (Y 8 9 0 3 版 M 2 1 A 1 ケース)	58
図 4.18	A F D M 実行用 J C L 例 (scalar)	60
図 4.19	A F D M 実行用 J C L 例 (V P)	60
図 4.20	A F D M M 2 1 S ケース出力リスト比較	61
図 A.1	P N C 用 H Y D S E S アッドファイル	66
図 A.2	P N C 用 H Y D S E S コレクションセット	67
図 A.3	U P D A T E 作成 / 実行 J C L	68
図 B.1	P N C 用 P U L コレクションセット	70
図 B.2	P U L 作成 J C L	71

表 目 次

表 2. 1	A F D M 取扱い成分	4
表 3. 1	A F D M システムルーチン	13
表 3. 2	A F D M 図形処理ルーチン (除 P U L)	13
表 3. 3	A F D M のコスト分布とベクトル化率 (M I N P A - L W B P ケース)	14
表 3. 4	A F D M のコスト分布とベクトル化率 (M I N P S - L W B P ケース)	14
表 3. 5	S R C H 3 D のベクトル化テスト結果 (M I N P S - L W B P ケース)	15
表 4. 1	A F D M テスト結果ケース	25
表 4. 2	A F D M 実行環境	25

1. 序 論

高速増殖炉の安全解析のため、米国ロスアラモス国立研究所(LANL)では各種シミュレーションコードを研究してきた。この中でLANLは、高速増殖炉の炉心崩壊事象の核的・流体力学的挙動をモデル化するため、SIMMERコードを開発した。このSIMMERコードは、第2版(SIMMER-II)まで開発されており〔Ref.1〕、動燃(動力炉・核燃料開発事業団)を含めて国際的に広く利用され、高速増殖炉の安全解析コードとして最も有用なものの一つとなっている。動燃事業団では1980年にLANLよりSIMMERを導入した。以後改良版が数回送付され、現在は、Version 2.12になっている。このSIMMER-IIコードは、動燃においても高速増殖炉の安全解析に有効に利用されている。

一方、多くのSIMMER-IIコードの利用の結果、いくつかの問題点も指摘されてきた。特に、熱・質量・運動量交換モデル、ラッパー管・燃料ピンの再配置、核計算部の正確かつ効率的な結合、流体力学部の数値計算法、多成分系のEOS(状態方程式)に関するものである。しかしながら、これらの問題を全て解決できるSIMMER-IIの改訂版、SIMMER-IIIコードの構築のためには、技術的問題を検討し、開発環境の整備が成されなければならない。

そこで、1985年、いくつかの問題を解決するための新プログラムの開発が提案された。SIMMER-IIは非常に複雑なプログラムであったため、この新プログラムは、SIMMER-IIにおける問題点のいくつかを解決し、かついくつかの重要な応用問題にも有効である様な比較的単純なコンピュータコードとなるように設計された。この新コードの開発は、動燃を含め国際協力で行なわれたものであり、AFDM(Advanced Fluid-Dynamics Model)と名付けられた〔Ref.2〕。

本報告書は以上の背景の下で実施されたAFDMコードの動燃大洗工学センター大型計算機(FACOM)への導入・整備作業について記述している。

第2章にAFDMの概要を記述している。また、本導入作業は、AFDMがまだ開発途上であったことと、LANLのCRAY機で開発されているために、PNCのFACOM機への導入に際し、一旦PNCで行った導入結果を再度LANLに報告し、新版のAFDMに反映させるという形態がとられた。このあたりの状況を含んだ整備作業概要を第3章にまとめている。一方、AFDMは、膨大なシミュレーション計算の高速化を図って、ベクトルプロセッサ向けに開発されている〔Ref.3〕。但し、前述の様に、AFDMはCRAY機用に開発・最適化されている。従って、FACOM-VP機で実行する際には、ベクトル化指示子等多少の修正を行う事が望ましい。このVP用の修正も第3章にまとめる。第4章では本導入作業のテスト結果を記述している。最後に、本研究で得られた結論を第5章にまとめる。

2. AFDMの概要

AFDMコードは、前章にある様に、長期的にはSIMMER-IIIの開発を目指して、現在のSIMMER-IIの問題点を解決し、大幅な精度向上と適用分野の拡大を達成するための技術開発及びモデル・手法のテストを目的として開発されたものである。

このAFDMコードは、3速度場、座標空間2次元、多相、多成分のEulerian流体力学コードである。各成分は表2.1の様に扱われている。

AFDMコードはこの表2.1にある様に構造材、粒子、2種の液体、3種の蒸気の7密度成分から成っている。構造材は不動であり、各々のセルで任意の体積を占めている。可動成分は3つの物質にまとめられている。物質1（通常は燃料）は固体粒子、液体、または蒸気として存在することができる。物質2（通常は冷却材）は、液体または蒸気の状態が存在すると仮定され、物質3は非圧縮性ガスである。7つの密度成分は、5つのエネルギー成分に割り当てられる。即ち構造材、粒子、それぞれの液体、ガス混合物である。

また、AFDMには3つの可動速度場がある。粒子と液体燃料は第1の速度で動き、液体冷却材は第2の速度に割り当てられ、蒸気は第3の速度を持っている。密度・エネルギー成分（さらにより多くの運動量場）数の拡張は可能であるが、提案されたモデル及び手法を開発・テストするためには、現在の選択で十分であると判断された。

ここで、表2.1の質量、エネルギー、運動量成分のリストに加えて、3種の追加情報が必要となることを付け加えておく。まず、高次差分法を用いた対流計算と、内部EOS計算の際、個々の蒸気密度成分に対する内部エネルギー e_{Gm} が必要となる。これらは、蒸気場の温度を仮定することによって得られる一時的変数である。第2に、蒸気・凝縮の計算のために、気液界面での物性値が必要となる。蒸発・凝縮が起こる際、この界面における物性値は、質量交換を行っている物質の蒸気分圧に対応した飽和状態にあると仮定されている。これらの飽和状態は、EOSより得られる。第3にAFDMにおける連続体運動量成分と不連続体運動量成分間の単位体積当りの境界面積 a_f^c モデル化している。ここで f は表2.1における運動量添字を用いて、L1/L2, L1/G, L2/Gと表す。この対流のモデル化により、Eulerianコード形式ではあるが、現象論的履歴を近似的に辿ることが可能である。

AFDMで使われている質量・運動量・エネルギーの3保存則から導かれる簡易差分方程式は基本的にはSIMMER-IIの方程式の3速度場への拡張である。基本的な流体力学の解法はSemi-Implicit法であるが、時間進行に対してセル内の熱及び質量移行過程とセル間の対流とを分離して計算する手法を用いている。その結果、AFDMのアルゴリズムでは、1タイムステップの計算が4ステップで構成されることになり、4ステップ法とも呼ぶ。ステップ1ではセル内物理現象を計算し、ステップ2では質量・運動量・エネルギーの3保存式から導かれた差分式により、密度と速度とエネルギーの各成分に対してタイムステップ終了時の予測値を陽に計算する。ステップ3ではいわゆる圧力反復により速度と圧力を計算してタイムステップ終了時に互いに矛盾がないよう値を決定すると共に、感度の高い変数（固体粒子とその液相物質の密度の和、冷却材の密度、気相密度、気相エネルギー）についてステップ2での予測値を修正し誤差を0に近づける。ステップ4では

その他の変数についてタイムステップ終了時の値を計算し直し、質量の対流と一貫性をもたせるよう運動量とエネルギーを対流させる。

一方、AFDMにおける流体力学計算では密度及び温度を内部エネルギー及び圧力に関係付けるための状態方程式(EOS)が必要となり、これは上記各計算ステップから呼ばれる。AFDMのEOSモデルでは、単に熱力学的状態量を計算するだけでなく、ステップ3の圧力反復及びステップ1の蒸発凝縮モデル等の計算で必要となる各微分係数も計算される。AFDMは、このEOSに対して2種類のオプションをもっている。まず、簡易解析的EOS(SAEOS)モデルであるが、これはSIMMERコードのEOSモデルを更に簡易化したものである。一方、詳細EOSモデルでは、テーブル形式のEOSが使用される。これは、LANLのEOSデータライブラリであるSESAMEを使用するため、SESAME EOSと呼ぶ〔Ref.4〕。SESAMEは70種の物質に対するデータテーブルとこれにアクセスするためのサブルーチン群からなるが、AFDM側ではデータを検索・補間して、これらを必要な情報に変換するためのインターフェイスモデルが開発されている。このインターフェイスには、例えば、多成分混合物からなるセル状態の機械的平衡などの物理的整合性を保つためのモデルも含まれている。

最後にコード整理法について述べておく。AFDMコードは、LANLのCRAY-1AでCTSS(Cray Time-Sharing System)を利用して開発された。プログラムはANSI X3.9-1978 FORTRANで書かれている。コードの保守・修正・改正にはOp-Code, Inc.のHISTORIAN〔Ref.5〕が使用された。従って利用者はAFDMの導入に際し、HISTORIANか同様の機能をもつCDC社のUPDATE〔Ref.6〕を用意する必要がある。PNCでは前処理コードとしてHISTORIANを利用している。

表 2.1 A F D M 取扱い成分

<u>Components</u>	<u>Densities</u>	<u>Internal Energies</u>	<u>Velocities</u>
Structure	$\bar{\rho}_s$ ($\bar{\rho}_S$)	e_s	-
Solid Particulate	$\bar{\rho}_p$ ($\bar{\rho}_P$)	e_p	$\bar{v}_p = \bar{v}_{L1}$
Liquid	$\bar{\rho}_{l1}$ ($\bar{\rho}_{L1}$)	e_{L1}	\bar{v}_{L1}
	$\bar{\rho}_{l2}$ ($\bar{\rho}_{L2}$)	e_{L2}	\bar{v}_{L2}
Vapor	$\bar{\rho}_{g1}$ ($\bar{\rho}_{G1}$)		
	$\bar{\rho}_{g2}$ ($\bar{\rho}_{G2}$)	e_g	\bar{v}_g
	$\bar{\rho}_{g3}$ ($\bar{\rho}_{G3}$)		
Schematic Representation	$\bar{\rho}_m$	e_r	\bar{v}_q

Assume: The structure EOS is limited.

$\bar{\rho}_p$, $\bar{\rho}_{L1}$, and $\bar{\rho}_{G1}$ compose Material 1.

$\bar{\rho}_{L2}$ and $\bar{\rho}_{G2}$ compose Material 2.

$\bar{\rho}_{G3}$ is a noncondensable gas and is Material 3.

In the absence of other information,

\bar{v}_{L1} can be considered as belonging to Momentum Field 1,

\bar{v}_{L2} can be considered as belonging to Momentum Field 2, and

\bar{v}_g can be considered as belonging to Momentum Field 3.

3. AFDM整備作業概要

3.1 動燃導入の経緯

当初AFDMの動燃への導入は、4月に暫定版をLANLより入手し、これに対してソース変換、エラー修正を行い、LANLに結果を報告し、改めて最終版のAFDMを10月頃入手するとの予定であった。

ところが、動燃側で行った変換作業の検討結果を包含する修正を含めて、合わせて4回AFDM修正版がLANLより送られてきた。さらに、この4回目に送付されたものには、最終版とそれに対する修正（つまり、真の最終版）が含まれていた。以下にLANLより送付されたAFDMの版を示しておく（カッコ内はLANLで呼ばれていた版名）。

- i) 88年4月: Y8804版 (W版)
- ii) 88年8月: Y8808版
- iii) 88年11月: Y8811版 (Y版プラス追加修正)
- iv) 89年1月: Y8901版 (Z版)
- v) 89年1月: Y8903版 (Z版プラス追加修正)

以下の各節では、混乱を避けるために上記Y8×××版を使用する。Y8903版が現在の最終版となる訳である。この各版の導入作業概要を以下の各節に、FACOM-VP用のAFDM効率化作業内容を3.6節にそれぞれ記述する。

一方、AFDMは前章で述べた様に、EOSモデルに、SESAME EOSテーブルを使用する。このSESAMEデータテーブルと共に、このSESAMEデータを使用するためのサブルーチンパッケージHYDSES [Ref.7]もLANLより送られてきた。今回は、このうちAFDMで使用する際に必要となるルーチン群を、AFDM同様動燃大洗工学センター計算機に導入した。この導入作業内容を Appendix A に記述する。

3.2 Y8804版

3.2.1 HISTORIANライブラリの作成

AFDMはLANLで作成され、PNCへは1988年4月、HISTORIANソースの形で送られてきた。送られてきたAFDMコードのHISTORIANソースは第T版であった。また、このT版に対してLANLにおける修正が成されており、これもまたHISTORIANのコレクションセット（修正情報リスト）の形でPNCに送られてきた。これは、W版作成用のコレクションセットであった。以上の2ファイルを用いてHISTORIANを実行し、AFDMのHISTORIANライブラリを作成した。

3.2.2 コード内容の調査

前項で作成したAFDM HISTORIANライブラリを用いてAFDM FORTRANソースを作成する。このHISTORIANライブラリはあくまでCRAY用であるため、PNC FACOMシステムで作動させるためには、いくつかの変換作業が必要となる。この変換作業のため、FORTRANコンパイラ等の各種ツールを用いてAFDMコードの調査分析を行った。

この調査の結果、オリジナルのAFDMコードには、いくつかの文法エラー、LANLで行われた修正ミス等が存在することが確認された。また、未定義の外部ルーチンも存在していることが明らかとなった。この外部ルーチンは、CRAY FORTRAN組み込み関数、CTSSシステムルーチンの他、グラフィックルーチンも存在していた。従ってPNCへの導入に当たっては、以下の作業が必要となる。

- i) FACOM用FORTRANへの変換（エラー修正含む）
- ii) システムライブラリの変換・作成

3.2.3 FORTRANプログラムの変換

前項の調査に基づいてCRAY FORTRAN→FACOM FORTRANの変換作業を行った。この作業は、HISTORIANのコレクションセットの形で行った。

このFORTRANプログラム変換内容は以下のとおり。

- (1) プログラム文の修正（パラメーターの削除）
- (2) 7文字以上のコモンブロック名を6文字に修正
- (3) 論理定数の修正 $\left\{ \begin{array}{l} \cdot T. \rightarrow \cdot TRUE. \\ \cdot F. \rightarrow \cdot FALSE. \end{array} \right.$
- (4) 文字定数の修正（" Z " → ' Z '）
- (5) IFブロックへの飛込みの回避
- (6) 8バイト整数型データへの対応
- (7) FORTRAN組み込み関数の対応（LOC, SECOND, CLOCK）
- (8) -CRAY/CTSS版のエラー修正（HISTORIAN指示子エラー等）

このうち(6)～(8)については、前項の調査では発見できず、サンプル問題を実行するなかで明らかとなり、対応した。以上の(1)～(8)の例をそれぞれ図3.1～3.8に示す。なおこのうち(7)のLOCについては、アセンブラルーチンをロードモジュール作成の際結合することで対処することとした。

3.2.4 システムライブラリの変換・作成

AFDMが必要とするシステムルーチンは、CRAY/CTSS特有のルーチンと、図形処理ルーチンである。このうち図形処理ルーチンは、プロットユーテ

ィリテイライブラリ (PUL) と呼ばれる図形処理ルーチン群が用いられている [Ref. 8] このPULは、図形処理コードT6Pの導入 [Ref. 9] の際に整備されたものを基本として、本AFDM導入に当って若干の再整備を行った。このPULの整備作業概要をAppendix Bに記述する。ところが、このPULには図形処理のうちの初期処理、終処理ルーチンは含まれていない。オリジナルのAFDMは、この初期処理終処理及び文字出力ルーチンにCGS (Common Graphics System) と呼ばれるLANL独自の図形処理ルーチンを使用していた。従って、PNC導入に当ってはこのPUL以外の図形処理ルーチンを、PNC利用可能な図形処理ルーチンに変更しなければならず、これを図形処理パッケージDISSPLA [Ref. 10] に変換した。

以上のAFDMのCRAY/CTSSシステムルーチンと図形処理ルーチンの変換方法をそれぞれ表3.1と3.2に示す。この作業はHISTORIANのコレクションセット (修正情報リスト) とアッドファイル (追加モジュール) の形で行った。

3.2.5 サンプル問題のテスト計算

以上の各項の変換を行ったPNC用AFDMソースプログラムからロードモジュールを作成し、LANLよりソースプログラムと共に送られてきたサンプル問題のテスト計算を実施した。テストケースは、AFDMのオプションの組 (4.1.2節参照) が6種類サンプル問題が4種類の計24ケースであった。このテストの計算中、いくつかのバグを発見した。これに対する修正を前述のPNC用コレクションセットに追加した。この修正を施したAFDMのテスト計算の結果、以下の事項が明らかとなった。

- 24ケース中、6ケースを除いてほぼLANLの結果と一致した。
- 一致しないケースについては、以下の様な現象が現われた。
 - ㉑ 計算途中で指数オーバーフローを起こした。
 - ㉒ 繰り返し計算が収束しないケースが起こった。
 - ㉓ AFDMは、ステップ3において、最も速い計算法 (行列解法) をコード内部で選択するが、この計算法が異った。
- 上記の相違は、以下の様な原因が考えられる。
 - ㉑ FOCOMは定数のとりうる範囲がCRAYより小さい (CRAYは 10^{2466} までに対し、FACOMは 10^{75} まで)
 - ㉒ CRAY-FACOM間には数値精度、組込み関数の精度の違いが存在する。この繰り返し計算は微量の複雑な計算、判定を行っているため精度上の誤差が増幅する可能性が大きい。
 - ㉓ ㉑の相違の他、CRAY-FACOM間ではハードウェア性能の違い、コンパイラによる最適化の違いがあるため、消費するCPU時間が異なる。

- これらの問題点の解決のため、以下の処置を行った。
 - (i) このオーバーフローを回避するため、ソース上の修正やSSL (Ref.11)の利用を行った。
 - (ii) LANLに報告し、プログラムソース上で解決できなかいかが相談した。
 - (iii) 計算法が異っていても、最終的な計算結果への影響は小さいので特に問題にはしない。

以上のY8804版のテスト結果をLANLに送り、その後のLANLでのAFDMの改訂に反映させた。

3.3 Y 8 8 0 8 版

1988年8月、LANLよりAFDM最新版（Y 8 8 0 8 版）用コレクションセットが送られてきた。これを用いて新バージョンのHistorian ライブラリを作成し、これに対応するよう、PNC用コレクションセットを修正した。このPNC用コレクションセットの修正は、主にLANLより送られてきたコレクションセットと前節で述べたY 8 8 0 4 版用PNC用コレクションセットとの整合性をとったものである。

これらのファイルを用いてAFDM新バージョン（Y 8 8 0 8 版）のロードモジュールを作成し、サンプル問題のテスト計算を実施した。この結果、Y 8 8 0 4 版で起こっていたEOS計算部の繰り返し計算未収束問題が解決されていることを確認した。

3.4 Y 8 8 1 1 版

AFDMのY版（Y 8 8 1 1 版）がLANLより送られてきたのは、1988年11月である。送られてきたものは、① AFDM Y版のHISTORIANソース、② 追加修正のためのコレクションセット、③ K f Kのメンバーにより作成されたK f K固体EOSコレクションセット、④ 固体EOS用アッドファイル2ファイル等のファイルである（その他入力データ、PNLの修正版等）、これらのファイルをそれぞれデータセット

① ' T 2 0 1 # . A F D M . Y 8 8 1 1 . H F O R T '
 ② ' T 2 0 1 # . A F D M . Y 8 8 1 1 . L A N L C S . F O R T (Y T O Z) '
 ③ ' T 2 0 1 # . A F D M . Y 8 8 1 1 . L A N L A F . F O R T (S E O S) '
 ④ { ' T 2 0 1 # . A F D M . Y 8 8 1 1 . L A N L A F . F O R T (E O S U P) ' ,
 ' T 2 0 1 # . A F D M . Y 8 8 1 1 . L A N L C S . F O R T (V A P D O M) ' }

に格納した。これを用いて新バージョンのHISTORIANライブラリを作る訳であるが、今回、Y 8 8 1 1 版ではK f K固体EOS用の修正を含まないものを標準とするため、上位2ファイル、①と②を使用してHISTORIANライブラリを作成した。続いて前節までと同様にPNC用コレクションセットを作成した。Y 8 8 1 1 版も前回同様、基本的には前版のPNC用コレクションセットとの整合化及びLANLコレクションセットのエラー修正で済む。また、今回は、固体EOS用のコレクションセットに対する修正も必要となるため、これに対するPNC用コレクションセットも作成した。

これらを用いてAFDM新バージョン（Y 8 8 1 1 版）のロードモジュールを作成し、サンプル問題のテスト計算を実施した。このロードモジュールの作成の際、一緒に送られてきたPUL改訂版を使用・結合した。また、テスト計算の実施の際も、共に送られてきたSESAMEファイルの改訂版を使用した。今回のテスト計算は、AFDMオプションの組9種類、サンプル問題3種類の計27ケースであった。このテスト計算の結果、やはりCRAY←→FACOM間にはわずかではあるが、相違があり、特にあるケースでは、蒸発・凝縮計算で、繰り返し計算未収束問題が発生した。しかし

この1ケースを除いては、全て正常終了することを確認した。

3.5 Y8901版及びY8903版

1989年1月、LANLよりAFDM Z版作成用コレクションセットと、これに対する追加コレクションセットが送られてきた。さらにこのZ版用のアッドファイルも送られた。上記3ファイルをそれぞれデータセット

'T201#. AFDM. Y8901. LANLCS. FORT'

'T201#. AFDM. Y8901. LANLCS. ADD. FORT'

'T201#. AFDM. Y8811. LANLAF. FORT (ARTR)'

に格納した。今回のZ版をY8901版とし、HISTORIANを実行して、HISTORIANライブラリを作成し、データセット

'T201#. AFDM. Y8901. LANL. HISLIB'

に格納した。さらに、PNC用のロードモジュールを作るために、PNC用コレクションセットを作成した。このコレクションセットは、Z版(Y8901版)用と、追加修正版(Y8903版)用との2種類作成し、このうちY8903版用コレクションセットをデータセット

'T201#. AFDM. Y8903. PNCCS. FORT'

に格納した。

これらを用いてAFDM最終版(Y8901版及びY8903版)のロードモジュールを作成し、サンプル問題のテスト計算を実施した。なお、この最終版のロードモジュールの作り方については第4章に記述する。このテスト計算は、Y8901版3ケース、Y8903版1ケースであった。このテスト計算の結果、前Y8811版で発生していた蒸発・凝縮繰り返し計算未収束問題は、解決していることを確認した。しかしなおCRAY \longleftrightarrow FACOM間で相違があり、

- 選択する行列解法
- 過度計算のタイムステップ幅
- 圧力計算繰り返し回数

の違いとなって現われている。

3.6 ベクトル計算機用効率化

AFDMは、膨大なシミュレーション計算の高速化処理を可能にするため、ベクトルプロセッサ向けに開発されている。但し、CRAY機用に開発・最適化されているために、FACOM-VP機で実行する際には多少手を入れる事が望ましい。前節までの導入作業に平行して、このVP用の効率化作業を行っており、本節ではこの作業についてまとめておく。

3.6.1 ベクトル化ディレクティブ

AFDMはCRAY FORTRANで開発されているため、CFT (CRAY社のFORTRAN) のベクトル化ディレクティブを用いて実行効率を改善している。そこでPNC導入に当って、このCFTベクトル化ディレクティブを、これらと同等な機能を持つ“FACOM OSIV/F4 MSP FORTRAN 77/VP”の最適化制御行(OCL) [Ref, 12] に変換した。

変換方法は以下のとおり。

```

CDIR$  IVDEV          CDIR$  SHORTLOOP
      ↓
*VOCL  LOOP, NOVREC          ↓
                                *VOCL  LOOP, REPEAT(64)
    
```

3.6.2 CFTライブラリ

CRAYでは、複雑なIF文を含むDOループはベクトル化できない場合がある。このため、AFDMはCRAY社の提供するCFTライブラリ固有の手法によりベクトル化している部分がある。このCFTライブラリはCRAY FORTRAN特有のものであるため、AFDMには非CRAY機用にCFTライブラリ模擬ルーチンがFORTRANソースの形で用意されている。このうち、DOループ内のIF文を除く関数の模擬ルーチンは、当初(Y8804版)では、

```
CVMGT, CVMGP, CVMGZ
```

等10数種類が存在していたが、Y8811版以降では、実際に使用している、

```
CVMGT, ICVMGT, LCVMGT
```

の3種類のみとなった。これらはDOループ中の条件処理で使用されているが、CRAY用には、IF THEN ELSE型のFORTRANファンクションで置き換えられている。ところがDOループ中でサブルーチンが呼び出されるとコンパイラによるベクトル化ができない。従って該当ループをベクトル化するために各呼び出し部分にコード・ブロックを展開する必要がある。これを自動的に行う機能としてFACOM FORTRAN 77/VPコンパイラにはPIオプションが用意されている。(V10L20以降) [Ref. 12]

その他のCFTライブラリとして、

- 収集・拡散ルーチン GATHER/SCATTER
- 最大(最小)値検索ルーチン ISAMAX/ISAMIN
- 総和計算ルーチン SDOT

が用いられている。これらは出現頻度が少ない上、AFDMがFORTRAN代替ルーチンを用意しており、しかもそれぞれこの代替ルーチン内にDOループをもっていることもあって、あえて代替ルーチンの利用するのみにとどめておいた(PIオプションを使用すれば呼出しのオーバーヘッドはなくなるが、それでも上位ループがベクトル化される可能性は少ない)。

3.6.3 AFDM動的分析

AFDM実行の際、CPU時間の高消費部を発見するために、AFDMの動的分析を行った。合わせてベクトル化の効果を見るため、ベクトル化率の測定を行った。この測定に当たって、“FORTUNE”〔Ref,13〕及び“会話型ベクトライサ”〔Ref,14〕を使用した。測定したAFDMのケースは、MINPA-LWEPケースと、MINPS-LWEPケースの2ケースである。

この動的分析の結果の各サブルーチン毎のCPUコスト分布とベクトル化率を、高コスト10モジュールについてそれぞれ表3.3, 3.4に掲げる。

これらの表より、MINPA（簡易解析的EOSモデル）ケースではほぼベクトル化が成されており、プログラムソースのチューニングで、効率化を図るのは相当の困難が予想される。一方、MINPS（SESAME EOSモデル）ケースでは上位ルーチンにベクトル化率が小さいものがあり、検討の余地があることが確認された。但し、WLCHとPICCONルーチンは図形処理ライブラリPULであり、元々ベクトル化されていない。

3.6.4 SRCH3Dルーチンの効率化

前項で明らかになった様に、MINPSケースは、サブルーチンSRCH3Dは、コスト分布64%の高さであるにもかかわらず、ベクトル化率わずか4.6%との結果となった。そこでこのSRCH3Dルーチンの効率化を検討した。

このSRCH3Dは、図3.9にある様に、変数Yが、昇順の配列Xの中でどの位置にあるかを求め、合わせてその差を計算している。この処理におけるベクトル化阻害の要因は、最内ループ（DO 10 loop）のオーバーヘッド（VPコンパイラが、ループ内の処理が複雑すぎると判断した事象）であった。そこでこのオーバーヘッドを除くため、CFTライブラリ模擬ルーチン、ICVMGTの部分を簡素化して、図3.10の様に書き換えた。こうしたところ、このループもベクトル化され、SRCH3Dのベクトル化率は99.5%にはね上がった。

一方、このSRCH3DにはCFTライブラリルーチンGATHERも存在している。このGATHERの呼び出しによるオーバーヘッドを軽減するために、この部分を展開し、図3.11の様に書き換えた。

以上の2つの修正を施したSRCH3Dと、ICVMGTの簡素化のみの修正及びオリジナルのSRCH3Dルーチンをそれぞれコンパイルしてテストした結果を表3.5に示す（MINPS-LWEPケース）この結果、ICVMGTを簡素化すれば、かなりの高速化が図られることがわかった（約3倍）

一方、GATHER展開の方は、SRCH3DルーチンのCPU時間自体は30 μ sec程速くなるが、AFDM全体のCPU時間は却って遅くなるという結果となった。このため、GATHER展開の修正は、PNCCSに組み込まず、ICVMGTの簡素化のみ組み込むこととした。

表3.1 AFDMシステムルーチン

ルーチン名	機 能	変 換 方 法
EXITA	プログラムの実行を終了させる。	その旨のメッセージを出力してリターン
GETJTL	実行中ジョブの残りCPU時間を得る。	同機能のアセンブラルーチンLEFTを呼び出す。
SECOND	ジョブが使用したCPU時間を得る。	FACOM FORTRANのサービスサブルーチンCLOCKで代用
MSGFT	コモン変数にセットされているメッセージを出力する。	当ルーチン通過のメッセージを出力
CLOCK	ジョブが使用したCPU時間を μ sec単位で得る。	FACOM FORTRANのサービスルーチンCLOCKで代用。
UNIT	該当機番のファイルが終了したかどうかの判定を行うCRAY組込み関数	READ文のパラメータ-ENDで制御
DATE	実行時の日付を得る。	変換せず (FACOM FORTRANに同名同機能のルーチン有)

表3.2 AFDM図形処理ルーチン (除PUL)

ルーチン名	機 能	変 換 方 法
GDONE	CGS終了処理	DISPLA終了処理に変換
GFNAME	グラフィックファイル名のセット	削 除
GPLBEG	図形処理装置の初期処理の開始	削 除
GPLEND	図形処理装置の初期処理の終了	削 除
GOLAY	CGSで使用するコモンブロックの初期処理	削 除
GSTART	CGS初期処理	DISPLA初期処理に変換
GTEXT	文字列の出力	DISPLA MESSAGEルーチンに変換

表3.3 AFDMのコスト分布とベクトル化率 (MINPA-LWEPケース)

モジュール名	コスト分布	ベクトル化率
RSLOPE	19.8 %	99.5 %
ASLOPE	12.7	99.3
VITERP	8.5	99.8
AFDMS3	7.7	99.3
EOSPD	6.1	99.9
VITER	5.8	99.6
CVR	3.8	99.5
CVZ	3.8	99.6
EOST	3.1	93.3
AFDMS2	3.0	99.2
プログラム全体	100.0	70.4

表3.4 AFDMのコスト分布とベクトル化率 (MINPS-LWEPケース)

モジュール名	コスト分布	ベクトル化率
SRCH3D	64.1 %	4.6 %
WLCH	13.1	0.0
PICCON	6.4	0.0
URANUS	6.2	99.5
LAGRAN	4.2	99.3
EOSPEQ	1.4	86.4
BIRATF	0.6	99.7
EOST	0.6	91.3
RSLOPE	0.6	99.5
EOSPD	0.3	99.0
プログラム全体	100.0	1.7

表3.5 SRCH3Dのベルト化テスト結果 (MINPS-LWEPケース)

	修正ケース		
	未修正	ICVMGTの簡素化	+GATHER展開
SRCH3D CPU (1回目)	2470 (μ sec)	363 (μ sec)	330 (μ sec)
SRCH3D CPU (2回目)	607 (μ sec)	238 (μ sec)	198 (μ sec)
1サイクル終了CPU	—	19" 86	19" 91
計算終了サイクル数	6	23	23
計算終了CPU	1' 05" 59	1' 07" 17	1' 07" 39
CPU/サイクル	10" 93	2" 92	2" 93

AFPROG	*IF -DEF,CRAY,1	D85A	16
1	PROGRAM AFPROG	D85A	17
AFPROG	*IF DEF,CRAY,1	D85A	18
AFPROG	PROGRAM AFPROG(TAPES,TAPE6,TAPE7,TAPE8)	AFPROG	5

図 3. 1 A F D M プログラム変換例 (1)

XEOSPRH	*IF -DEF,CRAY,1	D85A	10
XEOSPRH	COMMON /XEOSPR/HRTAB(MHDEH),HATER(HREGS)	D85A	11
XEOSPRH	*IF DEF,CRAY,1	D85A	12
XEOSPRH	COMMON/XEOSPRH/HRTAD(MHDEH),HATER(HREGS)	P158	13

図 3. 2 A F D M プログラム変換例 (2)

AFDHS2	*IF -DEF,CRAY,1	D85A	77
184	SPC(IJ) = .FALSE.	D85A	78
AFDHS2	*IF DEF,CRAY,1	D85A	79
AFDHS2	SPC(IJ) = .F.	AFDHS2	222

図 3. 3 A F D M プログラム変換例 (3)

PLTCEL	*IF -DEF,CRAY,2	D85A	108
43	LABX(1) = 'Z'	D85A	109
44	LABX(2) = 'R'	D85A	110
PLTCEL	*IF DEF,CRAY,2	D85A	111
PLTCEL	LABX(1) = 'Z'	PLTCEL	51
PLTCEL	LABX(2) = 'R'	PLTCEL	52

図 3. 4 A F D M プログラム変換例 (4)

25	IF(LREC.GE.0) THEN	RDFIL	29
RDFIL	*IF -DEF,CRAY,3	D85A	178
26	READ(FN,IOSTAT=IOS)(AR(I),I=1,N)	D85A	179
27	IF<IOS.LT.0> IRPT = -IRPT	D85A	180
28	ELSE	D85A	181
RDFIL	*IF DEF,CRAY,3	D85A	182
RDFIL	READ(FN,END=10)(AR(I),I=1,N)	RDFIL	30
RDFIL	ELSE	RDFIL	31
RDFIL	10 CONTINUE	RDFIL	32
29	IRPT=-IRPT	RDFIL	33
30	ENDIF	RDFIL	34

図 3.5 AFD Mプログラム変換例 (5)

INITIL	*IF -DEF,CRAY,1	D85A	33
12	READ(INP,'(20A4)',END=20)CARD	D85A	34
INITIL	*IF DEF,CRAY,1	D85A	35
INITIL	READ(INP,'(10A8)',END=20) CARD	INITIL	16

図 3.6 AFD Mプログラム変換例 (6)

AFDFLO	*IF -DEF,CRAY,1	D85A	68
22	CALL CLOCK(T1,0,1)	D85A	69
AFDFLO	*IF DEF,CRAY,1	D85A	70
AFDFLO	CALL SECONH(T1)	AFDFLO	60

図 3.7 AFD Mプログラム変換例 (7)

SIC	*DK SIC	SIC	1
SIC	*IF DEF,PSQL2	SIC	2
SIC	SUBROUTINE SIC(CL,CR,CB,CT,DL,DR,DB,DT,DO,V,IB,IBJB,IBPJB,IND	SIC	3
SIC	*CA INPDBL	SIC	4
SIC	1 ,IBM1)	SIC	5
SIC	*DK SIC	SIC	1
SIC	*IF DEF,PSQL2	SIC	2
SIC	SUBROUTINE SIC(CL,CR,CB,CT,DL,DR,DB,DT,DO,V,IB,IBJB,IBPJB,IND	SIC	3
SIC	1 ,IBM1)	D85A	192
SIC	*CA INPDBL	D85A	193

図 3.8 AFD Mプログラム変換例 (8)

DECK SRCH3D

SRCH3D	*DK SRCH3D	SRCH3D	1
SRCH3D	*IF DEF,SESAME	SRCH3D	2
SRCH3D	*IF -DEF,CTSS	SRCH3D	3
SRCH3D	SUBROUTINE SRCH3D (H,Y,IY,N,X,IX,J,D,F,E,IE)	SRCH3D	4
SRCH3D	C	SRCH3D	59
SRCH3D	*CA INPDBL	SRCH3D	60
SRCH3D	DIMENSION Y(IY,N),J(N),X(IX,N),D(3,H),F(H),E(H,3),IE(H,3)	SRCH3D	61
SRCH3D	DO 20 I=1,H	SRCH3D	62
SRCH3D	*IF -DEF,CRAY,3	DBSA	173
SRCH3D	J(I)=ICVMGT(1,ICVMGT(H,0,Y(1,I)).GE.X(1,N)),Y(1,I).LT.X(1,1))	DBSA	174
SRCH3D	DO 10 L=1,N-1	DBSA	175
SRCH3D	J(I)=ICVMGT(L,J(I),(X(1,L).LE.Y(1,I)).AND.(Y(1,I).LT.X(1,L+1)))	DBSA	176
SRCH3D	*IF DEF,CRAY,3	DBSA	177
SRCH3D	J(I)=CVHGT(1,CVHGT(H,0,Y(1,I)).GE.X(1,N)),Y(1,I).LT.X(1,1))	SRCH3D	63
SRCH3D	DO 10 L=1,N-1	SRCH3D	64
SRCH3D	J(I)=CVHGT(L,J(I),(X(1,L).LE.Y(1,I)).AND.(Y(1,I).LT.X(1,L+1)))	P468	3
SRCH3D	10 CONTINUE	SRCH3D	66
SRCH3D	20 CONTINUE	SRCH3D	67
SRCH3D	DO 30 I=1,H	SRCH3D	68
SRCH3D	IE(I,1)=J(I)-1	SRCH3D	69
SRCH3D	IE(I,2)=J(I)+1	SRCH3D	70
SRCH3D	IE(I,3)=J(I)+2	SRCH3D	71
SRCH3D	30 CONTINUE	SRCH3D	72
SRCH3D	CALL GATHER(H,E(1,1),X,IE(1,1))	SRCH3D	73
SRCH3D	CALL GATHER(H,E(1,2),X,J)	SRCH3D	74
SRCH3D	CALL GATHER(H,E(1,3),X,IE(1,2))	SRCH3D	75
SRCH3D	CALL GATHER(H,F,X,IE(1,3))	SRCH3D	76
SRCH3D	DO 40 I=1,H	SRCH3D	77
SRCH3D	D(1,I)=E(1,2)-E(I,1)	SRCH3D	78
SRCH3D	D(2,I)=E(1,3)-E(I,2)	SRCH3D	79
SRCH3D	D(3,I)=F(I)-E(1,3)	SRCH3D	80
SRCH3D	F(I)=(Y(1,I)-E(1,2))/D(2,I)	SRCH3D	81
SRCH3D	40 CONTINUE	SRCH3D	82
SRCH3D	RETURN	SRCH3D	83
SRCH3D	END	SRCH3D	84
SRCH3D	*EI	SRCH3D	85

図3.9 SRCH3Dソースリスト

```

000640      DO 10 L=1,N-1
000650 C      J(I)=ICVMGT(L,J(I),(X(1,L).LE.Y(1,I)).AND.(Y(1,I).LT.X(1,L+1)))
000651          IF((X(1,L).LE.Y(1,I)).AND.(Y(1,I).LT.X(1,L+1))) J(I) = L
000660      10 CONTINUE
    
```

図3.10 ICVMGTの簡素化 (in SRCH3D)

```

000730 C      CALL GATHER(M,E(1,1),X,IE(1,1))
000740 C      CALL GATHER(M,E(1,2),X,J)
000750 C      CALL GATHER(M,E(1,3),X,IE(1,2))
000760 C      CALL GATHER(M,F,X,IE(1,3))
000761      DO 35 I=1,M
000762          E(I,1) = X(IE(I,1),1)
000763          E(I,2) = X(J(I),1)
000764          E(I,3) = X(IE(I,2),1)
000765          F(I)   = X(IE(I,3),1)
000766      35 CONTINUE
    
```

図3.11 GATHERルーチンの展開 (in SRCH3D)

4. テスト計算結果

4.1 ロードモジュール作成

前章で述べた様な変換作業の結果、PNC FACOM機でAFDMが動作可能となった。本節では、その変換AFDMのテスト作業結果について述べる前に、AFDM最終版のロードモジュール作成法について若干述べておくことにする。

4.1.1 前処理コード

PNCでは前処理コードとしてHISTORIANを利用し、AFDMを管理している。AFDM最終版(Y8903版)のFORTRANソースを作成するためには、以下のデータセットが必要となる。

- (i) AFDM Y8901版 HISTORIANライブラリ
- (ii) Y8903版作成用LANLコレクションセット(修正情報リスト)
- (iii) HISTORIAN制御ファイル
- (iv) PNC用AFDMアッドファイル(追加モジュール)
- (v) PNC用AFDMコンクションセット

このうち(iii)~(v)について以下の各節で説明を加える。

4.1.2 HISTORIANオプション

上記HISTORIAN制御ファイルで、AFDMに対するHISTORIANオプションを指定する。HISTORIANの“*IF”指示子はAFDMコードに多くのオプションを指定可能にしている。現在のAFDMコードのHISTORIANオプションは以下のとおり。

- (1) CRAY ON CRAY 1 or XMP機用
OFF FACOM機用
- (2) CTSS ON CTSSシステム用
- (3) DBL ON 倍精度
OFF 単精度
- (4) FOURB ON 1語4バイト, 1バイト8ビット マシン用
OFF 1語8バイト, 1バイト8ビット マシン用
- (5) CGS ON LANL CGS (Common Graphic System)用
- (6) NCAR ON NCAR (National Center for Atmospheric Research)
SCD図形処理システム用
- (7) DISP ON DISSPLA用
- (8) SESAME ON SESAME EOS
OFF 簡易解析的EOS
- (9) KQQM1 ON 簡易交換関数モデル

- (0) KQQM2 ON Wilhelm/Berthier交換関数モデル
 - (1) DONOR ON ドナーセル差分法
OFF 高次差分法
 - (2) IVIS ON 仮想質量オプション
 - (3) PSOL1 ON 圧力方程式の解法モジュールMATSOL
 - (4) PSOL2 ON 圧力方程式の解法バンド行列直接解法
 - (5) NOPHASE ON 相変化計算を行わない (KQQMIオプションON
でなければならない。)
OFF 相変化計算を行う。
 - (6) ARCV1 ON 境界面積対流モデルはWilhelm モデル1
(KQQM2 ON)
 - (7) ARCV2 ON 境界面積対流モデルはWilhelm モデル2
(KQQM2 ON)
 - (8) F77VP ON FACOM VP機用
- なお、このうち(8)はPNCで追加された。

4.1.3 PNC用AFDMアッドファイル

PNC用AFDMアッドファイルはデータセット

'T201#. AFDM. PNCAF. FORT'

に格納されている。この内容を図4.1に示す。これは3.2.4節で示したシステムルーチンGETJTLをPNC用に交換したものである。

4.1.4 PNC用AFDMコレクションセット

AFDM最終版のPNC用コレクションセットはデータセット

'T201#. AFDM. Y8903. PNCCS. FORT'

に格納されている。この内容を図4.2に示す。このコレクションセットは3章で述べた様に、PNCでの変換作業が取入れられた後の版に対するものであるためCRAY-FACOM間の変換修正というよりもエラー修正の性格が強い。このコレクションセットについて以下に説明を加える。なお細項(1)~(3)はそれぞれ図4.2の項番に対応している。

(1) ファイルの静的割当てへの変更

オリジナルのAFDMでは、使用ファイルの割当てを動的に行っているため使用するデータセット名が固定されていたが、使用するデータセット名を自由にえられる様に、PNCでは静的割当てに変更し、JCL上で使用データセット名を指定できる様にした。

(2) 整合配列の大きさの指定

実行時異常終了を避けるため、整合配列の大きさをサブルーチン呼出し元で

指定した。

- (3) 順次入出力SESAMEファイルに関する修正
上記ファイル入力で起こっていたエラーに対する修正
- (4) 変数MULTIに関する修正
倍精度実数（DBLオプション使用時）と領域共有（EQUIVALENC E）されている整数型データ取扱い部のエラー修正
- (5) 4バイトマシン用の修正
- (6) 未定義変数に関する修正
未定義の文字変数（の配列）に空欄が入るものとしてオリジナルのAFDM は取扱っているが、FACOMではそうとは限らないため、これに対処する様に修正した。
- (7) システムルーチンの変換
CRAYルーチンSECONDをFACOM FORTRAN77サービスルーチンCLOCKに変換した。
- (8) DISPLAYへの対応
同名のルーチンがあるCOMMON名UNITSをUNITCに変更。
DISPLAY初期処理のPNC用への変換。
- (9) 指数オーバーフローエラー回避のための変更
実数の値域が小さいことにより発生するオーバーフローを回避するための修正。行列計算ではSSLIIのルーチンDLSEXを使用することとした。
- (10) 計算終了時間の設定の変更
マシン依存である計算終了時間の設定の変更
- (11) FACOM VP用の修正
 - VP用最適化制御行の追加
 - PIオプションに対応できる様にするための倍精度宣言文の追加
 - 計算結果に影響を及ぼさない様にするための、計算順序の変更を抑止するオプションADV（NOEVL）のサブルーチンEOSPDへの追加（翻訳指示文PROCESS行で対処）
 - 効率改善のためのSRCH3Dへの修正（3.6.4節参照）
- (12) Y8811版へのエラー修正
- (13) Y8901版へのエラー修正

4.1.5 ロードモジュール作成法

以上の各節で述べたファイルを使用し、HISTORIANを実行してAFDMのFORTRANソースを作成、翻訳、結合してロードモジュールを作成すればよい。FACOM VP用のリースプログラムを作るためには、HISTORIANオプションのF77VPを立てる必要がある。また、結合の際必要なライ

ブラリは以下

- ・ SYS1. FORTLIB FORTRANライブラリ
- ・ T201#. SIMMER. ASMLIB アセンブラライブラリ
- ・ T201#. PULIB PULライブラリ
- ・ SYS9. DISLIB DISPLAライブラリ
- ・ SYS9. INTLIB DISPLAインターフェ
ースライブラリ
- ・ SYS1. SSL2 科学用サブルーチンライブラ
リSSLII

以上の処理に対応したPNC用のAFDMロードモジュール作成用JCL例を図4.3に、また、PNC VP機用のAFDMロードモジュール作成用JCL例を図4.4に示す。これはデータセット

```
' T201#. AFDM. CNTL (HCL) '
```

```
' T201#. AFDM. CNTL (HVPCL) '
```

にそれぞれ格納されている。また、作成されたAFDMロードモジュール、VP用AFDMロードモジュールは、それぞれデータセット

```
' T201#. AFDM. LOAD (M21A1) '
```

```
' T201#. AFDM. LOAD (M21A1V) '
```

に格納されている。なお、このVP用モジュールの実行の際、実行時オプション(UNDERFL=0)が必要となるので注意されたい(次節参照)。

4.2 テスト計算

4.2.1 テスト計算結果

本節で示すテスト計算はY8811版9ケース、Y8901版3ケース、Y8903版1ケースの計13例である。この13例について表4.1に示しておく。以上13例についての蒸気体積率分布図を、定常計算、過渡計算結果それぞれについてCRAY-FACOM間の比較のため、図4.5~4.17に示しておく。なお、このテスト計算のための入力データは、Y8811版については

```
' T201#. AFDM. Y8811. DATA '
```

に、Y8901版、Y8903版については、

```
' T201#. AFDM. DATA '
```

に格納してある。このAFDM実行用JCL例とVP用JCL例をそれぞれ図4.18、4.19に示す。このJCLはそれぞれ

```
' T201#. AFDM. CNTL (AFDM) '
```

```
' T201#. AFDM. CNTL (AFDMV) '
```

に格納されている。このうち、VP用の実行のときは、実行時オプションのUN

DERFL=0を指定しなければならない。このオプションは、指数アンダーフローが発生した場合、指数部をゼロとするオプションである（スカラー用ロードモジュール実行時の場合と同じ）。

4.2.2 結果比較

図4.5～4.17より、オリジナルCRAY版と変換FACOM版を比較してみると、定常計算、過渡計算共に結果に殆ど違いはみられない。

LANLよりCOMで送られてきたリスト出力と比較しても、過渡計算1回目の各メッシュセルデータリスト（図4.5～4.17で示したタイミングと同じ）で、全ての変数が有効数字5桁か、それ以上の精度で一致している。ところがKQQM2オプションを立てたケースの場合はこれ程の一致は得られない。それについての検討結果を以下に示す。

3.2.5節で述べた様にCRAY-FACOM間には、計算結果に影響を与える以下の要因がある。

(1) 計算機ハードウェアの違い

浮動小数点内部表現形式は、CRAYの場合、指数部15ビット、仮数部48ビットであるのに対し、FACOMにおける倍精度表現の場合、指数部7ビット、仮数部56ビットである。このため、絶対値の範囲はCRAYの場合約 10^{-2466} ～ 10^{2466} 、FACOMの場合 10^{-78} ～ 10^{74} となる。一方、有効桁数はCRAYが約14桁に対しFACOMが約16桁である。従って絶対値がCRAYでは取扱える範囲ではあっても、FACOMでは指数オーバーフロー・アンダーフローを起こす可能性がある（これを回避するためにソースプログラムの修正を行ったがこれが計算結果に影響する可能性はある。）また、有効桁数の違いも、計算精度に影響を及ぼす要因となる。

(2) FORTRANライブラリの違い

CRAY-FACOM間では利用するFORTRANライブラリが異なり、従ってこのライブラリの持つ組込み関数の精度が異なる。この数値精度の相違が計算結果に影響を及ぼす可能性がある。

(3) FORTRANコンパイラの違い

FORTRANコンパイラの相違のため、CRAY-FACOM間では実行文あるいは式の評価の計算順序が異なる。そこで計算順序の違いが結果に影響を及ぼすかどうかの検討を以下のように行った。図4.7等から明らかのようにKQQM2オプションを立てたケースでは時間ステップのタイミングがCRAYと異なる。そこでY8901版M21S1ケースについて、FORTRANコンパイラオプションの最適化制御をパラメータとしてOPT(0)、OPT(3)のテストを行い、また、VPオプションについてもテストを行ってCRAYの結果と比較した。定常計算では違いが現われなかったので、1サイクル

以降の過渡計算について調べた。図4.20に、C R A Y, O P T (0), O P T (3), V P ケースについての11サイクルまでのリスト出力を掲げる。これを見ると、C R A Y - F A C O M 間のみならず、O P T (0) - O P T (3) の間でさえ結果に相違がみられる。この最適化オプションの相違によって計算順序に違いが発生するので、行列計算法 (P S O L V R) の選択が異なったり、微量の判定を行う事による圧力繰返し計算回数を (I T E R A , I C O P E Q) の違いとなって現われる。F A C O M ← C R A Y 間では、この計算順序の他、数値精度、組込み関数の精度に相違があり、このため上記のような違いが増幅して発生する可能性がある。

ところで、A F D M はタイムステップの制御法として6種あるが、このうち圧力繰返し計算に係わるものとして、次のタイムステップ Δt^{n+1} は、

$$\Delta t^{n+1} = \frac{O P T P I T}{I T E R A + 1} \Delta t^n$$

を越えないものとする、という制限がある。ここで O P T P I T は入力データである。従って圧力繰返し計算回数異なる場合 (現に異なった)、タイムステップの値が異なる可能性が示された (本ケースでは11サイクル以降)。

以上の検討の結果、A F D M は非常に微妙かつ複雑かつ大量の計算を行っているために、C R A Y - F A C O M 間で微妙に結果が異なる可能性があることが確認された (なお、この相違の定量的検討は、C R A Y の計算結果の情報が少ないため、非常に困難である。) しかし、一方、変換作業の“とりこぼし”の可能性が全くないとはいえず、今後の各種のサンプル問題の実施等で使用経験を積んで確認してゆく必要がある。

4.2.3 実行環境

Y 8 9 0 3 版 M 2 1 A 1 ケース (メッシュサイズ 3×12) の実行時に要する諸資源を表4.2に示す。この表より、本ケースに要するCPU時間は、ベクトル版はスカラー版に比べて1.69倍、短縮されていることがわかる。一方、F A C O M - C R A Y 間では要するCPU時間が大きく異なる。これはハードウェア、F O R T R A N コンパイラの差、F A C O M 用ベクトル化効率化作業の不十分な点、他、A F D M コードが使用している図形処理ルーチンが異なる (C R A Y は C G S システム、F A C O M は D I S S P L A システム) 事に大きな原因が求められる。従って、F A C O M 機、C R A Y 機の実行性能を比較する場合は、図形処理部を除いて比較する事が望ましい。

なお、使用機種は C R A Y X - M P 4 1 6 (4CPU, 16Mword)、F A C O M V P - 100 である。

表 4.1 AFDMテスト計算ケース

AFDM版名	ケース名	コードオプション (主なもの)	入力データ
Y8811	M1NPA1	IVIS, PSOL2, KQQM1, NOPHASE	LWEP
	M1PA1	IVIS, PSOL2, KQQM1	LWEP
	M21A1	IVIS, PSOL2, KQQM2, ARCV1	LWEP
	M1NPS1	IVIS, PSOL2, KQQM1, NOPHASE, SESAME	LWEP
	M1PS1	IVIS, PSOL2, KQQM1, SESAME	LWEP
	M21S1	IVIS, PSOL2, KQQM2, ARCV1, SESAME	LWEP
	K1NPS1	IVIS, PSOL2, KQQM1, NOPHASE, SESAME (KfK solid BOS コレクション追加)	LWEP
	K1PS1	IVIS, PSOL2, KQQM1, SESAME (KfK solid BOS コレクション追加)	LWEP
	K21S1	IVIS, PSOL2, KQQM2, ARCV1, SESAME (KfK solid BOS コレクション追加)	LWEP
Y8901	M21S1	IVIS, PSOL2, KQQM2, ARCV1, SESAME	LWEP
	M1PS2	IVIS, PSOL2, KQQM1, SESAME	LPOOL
	M21A3	IVIS, PSOL2, KQQM2, ARCV1	BPOOL
Y8903	M21A1	IVIS, PSOL2, KQQM2, ARCV1	LWEP

表 4.2 AFDM実行環境 (Y8903版 M21A1ケース)

使用環境	CPU (sec)	使用領域 (KB)
LANL	15.2	—
PNC (スカラー版)	193.0	1992
PNC (VP版)	114.1	2968

(注) 使用図形処理システムは、LANLはCGSシステム、PNCはDISPLAシステム

```
*/
*/          ***** AFDM PNC ADDFILE *****
*/          BY KOD 88.12.02
*/
*DECK GETJTL
SUBROUTINE GETJTL(TL)
C   AFDM - GETJTL
C   GET JOB(CPU) TIME LIMIT
C   TLEFT - RESIDUAL CPU TIME (1/100 SEC)
*IF DEF, DBL, 1
REAL*8 TL
TL = TLEFT(DUM)/100.
RETURN
END
```

☒ 4.1 PNC用AFDMアッドファイル

```

*IDENT PNCC1
*/
*/ ***** PNC CORRECTION SET FOR AFDM *****
*/          BY KOD : 88.05.20
*/          MOD BY KOD : 88.11.25 (FOR Y8811 VERSION)
*/          MOD BY KOD : 88.02.02 (FOR Y8901 VERSION)
*/          MOD BY KOD : 88.02.03 (FOR Y8901 VERSION PART2)
*/
-----
*/
*/ 1. CORRECTIONS FOR NON-CRAY (IBM) MACHINES
*/
*/ (1) CHANGED TO BE ABLE TO USE ALLOCATED FILES (IN PNC)
*/
*/ I AFPROG.39
*/ IF DEF,CTSS,8
*/
*/ I SESINI.60
*/ IF DEF,CTSS,23
*/ I SESINI.84
*/ IF DEF,CTSS,16
-----
*/
*/ (2) SET DIMENSION OF ADJUSTABLE ARGUMENT IO IN SEPER
*/
*/ I INITIL.19
*/   NF = 40
*/ B RDINP.188
*/   NF = 40
*/ B RDINP.203
*/   NF = 40
*/ I MSEP.28
*/   NAR = 10
-----
*/
*/ (3) CORRECT FOR READING SEQUENTIAL SESAME FILE
*/
*/ I SKIPF.10
*/   CHARACTER*4 TEST,FH/'VFH'/
*/ D SKIPF.14
*/   READ(LUN,END=30) TEST
*/   IF(TEST.EQ.FH) GO TO 20
*/ I SKIPF.16
*/   30 CONTINUE
-----
*/
*/ (4) CORRECT MULTI IN EOST & VITER & VITERP
*/
*/ D EOST.118
*/   IER(IJ*MULTI,11)=IJ
*/ D EOST.150,152
*/   TS(IER(IJ*MULTI,11),1) = ER(IJ,16)
*/   TS(IER(IJ*MULTI,11),2) = ER(IJ,17)
*/   T(IER(IJ*MULTI,11),5) = ER(IJ,12)
*/
*/ D VITER.125
*/   IER(IJ*MULTI,1)=ICVMGT(1,2,ALPQ(IJ,2)+ALPQ(IJ,3).GE.ALPO(IJ,4))
*/ D VITER.167
*/   IER(IJ*MULTI,20)=IJ
*/ D VITER.272
*/   TEMP1=CVHGT(ER(IJ,11),ER(IJ,12),1.GE.IER(IJ*MULTI,1))
*/ D VITER.274
*/   TEMP2=CVHGT(ER(IJ,25),ER(IJ,26),1.GE.IER(IJ*MULTI,1))
*/ D VITER.306,314
*/   VEL(IER(IJ*MULTI,20),1)=ER(IJ,11)
*/   VEL(IER(IJ*MULTI,20),2)=ER(IJ,12)
*/   VEL(IER(IJ*MULTI,20),3)=ER(IJ,13)
*/ IF DEF,IVIS
*/   VH(IER(IJ*MULTI,20),1)=+ER(IJ,23)*(ER(IJ,13)
*/     * -VELH(IER(IJ*MULTI,20),3)
*/     * -ER(IJ,21)*(ER(IJ,11)-VELH(IER(IJ*MULTI,20),1))
*/     * -ER(IJ,22)*(ER(IJ,12)-VELH(IER(IJ*MULTI,20),2)))
*/   VH(IER(IJ*MULTI,20),2)=ER(IJ,22)
*/
*/ D VITERP.126
*/   IER(IJ*MULTI,1)=ICVMGT(1,2,ALPQ(IJ,2)+ALPQ(IJ,3).GE.ALPO(IJ,4))
*/ D VITERP.174
*/   IER(IJ*MULTI,20)=IJ
*/ D VITERP.333
*/   TEMP1=CVHGT(ER(IJ,13)
*/     * ,CVHGT(ER(IJ,11),ER(IJ,12),1.GE.IER(IJ*MULTI,1))
*/ D VITERP.336
*/   TEMP2=CVHGT(ER19,CVHGT(ER17,ER18,1.GE.IER(IJ*MULTI,1))
*/ D VITERP.368,376

```

図 4.2 PNC用AFDMコレクションセット

```

VEL(IER(IJ*MULTI,20),1)=ER(IJ,11)
VEL(IER(IJ*MULTI,20),2)=ER(IJ,12)
VTS = C1P4*ER(IJ,26)
VEL(IER(IJ*MULTI,20),3)=CVMGT(ER(IJ,13)
*      ,VTS*SIGN(CONE,ER(IJ,13)),ABS(ER(IJ,13)).LT.VTS)
DVDP(IER(IJ*MULTI,20),1)=ER(IJ,37)
DVDP(IER(IJ*MULTI,20),2)=ER(IJ,38)
DVDP(IER(IJ*MULTI,20),3)=ER(IJ,39)
-----
*/
*/ (5) RESET NO. OF WORDS FOR 4 BYTES MACHINE.
*/
*/ I LENG.8
*/ IF DEF,FOURB,1
*/     LENG = LENG/4
*/
*/ D WRDHP.30
*/     IPRC = 1
*/
*/ D RDDHP.32
*/     IPRC = 1
-----
*/
*/ (6) NOT TO PLOT TOPOLOGY 0
*/     BECAUSE ALP(0) NE ' ',SO PLOT '<' IN FACOM
*/
*/ D TOPLOT.111
*/     IF(TOP(I,J).NE.0) CALL WLCH(JC,JC,1,ALP(TOP(I,J)),CHARS)
-----
*/
*/ IDENT PNCL1
*/
*/ PNCL1: LOCAL CORRECTIONS AT PNC O-ARAI
*/     THIS PART MAY OR MAY NOT BE INCLUDED IN
*/     LOS ALAMOS LIBRARY.
*/
*/ 1. CHANGE SYSTEM ROUTINE (SECOND TO CLOCK).
*/
*/ I AFD FLO.53
*/ IF -DEF,CRAY,1
*/     CALL CLOCK(T1,0,1)
*/ IF DEF,CRAY,1
*/ I AFD FLO.123
*/ IF -DEF,CRAY,1
*/     CALL CLOCK(T2,0,1)
*/ IF DEF,CRAY,1
*/ I AFD FLO.147
*/ IF -DEF,CRAY,1
*/     CALL CLOCK(TELASP,0,1)
*/ IF DEF,CRAY,1
-----
*/
*/ 2. CHANGE DISSPLA ROUTINES FOR PNC(NLP)
*/     & CHANGE LABEL UNITS TO UNITC
*/
*/ I UNITS.1
*/ IF DEF,DISP,1
*/     COMMON/UNITC/INP,OUT,PPFN,DUMP
*/ IF -DEF,DISP,1
*/ D AFPROG.63,66
*/     CALL NLP
*/     CALL UNITS('IN')
*/     CALL PAGE(9.4,8.5)
*/     CALL PHYSOR(0.5,0.5)
*/     CALL TRIPLX
-----
*/
*/ 3. TEMPORARY CORRECTION AT PNC
*/
*/ (1) CORRECT XEOSPHC & XEOSS1 & XEOSPD & XEOSTP & XEOSD2 & XEOSVD1
*/     TO ESCAPE FROM OVERFLOW ERRORS
*/
*/ I XEOSPHC.197
*/     IF(A.GT.1E35 .AND. B.GT.1E35 .OR.
*/       & A.GT.1E35 .AND. C.GT.1E35 .OR.
*/       & B.GT.1E35 .AND. C.GT.1E35) THEN
*/         RD = ZERO
*/     ELSE
*/ I XEOSPHC.198
*/     ENDF
*/ I XEOSS1.207
*/     IF(A.GT.1E35 .AND. B.GT.1E35 .OR.
*/       & A.GT.1E35 .AND. C.GT.1E35 .OR.
*/       & B.GT.1E35 .AND. C.GT.1E35) THEN
*/         RD = ZERO

```

図 4.2 (続 き)

```

ELSE
*I XE0SS1.208
ENDIF
*I XE0SPD.450
IF(A.GT.1E35 .AND. B.GT.1E35 .OR.
& A.GT.1E35 .AND. C.GT.1E35 .OR.
& B.GT.1E35 .AND. C.GT.1E35) THEN
RD = ZERO
ELSE
*I XE0SPD.451
ENDIF
*I XE0SPD.539
IF(A.GT.1E35 .AND. B.GT.1E35 .OR.
& A.GT.1E35 .AND. C.GT.1E35 .OR.
& B.GT.1E35 .AND. C.GT.1E35) THEN
ER(N,81) = ZERO
ELSE
*I XE0SPD.540
ENDIF
*I XE0STP.401
IF(A.GT.1E35 .AND. B.GT.1E35 .OR.
& A.GT.1E35 .AND. C.GT.1E35 .OR.
& B.GT.1E35 .AND. C.GT.1E35) THEN
RD = ZERO
ELSE
*I XE0STP.402
ENDIF
*I XE0STP.491
IF(A.GT.1E35 .AND. B.GT.1E35 .OR.
& A.GT.1E35 .AND. C.GT.1E35 .OR.
& B.GT.1E35 .AND. C.GT.1E35) THEN
ER(N,81) = ZERO
ELSE
*I XE0STP.492
ENDIF
*I XE0SD2.221
IF(A.GT.1E35 .AND. B.GT.1E35 .OR.
& A.GT.1E35 .AND. C.GT.1E35 .OR.
& B.GT.1E35 .AND. C.GT.1E35) THEN
RD = ZERO
ELSE
*I XE0SD2.222
ENDIF
*I XE0SVD1.380
IF(A.GT.1E35 .AND. B.GT.1E35 .OR.
& A.GT.1E35 .AND. C.GT.1E35 .OR.
& B.GT.1E35 .AND. C.GT.1E35) THEN
RD = ZERO
ELSE
*I XE0SVD1.381
ENDIF
*I XE0SVD1.430
IF(A.GT.1E35 .AND. B.GT.1E35 .OR.
& A.GT.1E35 .AND. C.GT.1E35 .OR.
& B.GT.1E35 .AND. C.GT.1E35) THEN
ER(N,81) = ZERO
ELSE
*I XE0SVD1.431
ENDIF
*/
*/
*/ (2) CHANGE MATRIX SOLVER TO DLSIX FOR ESCAPING OVERFLOW
*/ IN XE0ST & XE0SPEQ & S4X4S (FOR Y8808 VERSION)
*/ BY KOD '88.09.01
*/ MOD BY KOD '88.12.22 (FOR Y8811 VERSION)
*/
*I XE0ST.103
*IF -DEF,CRAY,1
DIMENSION AJ(6),B(3),VW(6),IP(3),IVW(3)
*I XE0ST.301
*IF -DEF,CRAY,16
AJ(1) = AJ11
AJ(2) = AJ12
AJ(3) = AJ22
AJ(4) = AJ13
AJ(5) = AJ23
AJ(6) = AJ33
C
B(1) = ER(1,60)
B(2) = ER(1,61)

```

図 4.2 (続 き)

```

        B(3) = ER(1,62)
C
        CALL DLSIX(AJ,3,B,0.0,1,VW,IP,IVW,ICON)
C
        ER(1,57) = B(1)
        ER(1,58) = B(2)
        ER(1,59) = B(3)
*IF DEF,CRAY,8
*/
*I XEOSPEQ.130
*IF -DEF,CRAY,1
        DIMENSION AJ(6),B(3),VW(6),IP(3),IVW(3)
*I XEOSPEQ.345
*IF -DEF,CRAY,16
        AJ(1) = AJ11
        AJ(2) = AJ12
        AJ(3) = AJ22
        AJ(4) = AJ13
        AJ(5) = AJ23
        AJ(6) = AJ33
C
        B(1) = ER(1,13)
        B(2) = ER(1,14)
        B(3) = ER(1,15)
C
        CALL DLSIX(AJ,3,B,0.0,1,VW,IP,IVW,ICON)
C
        ER(1,2) = B(1)
        ER(1,3) = B(2)
        ER(1,4) = B(3)
*IF DEF,CRAY,8
*/
*I S4X4S.17
*IF -DEF,CRAY,1
        DIMENSION A(10),VW(8),IP(4),IVW(4)
*I S4X4S.28
*IF -DEF,CRAY
        A(1) = R1(1,N)
        A(2) = R1(2,N)
        A(3) = R2(1,N)
        A(4) = R1(3,N)
        A(5) = R2(2,N)
        A(6) = R3(1,N)
        A(7) = R1(4,N)
        -A(8) = R2(3,N)
        A(9) = R3(2,N)
        A(10) = R4(N)
C
        X(1,N) = B(1,N)
        X(2,N) = B(2,N)
        X(3,N) = B(3,N)
        X(4,N) = B(4,N)
C
        CALL DLSIX(A,4,X(1,N),0.0,1,VW,IP,IVW,ICON)
C
*ENDIF
*IF DEF,CRAY
*I S4X4S.45
*ENDIF
*/
*/ (3) CHANGE CLOSE TIME FOR FACOM M-380 (TEMPORARY)
*/ THIS CORRECTION SHOULD BE MODIFIED FOR VP OR OTHER MACHINES
*/
*/
*/D INIPAR.209
        IF(FILM) TCLOSE = TCLOSE - MAX(CP1*FLOAT(IJOB2),80.)
*/
*/ 4. CORRECT FOR VP
*/ (1) ADD VOCL STATEMENT
*/
*I INIKQ0.63
*VOCL LOOP,NOVREC
*I INIKQ0.107
*VOCL LOOP,NOVREC
*/
*/ (2) ADD IMPDBL FOR PI OPTION
*/
*I ICVMGT.9
*CALL IMPDBL
*/
*/ (3) ADD @PROCESS FOR VP

```

図 4.2 (続 き)

```

*/
*/I XE0SPD.2
*/IF DEF,F77VP,1
@PROCESS ADV(NOEVL)
*/
*/ (4) MOD ICVMGT FOR VP
*/
*/D SRCH3D.66
      IF((X(1,L).LE.Y(1,J)).AND.(Y(1,I).LT.X(1,L+1))) J(I) = L
*/
*/
*/
*/
*/      END OF PNCL1
*/-----+
*/IDENT D8BA
*/
*/ CORRECTION FOR AFDM Y8811 VERSION
*/      BY KOD '88.11.29
*/
*/ 1. CORRECT SPELL ERROR
*/
*/D DEFINP.68
      * / 1.0D-6,5.0D-3,1.0D-4,1000.0D0,0.010D,10.0D0,10.0D0
*/D AFPROG.12
      COMMON /AREA/ XAXIS, YAXIS
*/
*/ 2. CORRECT DBL ERROR
*/
*/D DEFINP.88
      DATA XICC,HLS/10.0D0,0.0010D,0.0001D0,0.025D0/
*/D DEFINP.141
      * ,ECRIT/6.003D6,6.057D6,3.2604D6/
*/
*/D AFPROG.8
*/IF DEF,DISP,5
*/I AFPROG.12
      REAL XAXIS,YAXIS
*/
*/-----+
*/IDENT D8CA
*/
*/ CORRECTION FOR AFDM Y8811 SEOS VERSION
*/      BY KOD : 88.12.23
*/
*/ 1. PURGE TPCRF
*/
*/D XE0SPRM.8
      LOGICAL SESPR,SPNDMF
*/
*/D DEFINP.213
      DATA NISOT,SPNDMF/30,.TRUE./
*/
*/D INIPRT.710
*/
*/IDENT D92A
*/
*/ CORRECTION FOR AFDM Y8901 VERSION
*/      BY KOD '88.2.2
*/
*/ 1. CORRECT SPELL ERROR
*/
*/D PAC8.9
      DATA AH,EPSRO,EPSTEM,LANCUT/1.D+3,1.0+3,0.10D,1.0+3,1.0+5/
*/
*/ 2. CORRECT DBL ERROR
*/
*/I ARTR.8
*/CALL IMPDBL
*/

```

12

12

13

13

図 4.2 (続 き)

```

//T201#AFD JOB (PWD),AFDM,MSGCLASS=X,
//  MSGLEVEL=(2,0),NOTIFY=T201#,
//  ATTR=(T4,C4,W3)'
//*****
//*      AFD M Y8903 M21A VERSION                *
//*      HISTORIAN/COMPILE/LINK                  *
//*****
//HISTOR EXEC PGM=HISTOR
//STEPLIB DD DISP=SHR,DSN=SYS9.HISTOR.LOAD
//FT05F001 DD *
HISTORIAN(P,T,F)
*LINE$ ,78
*READ ,9
*DEFINE DBL
*DEFINE FOURB
*DEFINE DISP
*/ FINE SESAME
*DEFINE KQGM2
*DEFINE IVIS
*DEFINE PSQ2
*/ FINE NOPHASE
*DEFINE ARCV1
*IDENT MYDIM
*D DIMEN.38,39
      *      IBM=3
      *      ,JBM=12
/*
//FT06F001 DD SYSOUT=X
//FT09F001 DD DISP=SHR,DSN=T201#.AFDM.Y8901.LANLCS.ADD.FORT
//          DD DISP=SHR,DSN=T201#.AFDM.PNCAF.FORT
//          DD DISP=SHR,DSN=T201#.AFDM.Y8903.PNCCS.FORT
//FT21F001 DD DISP=SHR,DSN=T201#.AFDM.Y8901.LANL.HISLIB
//FT22F001 DD UNIT=WORK,SPACE=(4080,(1000,50)),DISP=NEW
//FT23F001 DD DISP=(,PASS),DSN=&&SSOR,UNIT=WORK,
//          SPACE=(TRK,(50,10),RLSE),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//FT25F001 DD UNIT=WORK,SPACE=(4080,(500,50)),DISP=NEW
//*T26F001 DD DUMMY
//FT27F001 DD UNIT=WORK,SPACE=(4080,(500,50)),DISP=NEW
//*
//***** STEP FORT77 =====
//FORT EXEC PGM=JZK@FORT,COND=EVEN,
//      PARM='LC(78),OPT(3)'
//SYSPRINT DD DSN=T201#.AFDM.SLIST,DISP=SHR
//SYSUT1 DD UNIT=SYSSQ,SPACE=(TRK,(5,5))
//SYSLIN DD DISP=(,PASS),DSN=&&SOBJ,UNIT=WORK,
//        SPACE=(TRK,(50,10),RLSE),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//SYSIN DD DISP=(OLD,DELETE),DSN=&&SSOR
//*
//***** STEP LINK. =====
//LKED EXEC PGM=JQAL,COND=(8,LT,FORT),
//      PARM='LET,NOMAP,LIST,LINECOUNT(78)'
//SYSLIB DD DISP=SHR,DSN=SYS1.FORTLIB
//          DD DISP=SHR,DSN=T201#.SIMMER.ASHLIB,LABEL=(,,IN)
//          DD DISP=SHR,DSN=T201#.PULIB
//          DD DISP=SHR,DSN=SYS9.DISLIB
//          DD DISP=SHR,DSN=SYS9.INTLIB
//          DD DISP=SHR,DSN=SYS1.SSL2
//SYSUT1 DD UNIT=SYSSQ,SPACE=(1024,(200,20))
//SYSPRINT DD SYSOUT=X
//SYSLIN DD DISP=(OLD,DELETE),DSN=&&SOBJ
//SYSLMOD DD DSN=T201#.AFDM.LOAD(M21A1),DISP=(NEW,CATLG),
//          SPACE=(TRK,(30,10,1),RLSE),UNIT=DASD
//

```

図 4.3 AFD Mロードモジュール作成 JCL 例 (scalar版)

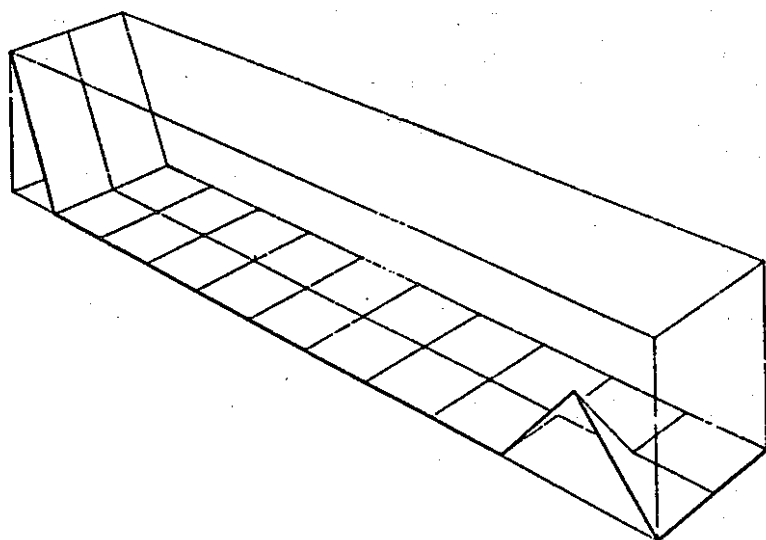

```

//T201#AFD JOB (PWD),AFDM,MSGCLASS=X,
//  MSGLEVEL=(2,0),NOTIFY=T201#,
//  ATTR=(T4,C7,W3)
//*****
//*      AFDM Y8903 M21A VERSION          *
//*      HISTORIAN/VP-COMPILE/LINK      *
//*****
//HISTOR EXEC PGM=HISTOR
//STEPLIB DD DISP=SHR,DSN=SYS9.HISTOR.LOAD
//FT05F001 DD *
HISTORIAN(P,T,F)
*LINE,78
*READ,9
*DEFINE DBL
*DEFINE FOURB
*DEFINE DISP
*/ FINE SESAME
*DEFINE KQQM2
*DEFINE IV1S
*DEFINE PSOL2
*/ FINE NOPHASE
*DEFINE ARCV1
*DEFINE F77VP
*IDENT MYDIM
*D DIMEN.38,39
      *      IBM=3
      *      ,JBM=12
/*
//FT06F001 DD SYSOUT=X
//FT09F001 DD DISP=SHR,DSN=T201#.AFDM.Y8901.LANLCS.ADD.FORT
//      DD DISP=SHR,DSN=T201#.AFDM.PNCAF.FORT
//      DD DISP=SHR,DSN=T201#.AFDM.Y8903.PNCCS.FORT
//FT21F001 DD DISP=SHR,DSN=T201#.AFDM.Y8901.LANL.HISLIB
//FT22F001 DD UNIT=WORK,SPACE=(4080,(1000,50)),DISP=NEW
//FT23F001 DD DISP=(,PASS),DSN=&&SSOR,UNIT=WORK,
//      SPACE=(TRK,(50,10),RLSE),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//FT25F001 DD UNIT=WORK,SPACE=(4080,(500,50)),DISP=NEW
//*T26F001 DD DUMMY
//FT27F001 DD UNIT=WRK,SPACE=(4080,(500,50)),DISP=NEW
//*
//***** STEP FORT77/VP =====
//FORT EXEC PGM=JZK@FORT,COND=EVEN,
//      PARM=( 'LC(78),VP,NOAE,ALC',
//      'PI(CVMGT,ICVMGT,LCVMGT)')
//SYSPRINT DD DSN=T2022.AFDM.VLIST,DISP=SHR
//SYSUT1 DD UNIT=SYSSQ,SPACE=(TRK,(10,10))
//SYSFIN DD UNIT=SYSSQ,SPACE=(TRK,(50,50))
//SYSLIN DD DISP=(,PASS),DSN=&&SOBJ,UNIT=WORK,
//      SPACE=(TRK,(50,10),RLSE),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//SYSIN DD DISP=(OLD,DELETE),DSN=&&SSOR
//*
//***** STEP LINK. =====
//LKED EXEC PGM=JQAL,COND=(8,LT,FORT),
//      PARM='LET,NOMAP,LIST,LINECOUNT(78)'
//SYSLIB DD DISP=SHR,DSN=SYS1.FORTLIB
//      DD DISP=SHR,DSN=T201#.SIMMER.ASMLIB,LABEL=(,,IN)
//      DD DISP=SHR,DSN=T201#.PULIB
//      DD DISP=SHR,DSN=SYS9.DISLIB
//      DD DISP=SHR,DSN=SYS9.INTLIB
//      DD DISP=SHR,DSN=SYS1.SSL2
//SYSUT1 DD UNIT=SYSSQ,SPACE=(1024,(200,20))
//SYSPRINT DD SYSOUT=X
//SYSLIN DD DISP=(OLD,DELETE),DSN=&&SOBJ
//SYSLMOD DD DSN=T201#.AFDM.LOAD(M21A1V),DISP=(NEW,CATLG),
//      SPACE=(TRK,(30,10,1),RLSE),UNIT=DASD
//

```

図4.4 AFDMロードモジュール作成JCL例(VP版)

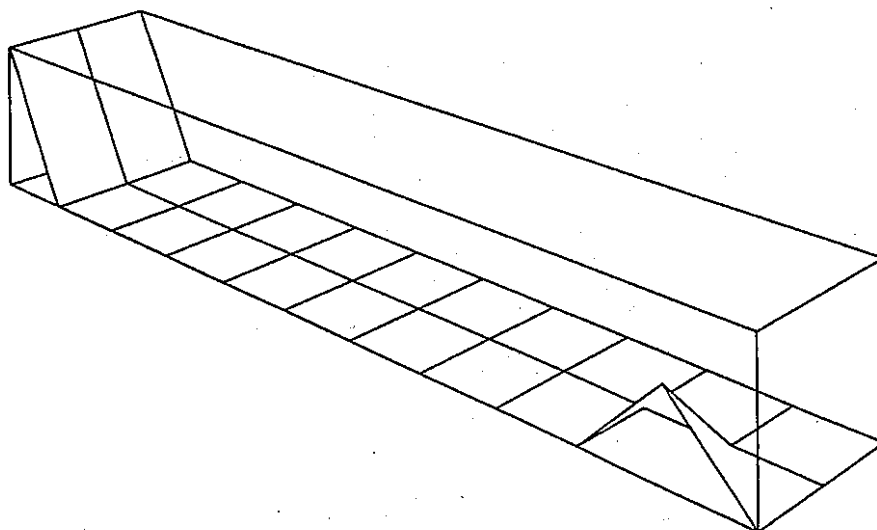
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR REGION I=(1, 3) J=(1, 12) VOLUME FRACTION
MINIMUM VALUE= -5.28618E-05 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

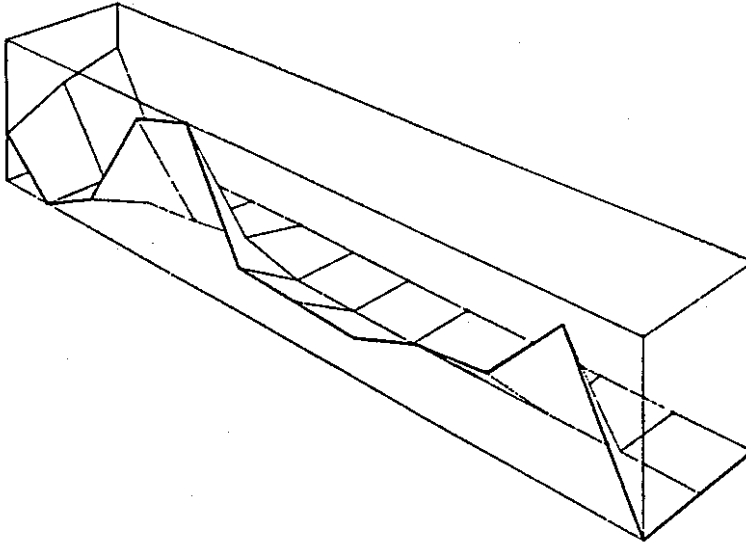


VAPOR REGION I=(1, 3) J=(1, 12) VOLUME FRACTION
MINIMUM VALUE= -5.28617E-05 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(b) F A C O M 版

図4.5 AFDM出力例 (Y8811版MINPA1ケース)

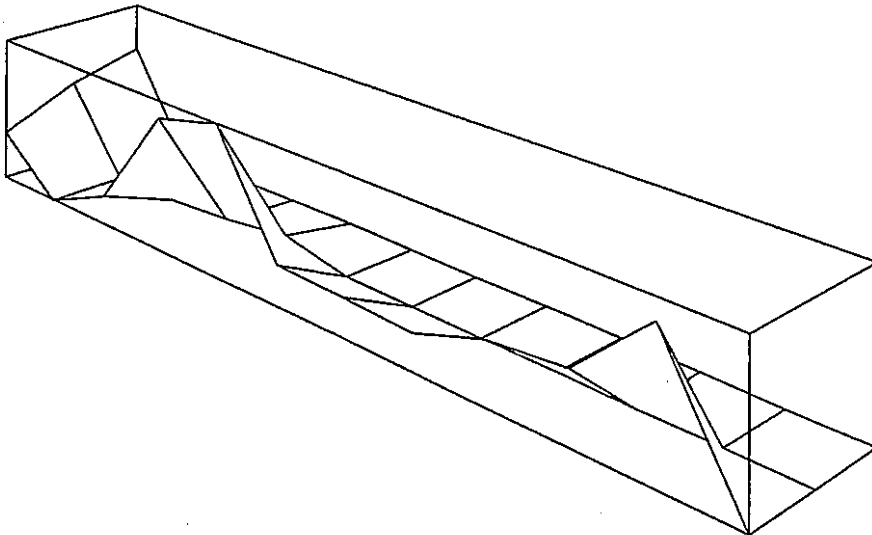
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -3.24881E-04 TIME = 64.168 MS.
MAXIMUM VALUE= 9.27675E-01

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

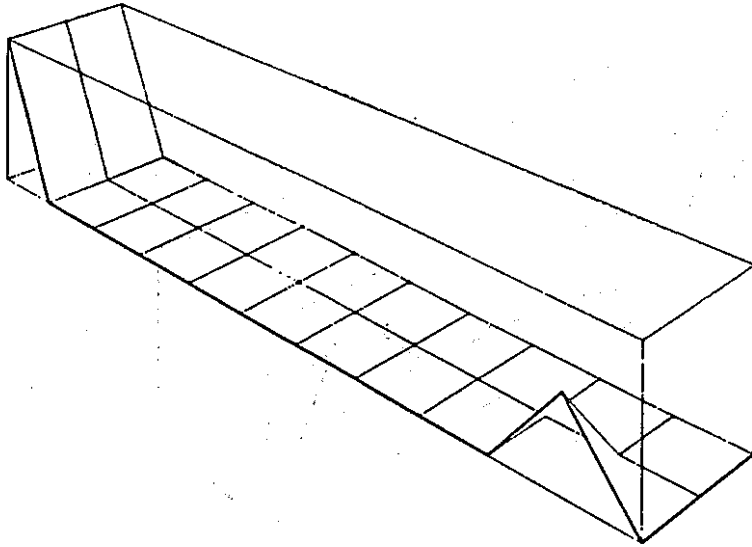


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -3.24881E-04 TIME = 64.168 MS.
MAXIMUM VALUE= 9.27675E-01

(b) F A C O M 版

図 4.5 (続 き)

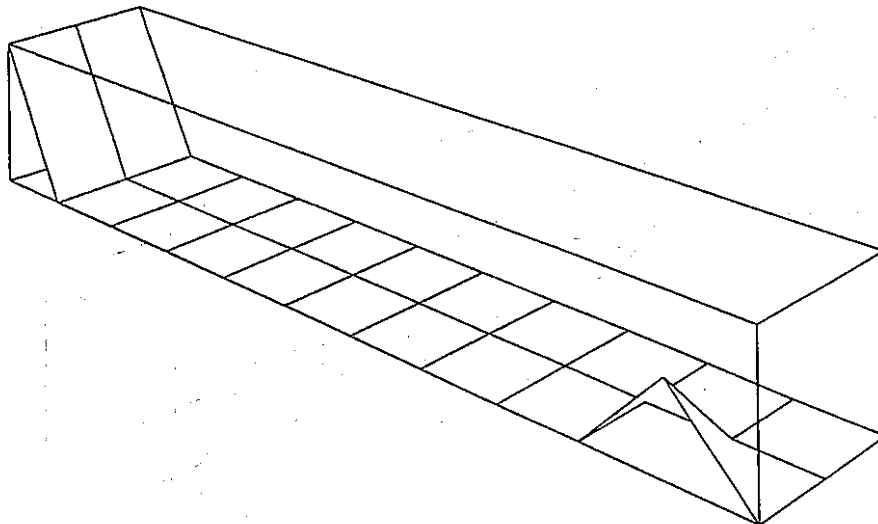
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR REGION I=(1. 3) J=(1. 12)
MINIMUM VALUE= -5.28618E-05 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

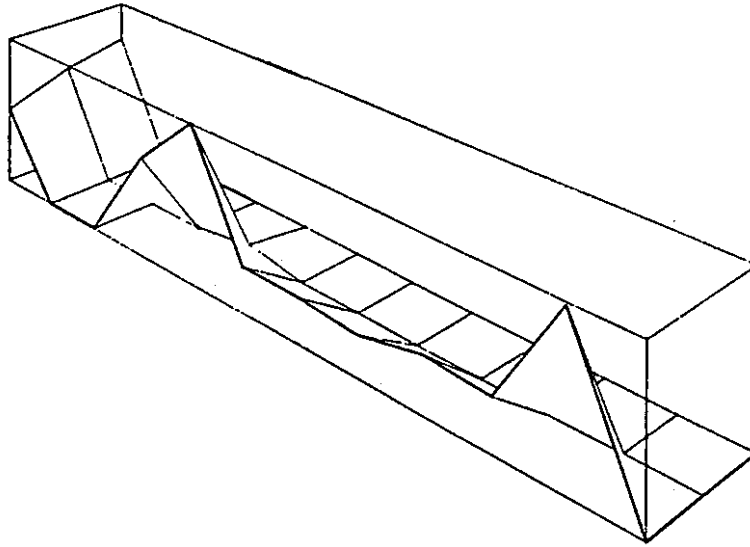


VAPOR REGION I=(1. 3) J=(1. 12)
MINIMUM VALUE= -5.28617E-05 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(b) F A C O M 版

図 4. 6 A F D M 出力例 (Y 8 8 1 1 版 M 1 P A 1 ケース)

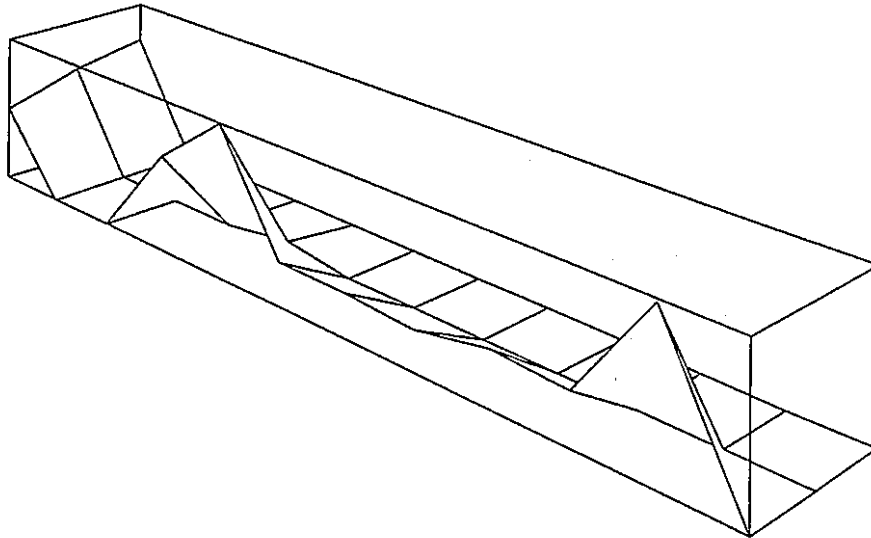
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -2.91094E-04 TIME = 55.842 MS.
MAXIMUM VALUE= 9.21715E-01

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

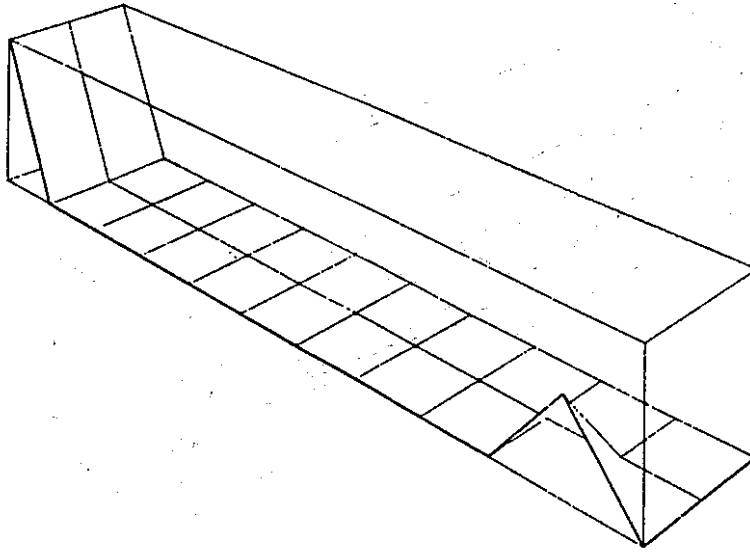


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -2.91094E-04 TIME = 55.842 MS.
MAXIMUM VALUE= 9.21715E-01

(b) F A C O M 版

図 4.6 (続 き)

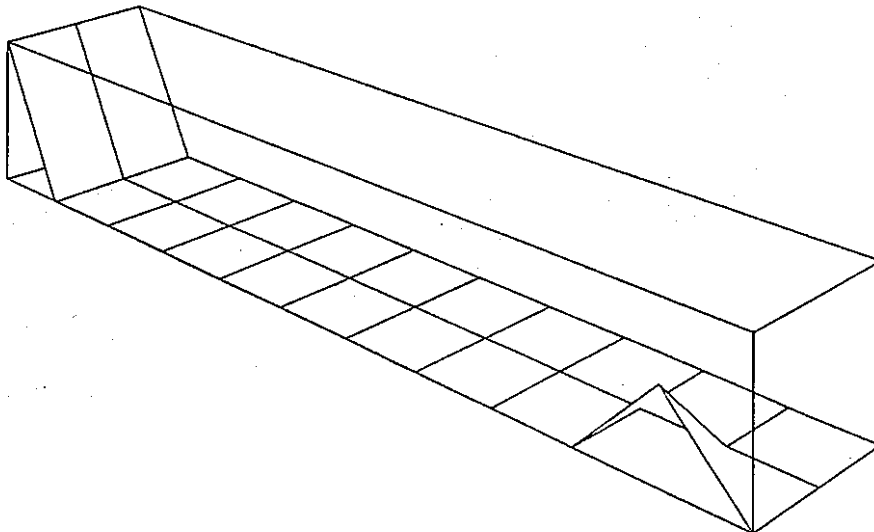
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -5.28618E-05 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

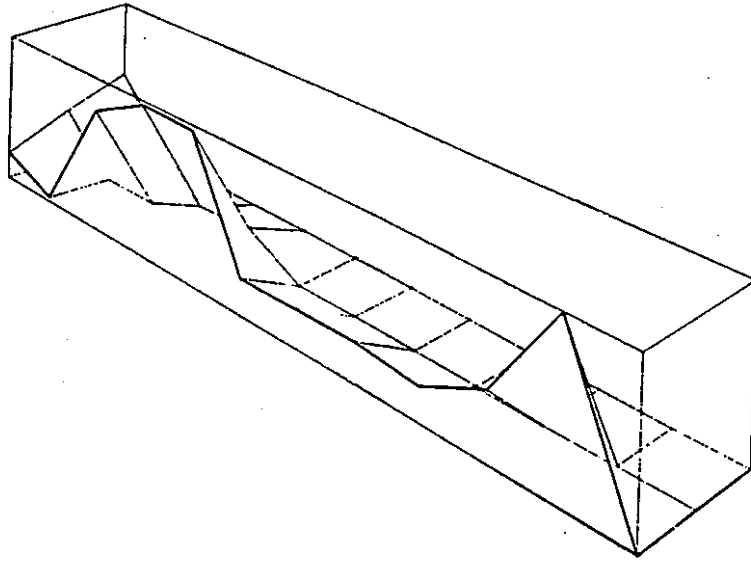


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -5.28617E-05 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(b) F A C O M 版

図 4.7 A F D M 出力例 (Y 8 8 1 1 版 M 2 1 A 1 ケース)

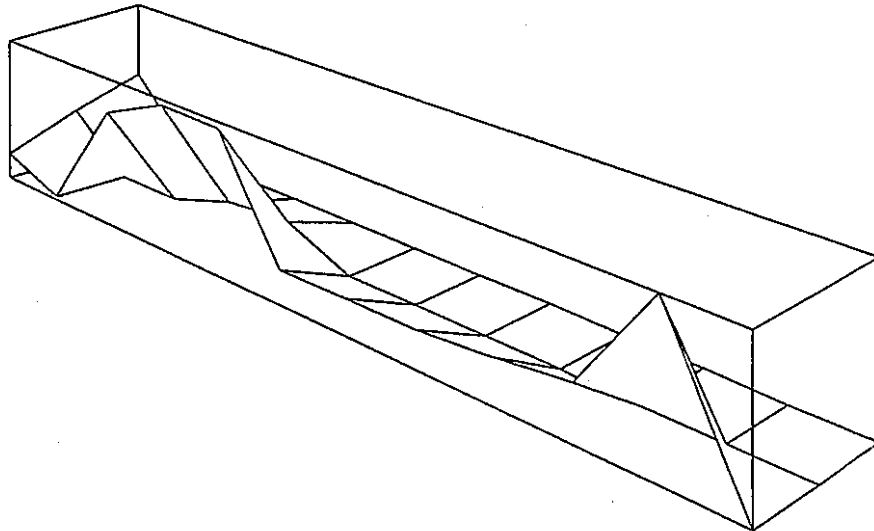
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -2.33186E-04 TIME = 90.762 MS.
MAXIMUM VALUE= 9.99135E-01

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

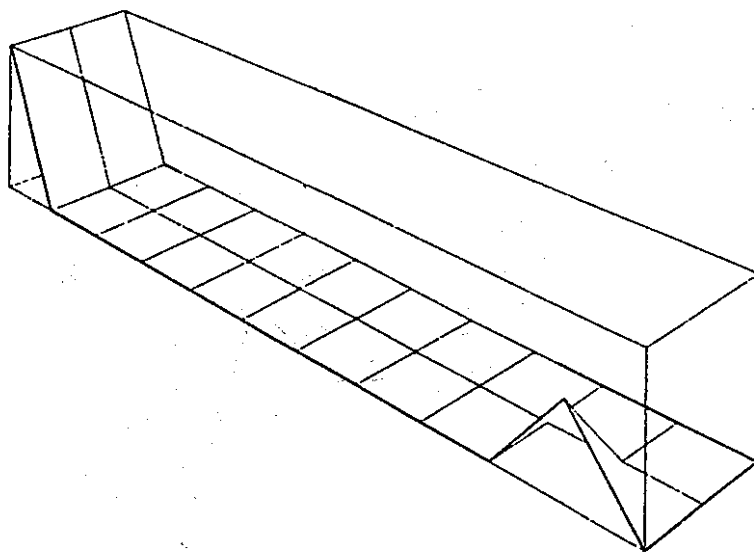


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -2.28820E-04 TIME = 90.049 MS.
MAXIMUM VALUE= 9.98993E-01

(b) F A C O M 版

図 4.7 (続 き)

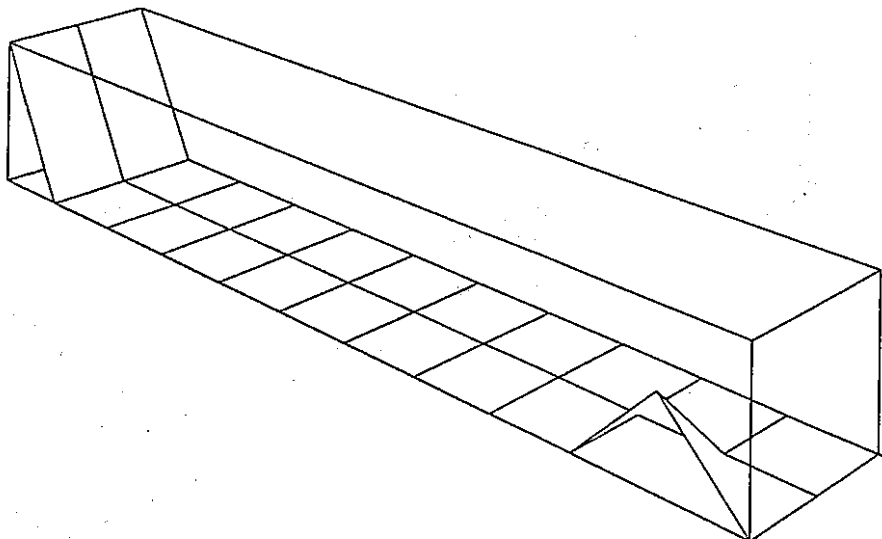
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= 0.00000E+00 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

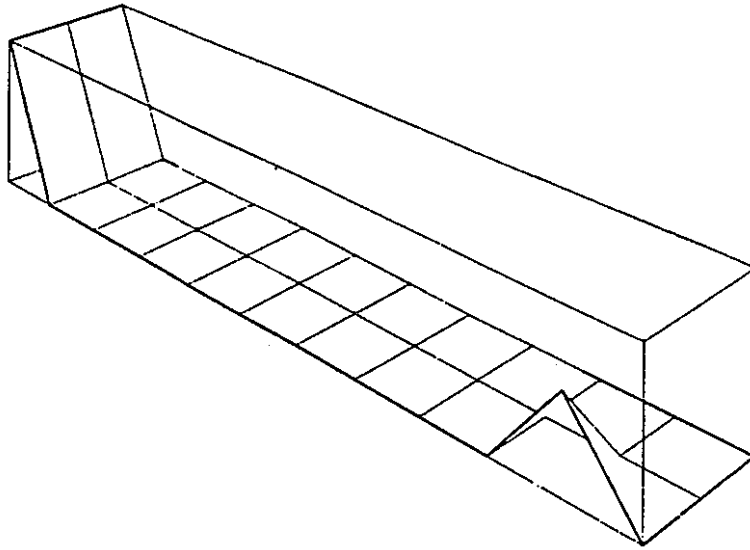


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= 0.00000E+00 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(b) F A C O M 版

図4.8 AFDM出力例 (Y8811版MINPS1ケース)

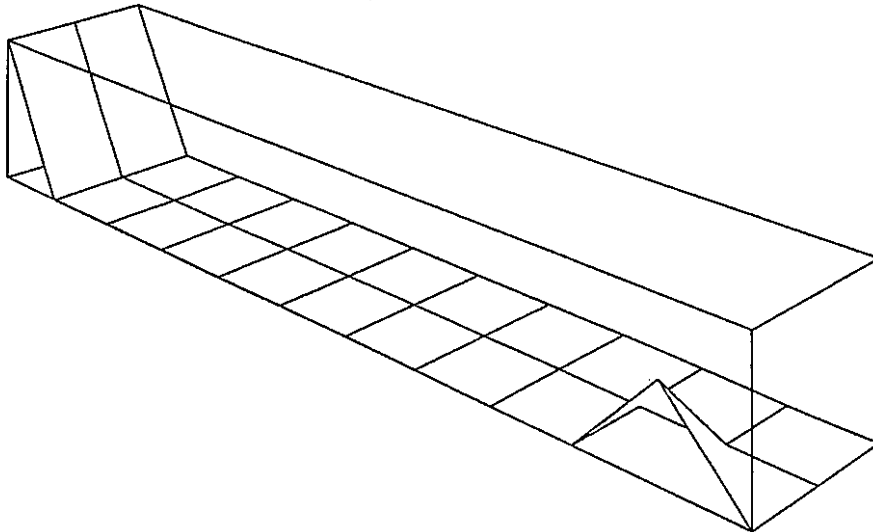
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR REGION I=(1, 3) J=(1, 12) VOLUME FRACTION
MINIMUM VALUE= -4.77730E-23 TIME = 2.647 MS.
MAXIMUM VALUE= 9.99988E-01

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

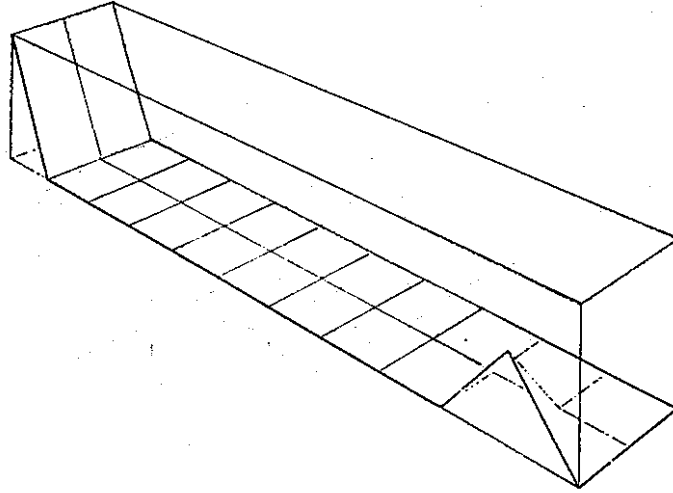


VAPOR REGION I=(1, 3) J=(1, 12) VOLUME FRACTION
MINIMUM VALUE= -4.77730E-23 TIME = 2.647 MS.
MAXIMUM VALUE= 9.99988E-01

(b) F A C O M 版

図 4.8 (続 き)

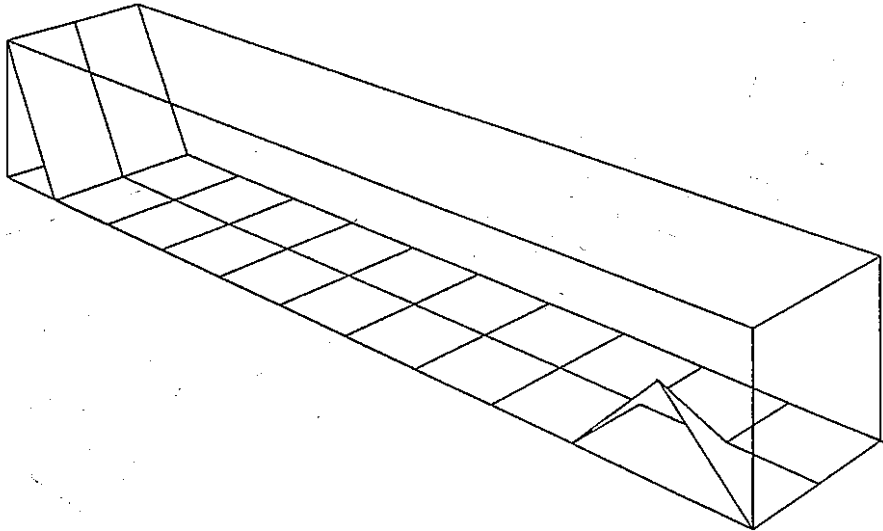
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
 REGION I=(1, 3) J=(1, 12)
 MINIMUM VALUE= 0.00000E+00 TIME = 0.000 MS.
 MAXIMUM VALUE= 1.00000E+00

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

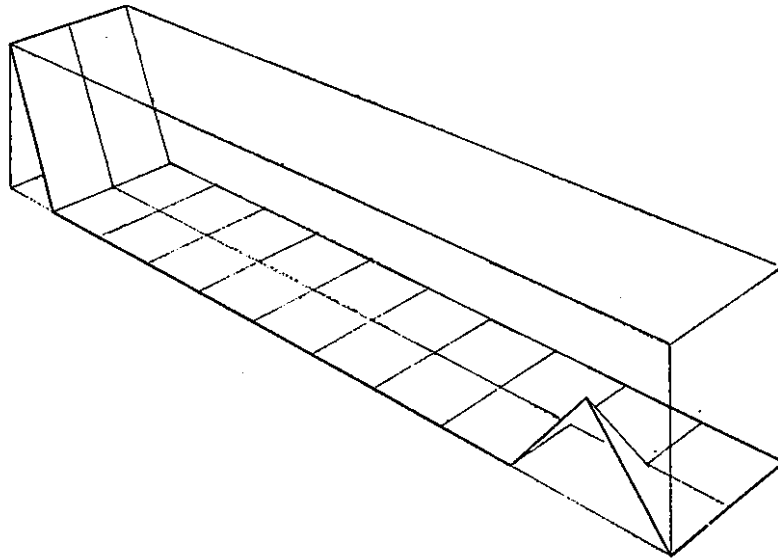


VAPOR VOLUME FRACTION
 REGION I=(1, 3) J=(1, 12)
 MINIMUM VALUE= 0.00000E+00 TIME = 0.000 MS.
 MAXIMUM VALUE= 1.00000E+00

(b) F A C O M 版

図 4.9 A F D M 出力例 (Y 8 8 1 1 版 M 1 P S 1 ケース)

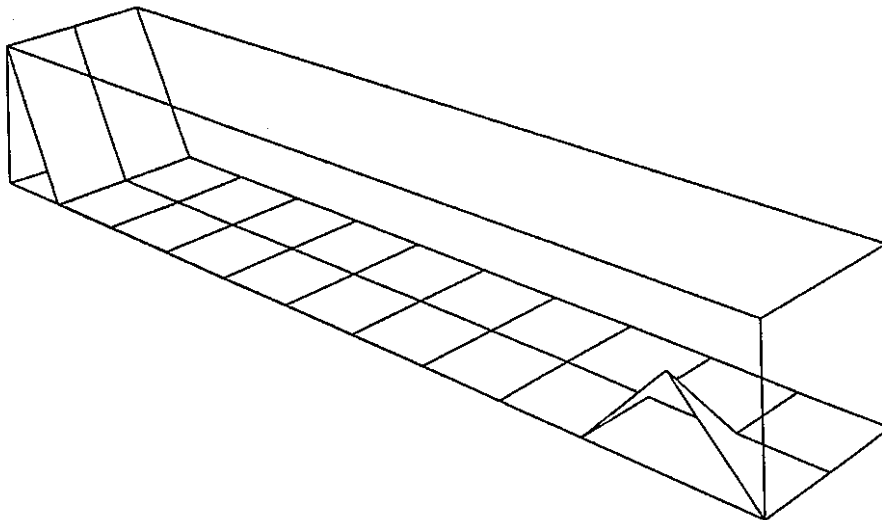
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= 3.06733E-23
MAXIMUM VALUE= 9.99996E-01 2.135 MS.

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

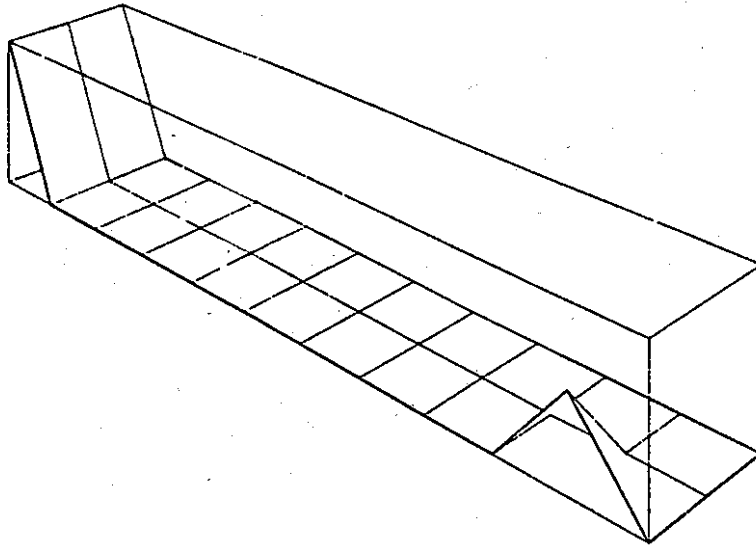


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -3.06733E-23
MAXIMUM VALUE= 9.99996E-01 TIME = 2.135 MS.

(b) F A C O M 版

図 4.9 (続 き)

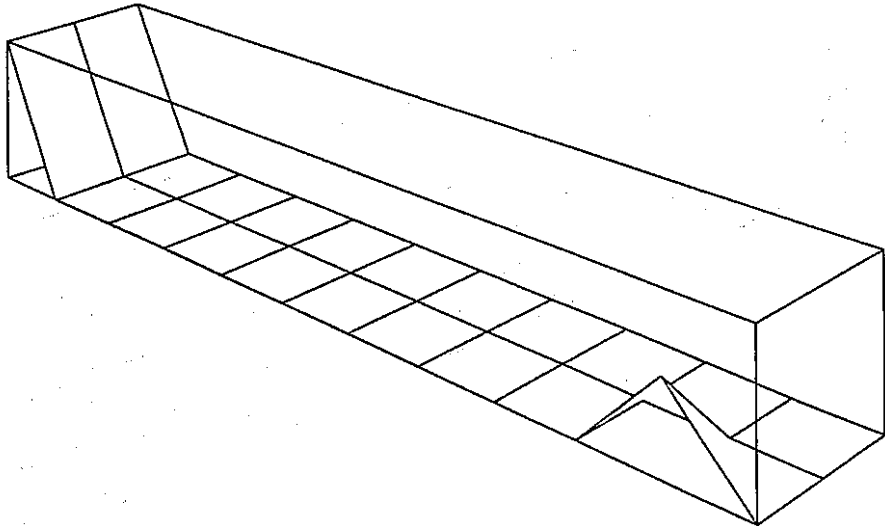
LITTLE WORK ENERGY PROBLEM FOR AFDM



```
VAPOR          VOLUME FRACTION
REGION I=( 1, 5) J=( 1, 12)
MINIMUM VALUE= 0.00000E+00   TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00
```

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

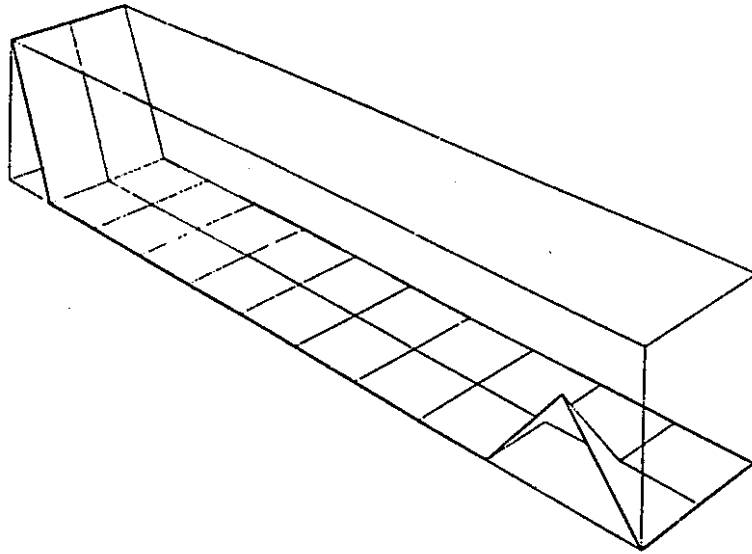


```
VAPOR          VOLUME FRACTION
REGION I=( 1, 3) J=( 1, 12)
MINIMUM VALUE= 0.00000E+00   TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00
```

(b) F A C O M 版

図 4.10 AFDM出力例 (Y8811版M21S1ケース)

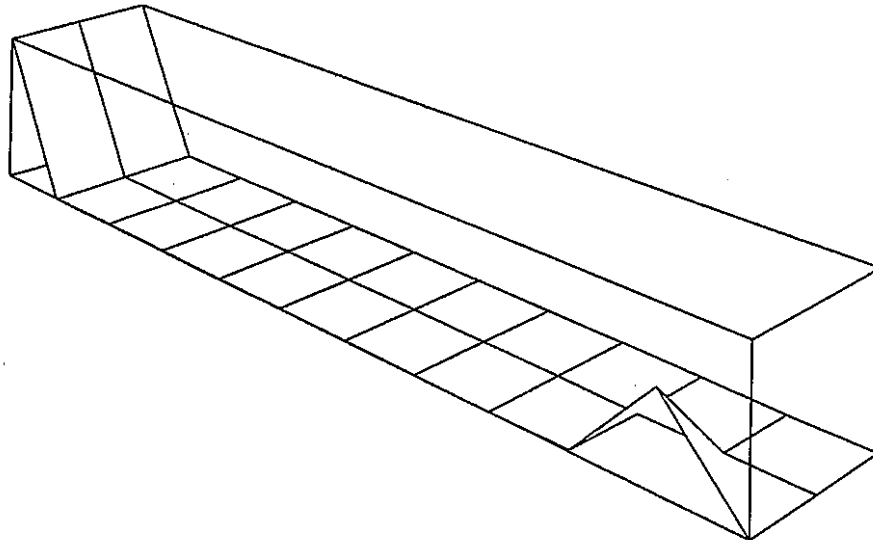
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -6.76635E-24 TIME = 10.426 MS.
MAXIMUM VALUE= 9.88417E-01

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

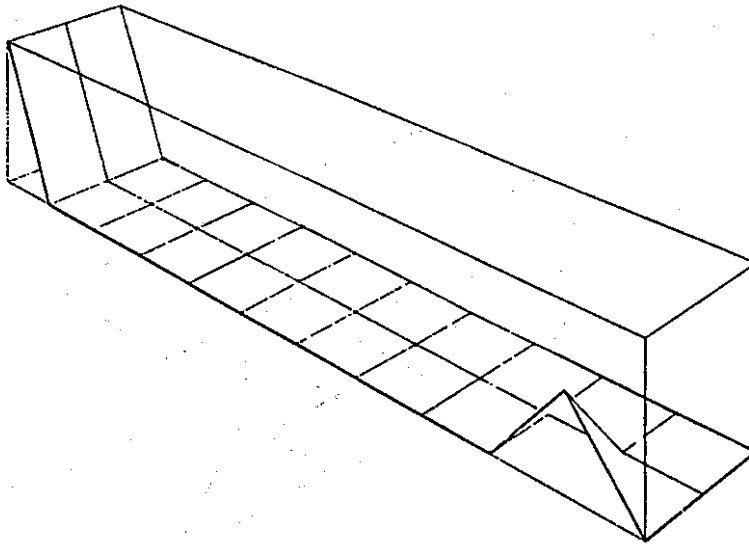


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -4.52837E-24 TIME = 9.977 MS.
MAXIMUM VALUE= 9.89758E-01

(b) F A C O M 版

図 4.10 (続 き)

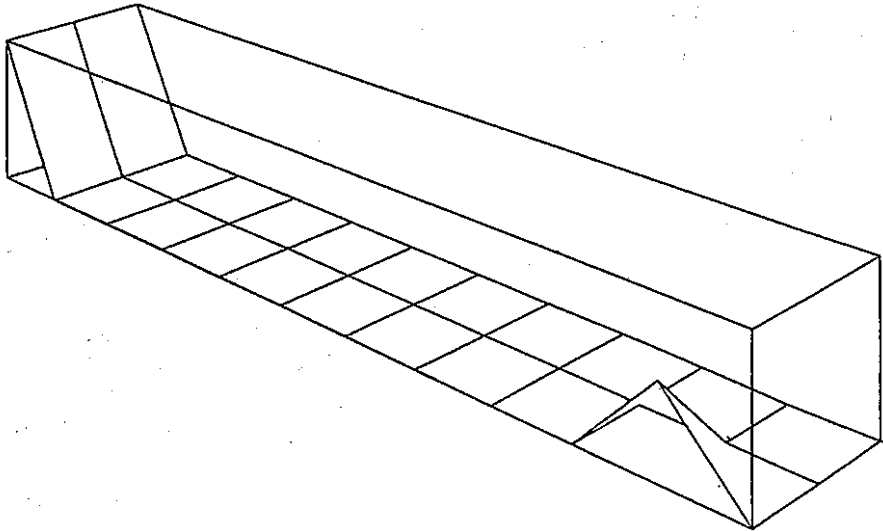
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= 0.00000E+00 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(a) CRAY版

LITTLE WORK ENERGY PROBLEM FOR AFDM

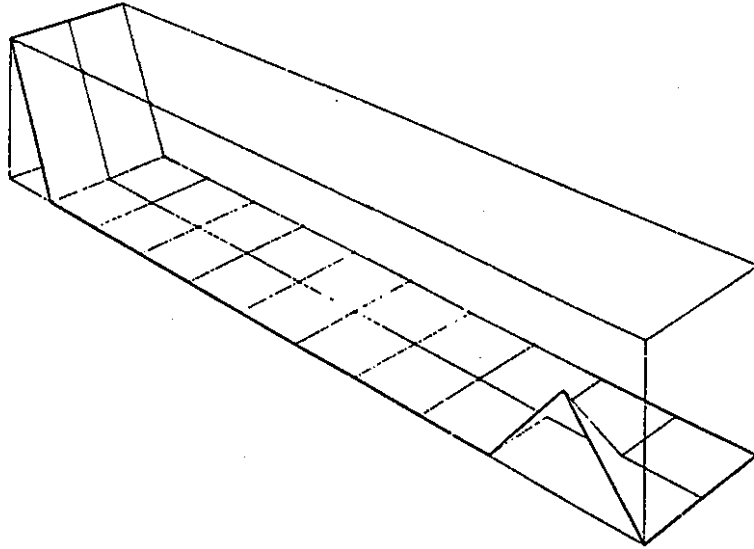


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= 0.00000E+00 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(b) FACOM版

図4.11 AFDM出力例 (Y8811版K1NPS1ケース)

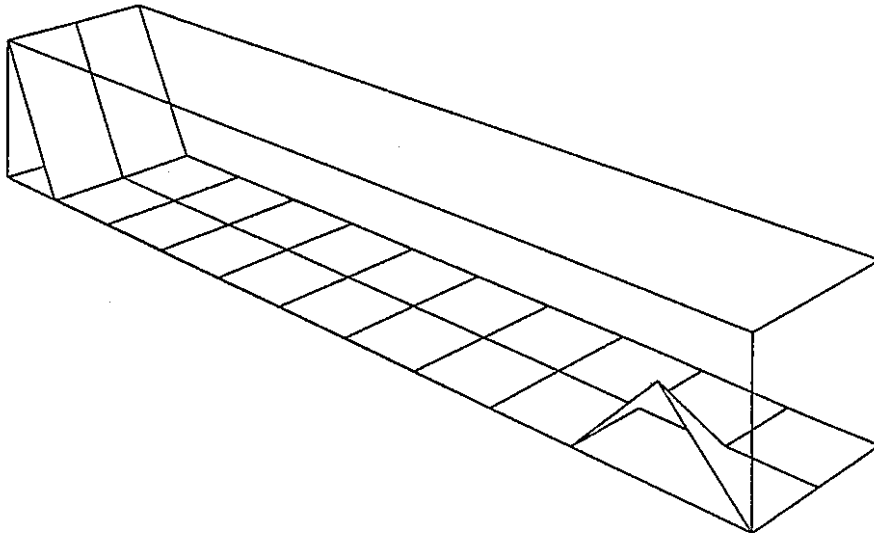
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -4.76703E-23 TIME = 2.647 MS.
MAXIMUM VALUE= 9.99928E-01

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

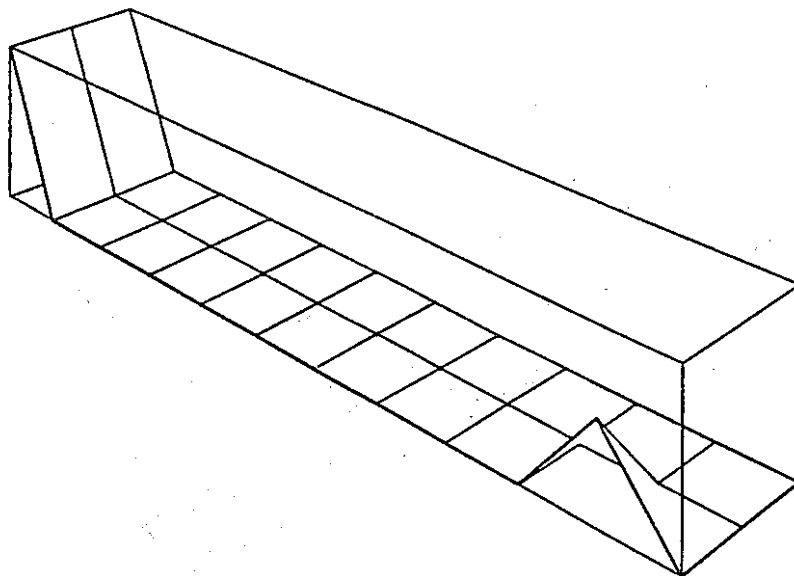


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -4.76703E-23 TIME = 2.647 MS.
MAXIMUM VALUE= 9.99988E-01

(b) F A C O M 版

図 4.11 (続 き)

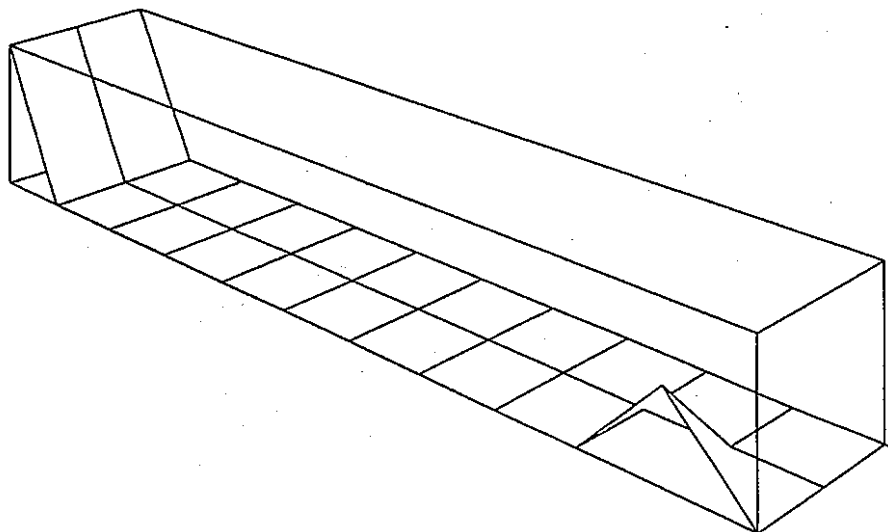
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= 0.00000E+00 TIME == 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

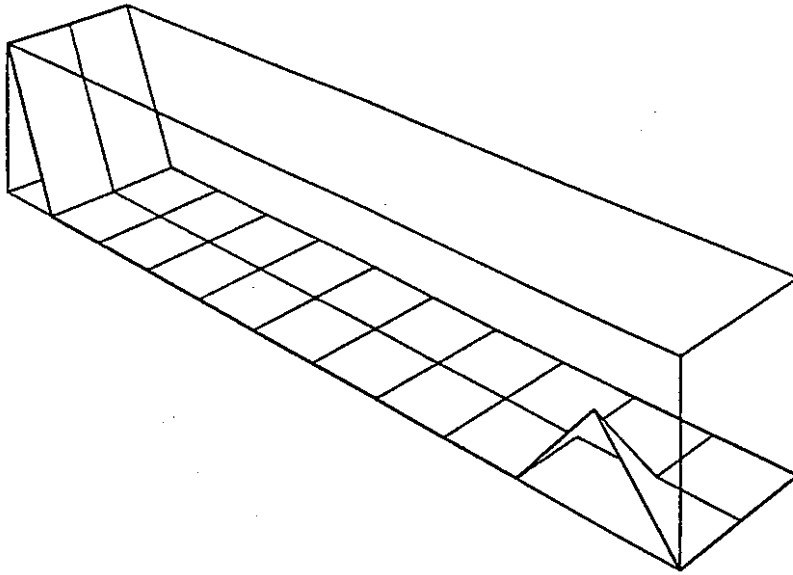


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= 0.00000E+00 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(b) F A C O M 版

図4.12 AFDM出力例 (Y8811版K1PS1ケース)

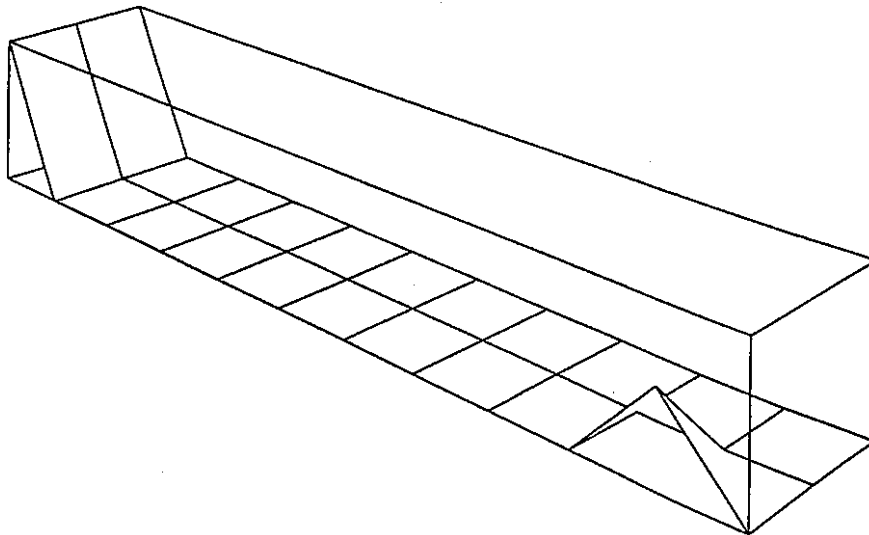
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -3.14759E-23 TIME = 2.204 MS.
MAXIMUM VALUE= 9.99994E-01

(a) CRAY版

LITTLE WORK ENERGY PROBLEM FOR AFDM

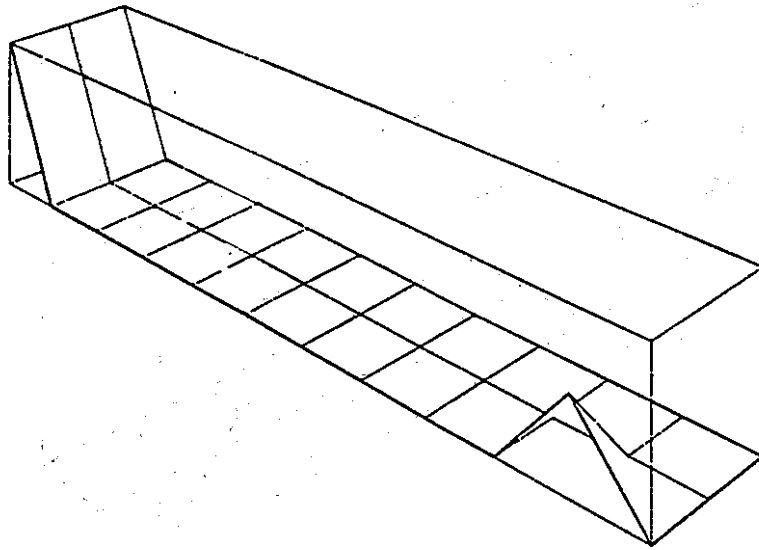


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -3.14759E-23 TIME = 2.204 MS.
MAXIMUM VALUE= 9.99994E-01

(b) FACOM版

图 4.12 (続 き)

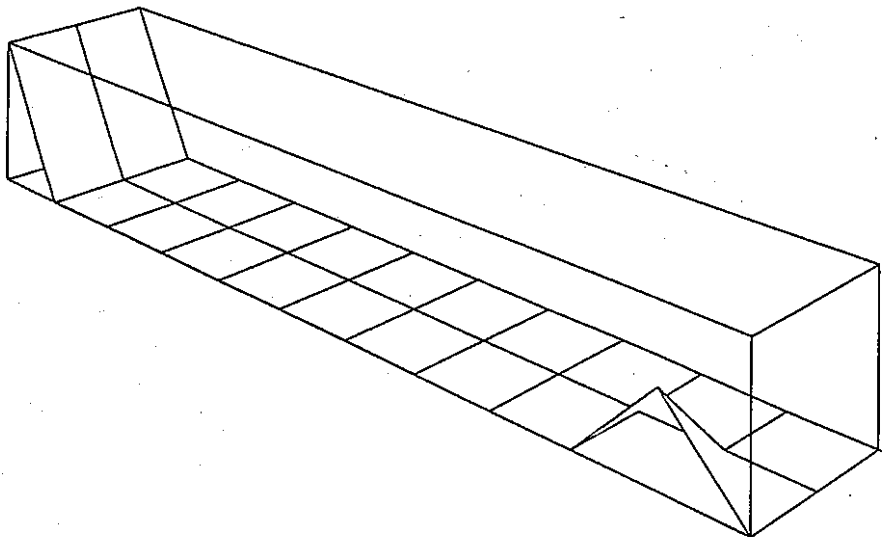
LITTLE WORK ENERGY PROBLEM FOR AFDM



```
VAPOR          VOLUME FRACTION  
REGION I=( 1, 3) J=( 1, 12)  
MINIMUM VALUE= 0.00000E+00    TIME = 0.000 MS.  
MAXIMUM VALUE= 1.00000E+00
```

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

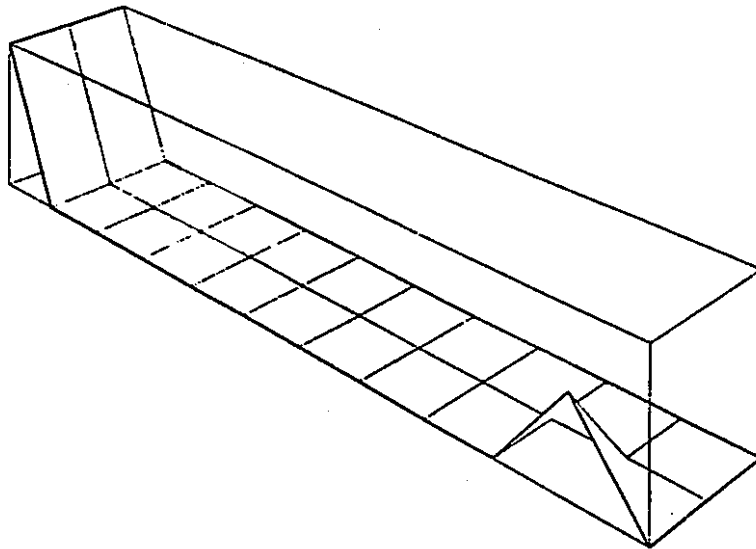


```
VAPOR          VOLUME FRACTION  
REGION I=( 1, 3) J=( 1, 12)  
MINIMUM VALUE= 0.00000E+00    TIME = 0.000 MS.  
MAXIMUM VALUE= 1.00000E+00
```

(b) F A C O M 版

図 4.13 A F D M 出力例 (Y 8 8 1 1 版 K 2 1 S 1 ケース)

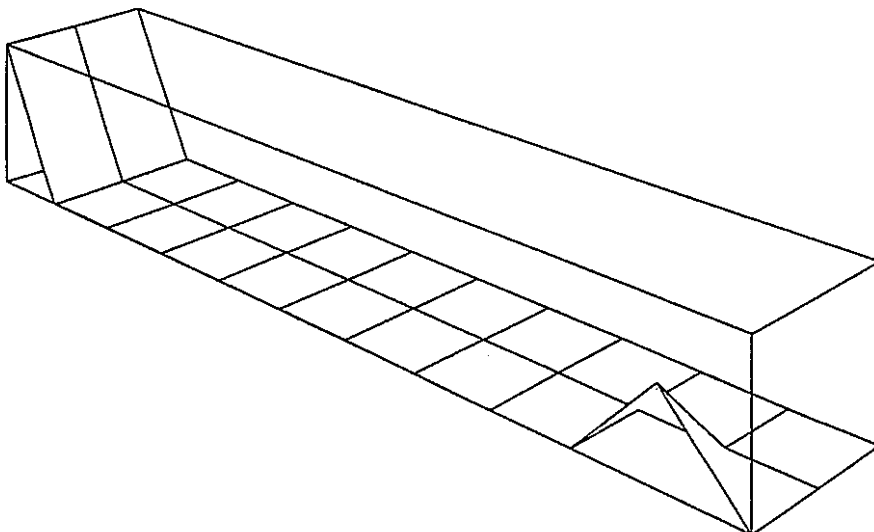
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR REGION I=(1, 3) J=(1, 12) VOLUME FRACTION
MINIMUM VALUE= -6.76635E-24 TIME = 10.426 MS.
MAXIMUM VALUE= 9.98417E-01

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

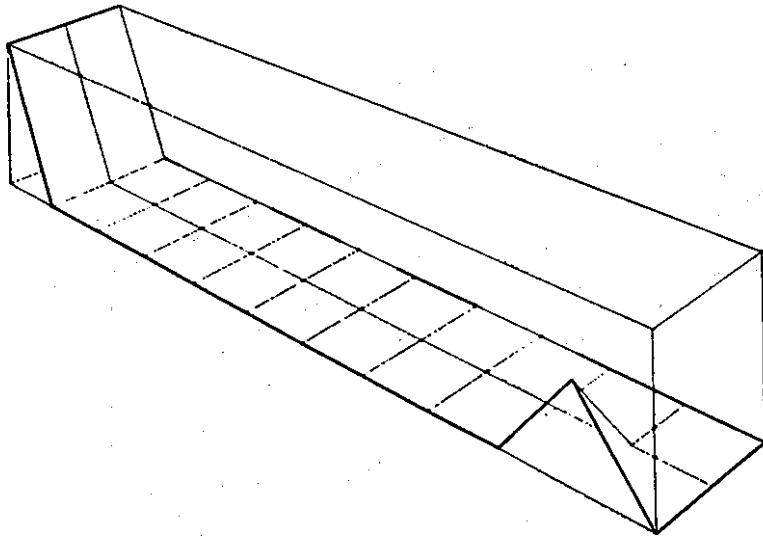


VAPOR REGION I=(1, 3) J=(1, 12) VOLUME FRACTION
MINIMUM VALUE= -2.71483E-23 TIME = 9.312 MS.
MAXIMUM VALUE= 9.91628E-01

(b) F A C O M 版

図 4.13 (続 き)

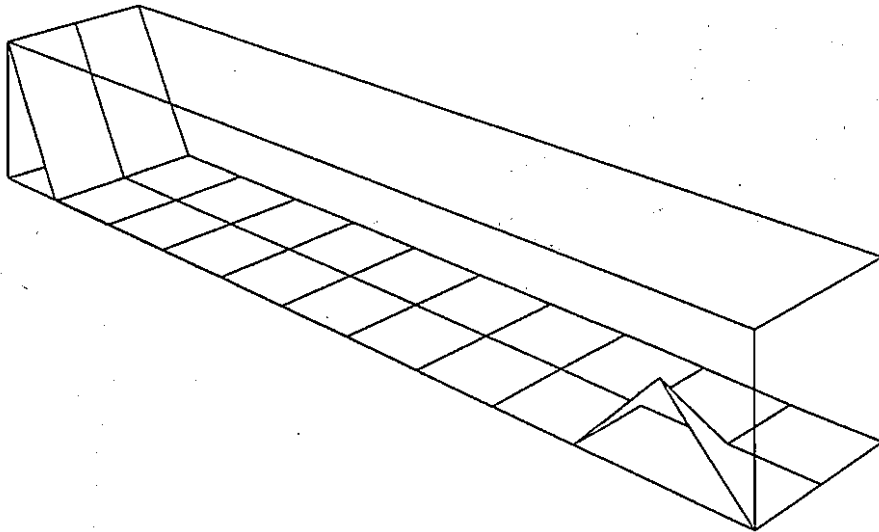
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR REGION I=(1, 3) J=(1, 12) VOLUME FRACTION
MINIMUM VALUE= -4.75727E-23 TIME = 5.325 MS.
MAXIMUM VALUE= 9.99116E-01

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

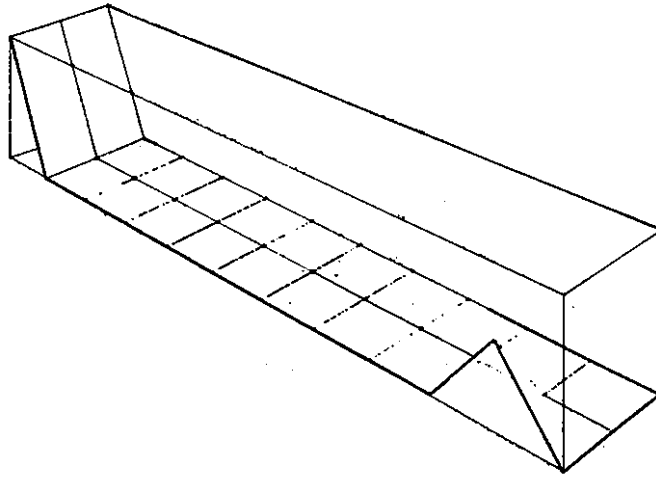


VAPOR REGION I=(1, 3) J=(1, 12) VOLUME FRACTION
MINIMUM VALUE= -4.85598E-23 TIME = 5.342 MS.
MAXIMUM VALUE= 9.99110E-01

(b) F A C O M 版

図 4.14 A F D M 出力例 (Y 8 9 0 1 版 M 2 1 S 1 ケース)

LITTLE WORK ENERGY PROBLEM FOR AFDM

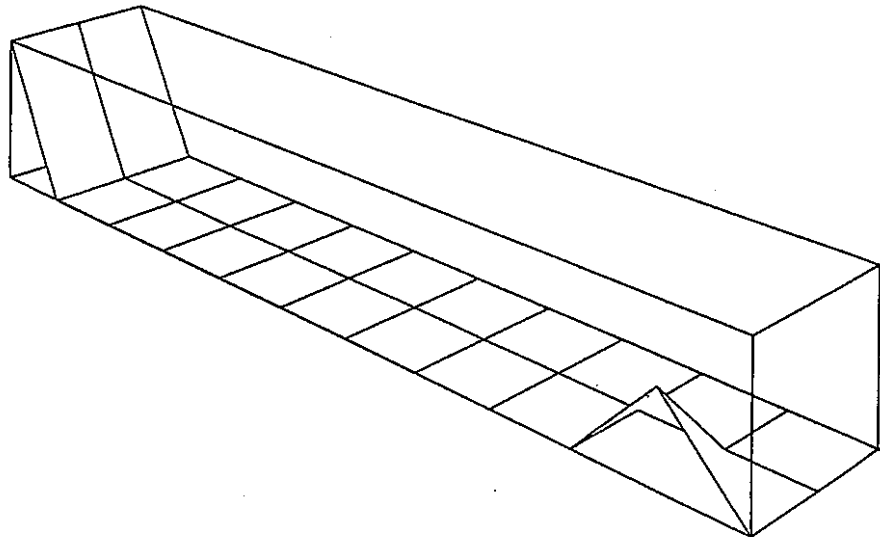


```
VAPOR          VOLUME FRACTION
REGION I=( 1, 3) J=( 1, 12)
MINIMUM VALUE= 0.00000E+00
MAXIMUM VALUE= 1.00000E+00
```

TIME = 0.000 MS.

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM



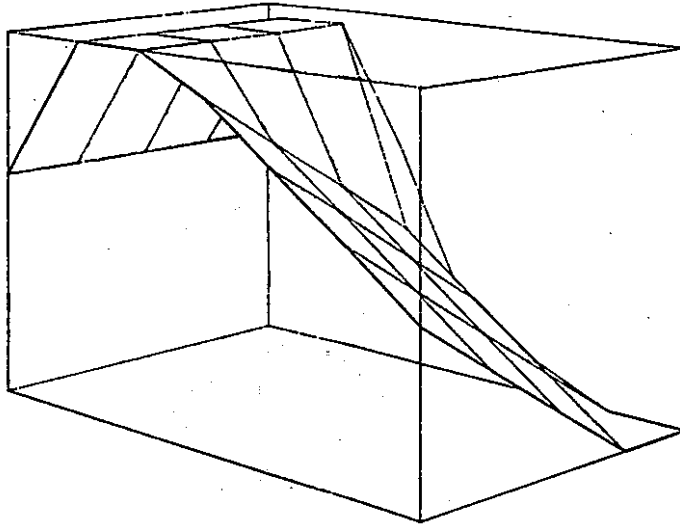
```
VAPOR          VOLUME FRACTION
REGION I=( 1, 3) J=( 1, 12)
MINIMUM VALUE= 0.00000E+00
MAXIMUM VALUE= 1.00000E+00
```

TIME = 0.000 MS.

(b) F A C O M 版

図 4.14 (続 き)

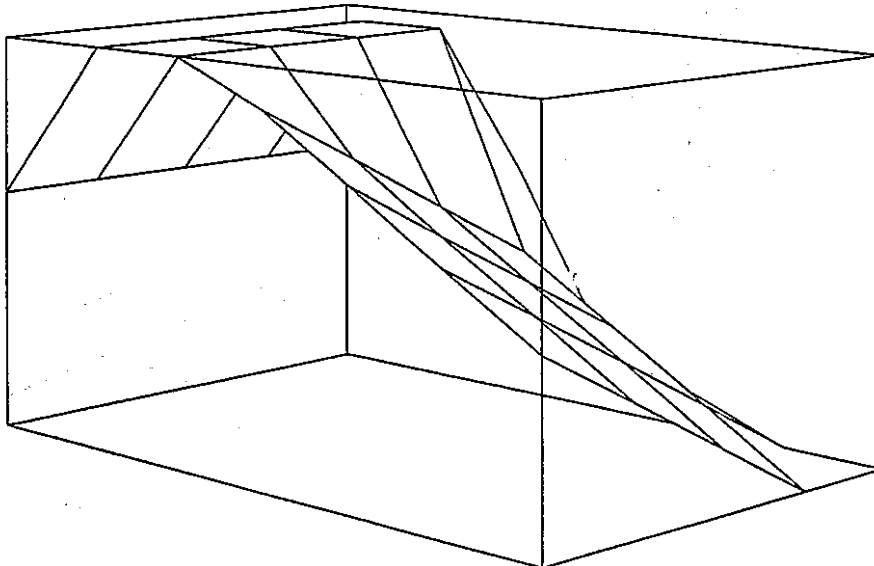
BOILING POOL TEST FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 5) J=(1, 7)
MINIMUM VALUE= 0.00000E+00 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(a) C R A Y 版

BOILING POOL TEST FOR AFDM

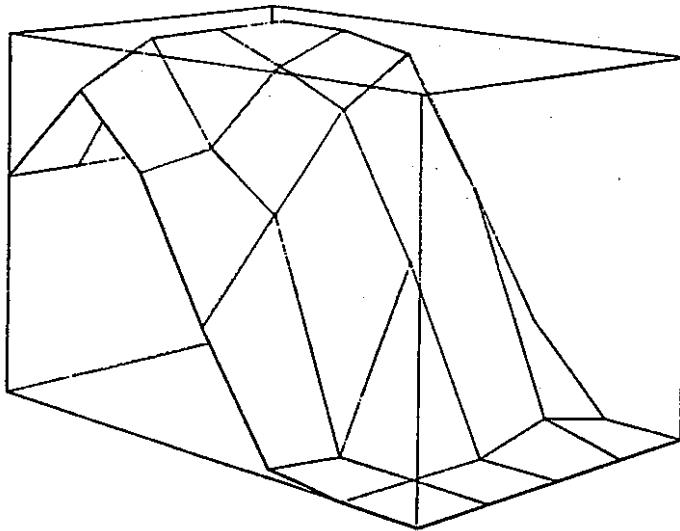


VAPOR VOLUME FRACTION
REGION I=(1, 5) J=(1, 7)
MINIMUM VALUE= 0.00000E+00 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(b) F A C O M 版

図4.15 AFDM出力例 (Y8901版M1PS2ケース)

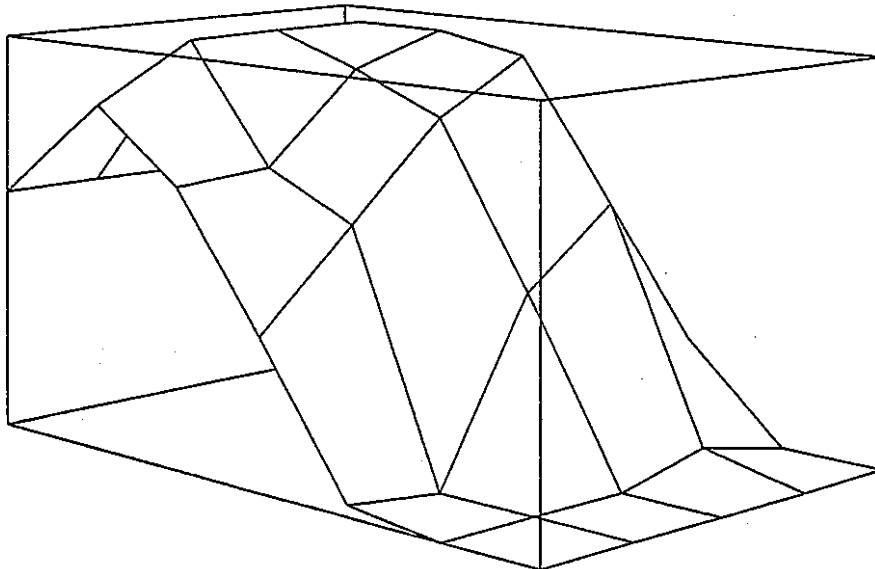
BOILING POOL TEST FOR AFDM



VAPOR REGION I=(1, 5) J=(1, 7) VOLUME FRACTION
MINIMUM VALUE= -8.76178E-27 TIME = 782.379 MS.
MAXIMUM VALUE= 9.99634E-01

(a) C R A Y 版

BOILING POOL TEST FOR AFDM

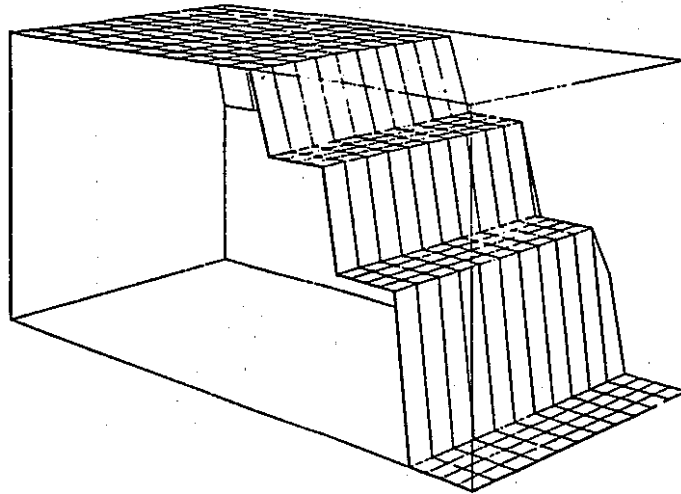


VAPOR REGION I=(1, 5) J=(1, 7) VOLUME FRACTION
MINIMUM VALUE= -9.21426E-27 TIME = 775.746 MS.
MAXIMUM VALUE= 9.99698E-01

(b) F A C O M 版

図 4.15 (続 き)

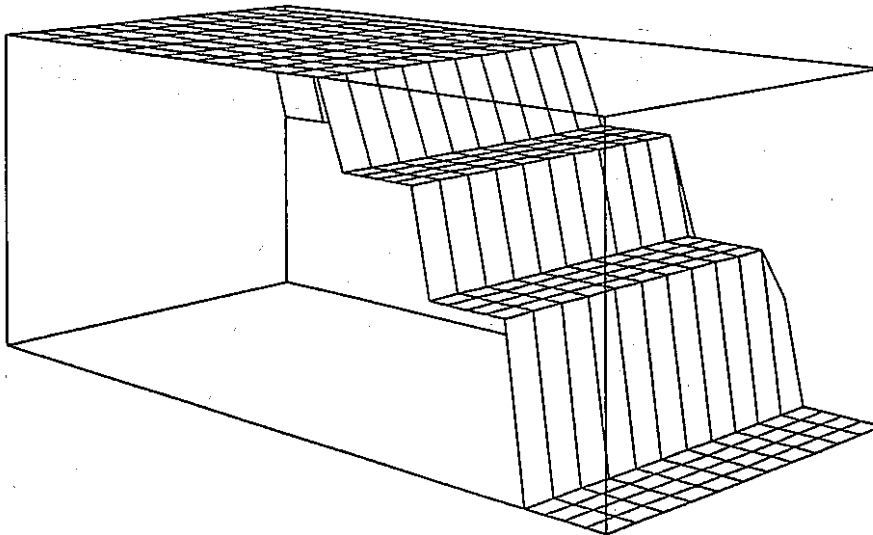
BOILING POOL TEST FOR AFDM



VAPOR REGION I=(1, 12) J=(1, 25) VOLUME FRACTION
MINIMUM VALUE= -1.55274E-08 TIME = 0.000 MS.
MAXIMUM VALUE= 9.99945E-01

(a) C R A Y 版

BOILING POOL TEST FOR AFDM

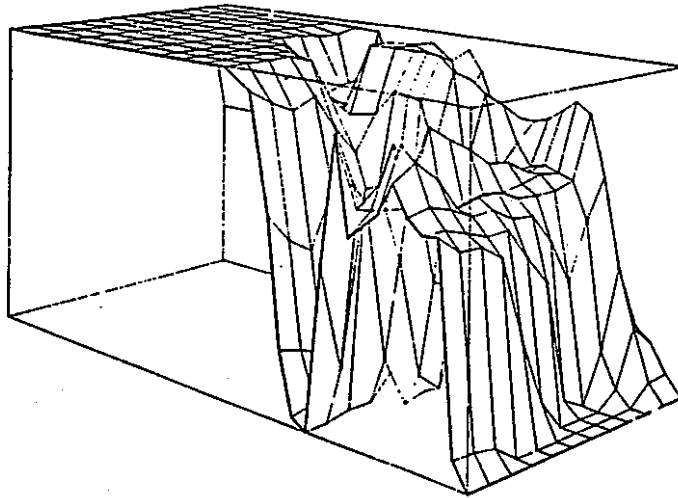


VAPOR REGION I=(1, 12) J=(1, 25) VOLUME FRACTION
MINIMUM VALUE= -1.55274E-08 TIME = 0.000 MS.
MAXIMUM VALUE= 9.99945E-01

(b) F A C O M 版

図 4.16 A F D M 出力例 (Y 8 9 0 1 版 M 2 1 A 3 ケース)

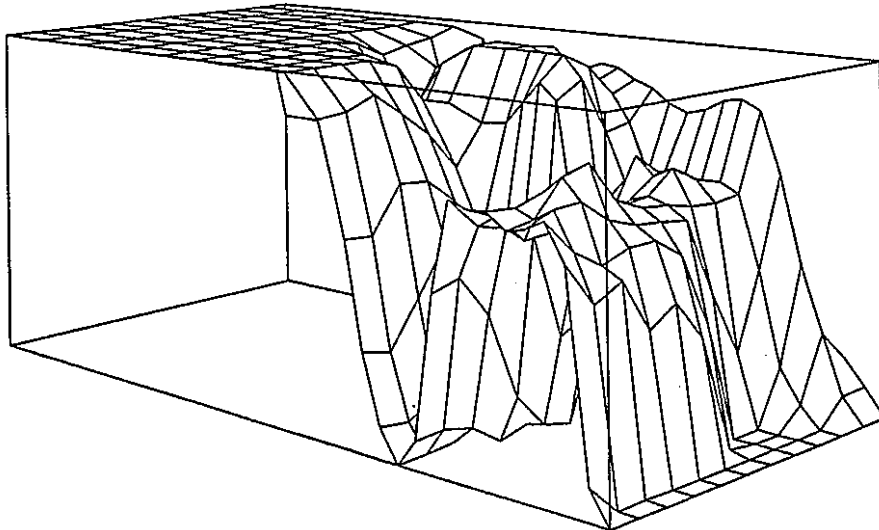
BOILING POOL TEST FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 12) J=(1, 25)
MINIMUM VALUE= -1.22934E-06 TIME = 810.628 MS.
MAXIMUM VALUE= 9.99933E-01

(a) C R A Y 版

BOILING POOL TEST FOR AFDM

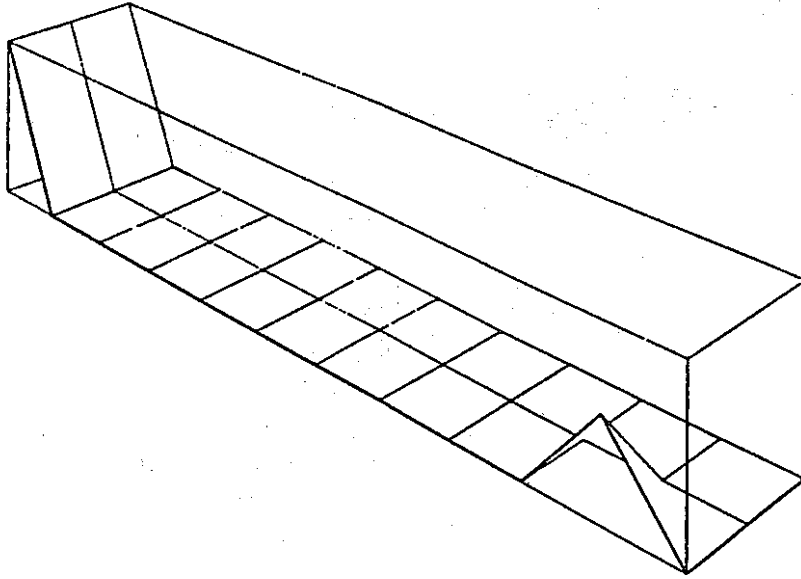


VAPOR VOLUME FRACTION
REGION I=(1, 12) J=(1, 25)
MINIMUM VALUE= -1.21203E-06 TIME = 818.910 MS.
MAXIMUM VALUE= 9.99928E-01

(b) F A C O M 版

図 4.16 (続 き)

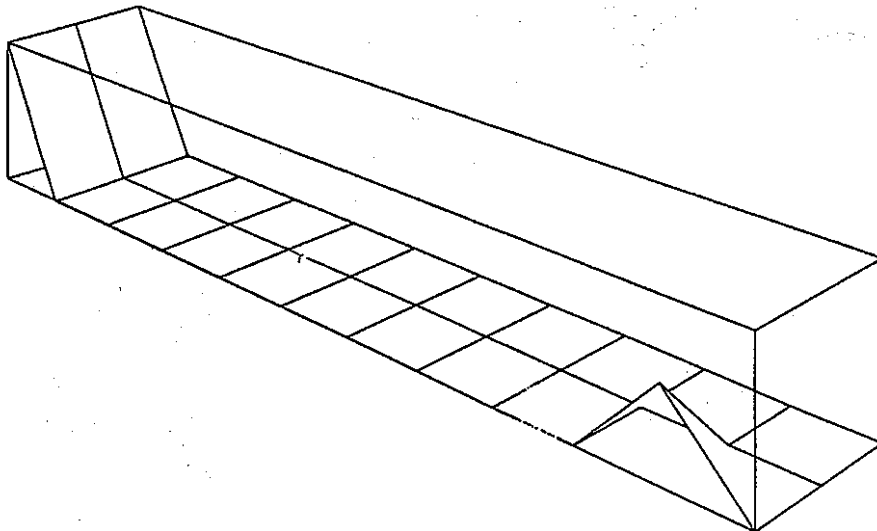
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -5.28618E-05 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM

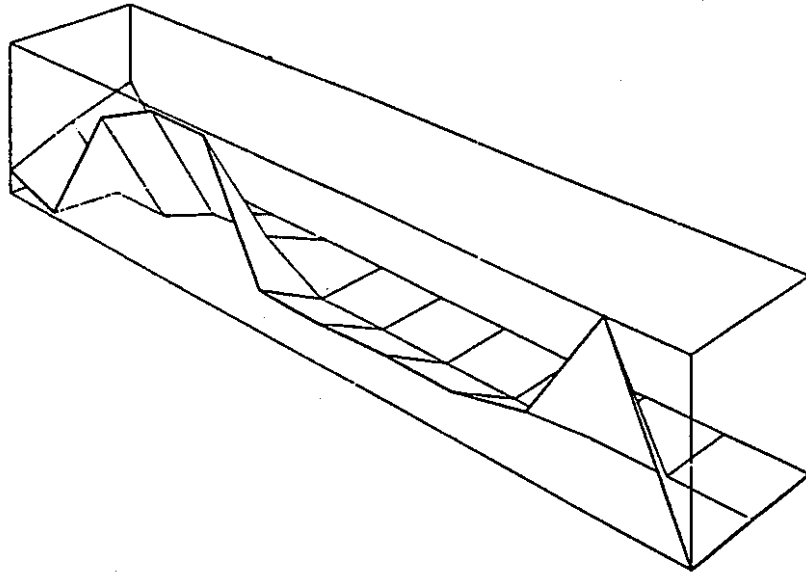


VAPOR VOLUME FRACTION
REGION I=(1, 3) J=(1, 12)
MINIMUM VALUE= -5.28617E-05 TIME = 0.000 MS.
MAXIMUM VALUE= 1.00000E+00

(b) F A C O M 版

図 4.17 A F D M 出力例 (Y 8 9 0 3 版 M 2 1 A 1 ケース)

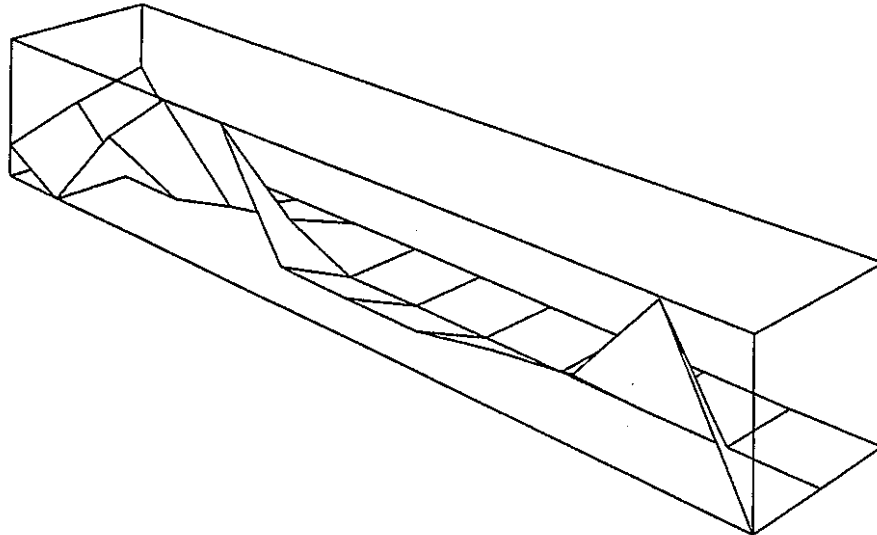
LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1. 3) J=(1. 12)
MINIMUM VALUE= -2.84910E-04 TIME = 90.957 MS.
MAXIMUM VALUE= 9.99037E-01

(a) C R A Y 版

LITTLE WORK ENERGY PROBLEM FOR AFDM



VAPOR VOLUME FRACTION
REGION I=(1. 3) J=(1. 12)
MINIMUM VALUE= -2.89432E-04 TIME = 93.370 MS.
MAXIMUM VALUE= 9.59201E-01

(b) F A C O M 版

図 4.17 (続 き)

```

//T201#AFD JOB (PWD),AFDM,MSGCLASS=X,
//  MSGLEVEL=(2,0),NOTIFY=T201#,
//  ATTR=(T4,C4,W3)
//*****
//*      AFD M Y8903 M21A VERSION      *
//*****
//AFDM EXEC PGM=M21A1
//STEPLIB DD DSN=T201#.AFDM.LOAD,DISP=SHR
//FT05F001 DD DISP=SHR,LABEL=(,,,IN),DSN=T201#.AFDM.DATA(LWEP)
//FT06F001 DD SYSOUT=X
//FT07F001 DD DUMMY
//FT08F001 DD DUMMY
//FT99F001 DD DISP=SHR,LABEL=(,,,IN),DSN=T201#.SESAME.Y8811.DATA
//** NLP PLOT FILE
//GDFILE DD SYSOUT=P
//PLOTLOG DD SYSOUT=X
//PLOTPRM DD DUMMY
//

```

图 4.18 AFD M实行用JCL例 (scalar)

```

//T201#AFD JOB (PWD),AFDM,MSGCLASS=X,
//  MSGLEVEL=(2,0),NOTIFY=T201#,
//  ATTR=(T4,C7,W3)
//*****
//*      AFD M Y8903 M21A VERSION      *
//*****
//AFDM EXEC PGM=M21A1V,PARM='FLIB(UNDERFL=0)'
//STEPLIB DD DSN=T201#.AFDM.LOAD,DISP=SHR
//SUBSYS DD SUBSYS=(VPCS)
//FT05F001 DD DISP=SHR,LABEL=(,,,IN),DSN=T201#.AFDM.DATA(LWEP)
//FT06F001 DD SYSOUT=X
//FT07F001 DD DUMMY
//FT08F001 DD DUMMY
//FT99F001 DD DISP=SHR,LABEL=(,,,IN),DSN=T201#.SESAME.Y8811.DATA
//** NLP PLOT FILE
//GDFILE DD SYSOUT=P
//PLOTLOG DD SYSOUT=X
//PLOTPRM DD DUMMY
//

```

图 4.19 AFD M实行用JCL例 (VP)

5. 結 論

既存の熱流体力学コードの大幅な精度向上と適用分野の拡大を達成するために開発されたコードAFDMのPNC大洗工学センターへの導入・整備を実施した。本作業でAFDMの各オプションが正常に機能することが確認され、この結果AFDMがPNCにおいて使用可能となった。

CRAY計算機用ソースプログラムのFACOM版への変換は、本来膨大な作業を必要とするが、このAFDMはHISTORIAN管理によってCRAY以外の計算機にもある程度対応できるよう開発されているため、変換作業はかなり軽減された。しかしなお、FACOM版としては修正の必要となる部分が多くあり、このFACOM版への変換作業について本報告書にまとめられている。今後の同様の変換作業あるいはAFDMの改訂等があった場合には、参考になるであろう。

CRAY版からFACOM版への変換によってサンプル問題のテストに若干の相違がみられるケースが現われたが、これはコンパイラの相違による計算順序及び浮動小数点内部表現形式の相違に基づくものと考えられる。

一方、この変換作業に伴って行った若干のFACOM-VP用効率化作業の結果、一層の計算時間の短縮が実現した。しかしながら計算アルゴリズムの変換等を含め、より詳細な検討を加えて更にFACOM-VP用に適した変換を行えば、より一層の計算時間の短縮が図られる可能性も残されている。

謝 辞

本報告書で述べたAFDMコードはLANL, KfK, PNC等の国際協力で開発されたものであり、開発に携わった各氏に先ず謝意を述べたい。特にLANLのW. R. Bohl氏, F. R. Parker 氏には、AFDMのPNCへの送付と共に、本研究に伴って発生した様々な疑問点に対して随時的確に対応していただいた。

一方、動燃大洗工学センター高速炉安全工学室の近藤悟氏からは本作業の全般に渡って様々な御協力、御助言を賜わった。さらに高速炉安全工学室の各氏には、AFDMオリジナルコードの動燃計算機システムへの移植に際し、御協力頂き、また、本作業に当って有益なコメントを頂いた。

さらに、FHLシステム五部三課の市川真一氏には、AFDMのVP機への導入に際し、貴重な御助言を賜わった。また、富士通エフ・アイ・ピー(株)科学技術システム部第4開発課の各氏からも多くのコメントを頂いた。

最後に、本作業の実施が可能となったのは、高速炉安全工学室の石川真氏（現在(株)日本原子力発電）の御尽力の賜物である。

ここに深く感謝致します。

参 考 文 献

1. L. L. Smith, "SIMMER-II: A Computer Program for LMFBR Disrupted Core Analysis", NUREG/CR-0453, LA-7515-M, Rev, (June 1980).
2. W. R. Bohl, et al., "AFDM" (being written).
3. 市川真一, "AFDM コードの最適化に関する研究", PNC SJ9067 87-001, (1987年10月).
4. EOS and Opacity Group, "SESAME '83: Report on the Los Alamos Equation-of-State Library", LALP-83-4, (February 1983).
5. OpCode, "HISTORIAN Plus User's Manual", OpCode, Inc. (April 1983).
6. CDC, "Update Reference Manual", Control Data Corporation Publications No. 60342500 (undated).
7. J. Abdallah, Jr., et al., "HYDSEBS: A Subroutine Package for Using Sesame in Hydrodynamic Codes", LA-8209, (June 1980).
8. F. R. Parker, "SIMMER Postprocessor Manual" Appendix D, NUREG/CR-4396, LA-10549-M, Manual, (October 1985).
9. 小寺慶一, "SIMMER-II/AFDM の図形処理コード T6Pの整備", PNC SJ 9072 88-001, (1988年3月).
10. ISSCO, "DISSPLA (Version 9.0) User's Manual" ISSCO, (1981).
11. 富士通, "SSL II 使用手引書", 富士通マニュアル 99SP-0050-5, (昭和55年12月)
12. 富士通, "FORTRAN 77/VP 使用手引書", 富士通マニュアル 78SP-5680-3, (昭和63年1月).
13. 富士通, "FORTUNE 使用手引書" 富士通マニュアル 78SP-5360-1, (昭和57年11月).
14. 富士通, " 会話型ベクトライザ使用手引書", 富士通マニュアル 78SP-5690-1, (昭和59年3月).

Appendix A SESAME ファイルユーティリティの導入

AFDMはEOS計算にEOSデータライブラリSESAMEを利用するオプションを持っている。LANLより送られてきたこのSESAME関連のファイルは、カードイメージのSESAMEデータファイルとSESAMEデータ関連サブルーチンパッケージHYDSESであった。ところでAFDMはSESAMEデータを使用する際、バイナリー・ファイルとして取扱う。このため、送られてきたカードイメージSESAMEファイルをバイナリーに変換しなければならない。そこで、サブルーチンパッケージHYDSESのうち、この変換ユーティリティであるUPDATEのFACOMへの変換を行った。この作業はAFDM同様HISTORIAN管理をすることとした。変換手順は以下のとおり。

- (1) サブルーチン群HYDSESのHISTORIANライブラリを作る。
- (2) UPDATEに対するメインプログラムをつくる（アッドファイルとする。）
- (3) PNC用のコレクションセットをつくる。内容は、
 - IBMオプションの追加（READ/WRITEの変換，文字型宣言）
 - バッファサイズの拡張
 である。

(4) SESAMEファイル変換

なお、当作業に関連するデータセットは以下。

- HYDSES HISTORIANライブラリ.....
 ' T 2 0 1 # . H Y D S E S . H I S L I B '
- PNC用 HYDSES アッドファイル.....
 ' T 2 0 1 # . H Y D S E S . P N C A F . F O R T '
- PNC用 HYDSES コレクションセット.....
 ' T 2 0 1 # . H Y D S E S . P N C C S . F O R T '

このうち、アッドファイルとコレクションセットの内容をそれぞれ図A.1, A.2に示す。また、UPDATEロードモジュール作成/実行JCLを図A.3に示す。この例では

```

カードイメージ SESAMEファイル .....
' T 2 0 1 # . S E S A M E . Y 8 8 1 1 . B C D . D A T A '
バイナリー SESAMEファイル .....
' T 2 0 1 # . S E S A M E . Y 8 8 1 1 . D A T A '
    
```

である。なお、この例ではHISTORIANステップでFORTRANソースを作成する際、UPDATE関連ルーチンのみを“*EDIT”指示子で指定している。このサンプルJCLは

```
' T 2 0 1 # . H Y D S E S . C N T L ( H C L G ) '
```

に格納されている。

```
*DECK SESAME
C
C ***** SESAME PNC ADDFILE FOR UPDATE *****
C                                     '88.05.15 BY KOD
C
CHARACTER*6 FMT /'(20A4)'/
DATA DATE /880516/
& ,NCPW /5/ ,INEW /0/ ,LUP /10/ ,LNEW/11/
& ,LP /6/
C
CALL UPDATE<DATE,NCPW,FMT,INEW,LUP,L1B,LM,LNEW,LP>
STOP
END
```

図A.1 PNC用HYDSESアッドファイル

```

*IDENT D85A
*/
*/      ***** PNC CORRECTION SET FOR SESAME *****
*/                      BY KOD : 88.05.15
*/
*/      ADD IBM VERSION
*/
*/I INBUF.30
*IF DEF,IBM,12
    INTEGER FM,TEST
    DATA FM/'*FM'/
C----- FIRST TEST FOR END-OF-FILE
    READ(LU,END=1) TEST
    IF(TEST.EQ.FM) GO TO 1
    BACKSPACE LU
C----- READ IN REQUESTED NUMBER OF WORDS
    READ(LU,END=1)(Z(I),I=1,N)
    IM = 1
    RETURN
C----- END-OF-FILE WAS READ, SET FLAG
    1 IM = 0
*IF -DEF,IBM,5
*I OUTBUF.25
*IF DEF,IBM,15
    INTEGER FM,TEST
    DIMENSION Z(1)
    DATA FM/'*FM'/
    IF(N.EQ.0) GO TO 1
C----- WRITE OUT REQUESTED NUMBER OF WORDS
    WRITE(LU) (Z(I),I=1,N)
C----- WRITE EOF IN CASE NO MORE WRITES TO -LU-
    WRITE(LU) FM
    WRITE(LU) FM
C----- BACKSPACE OVER EOF TO BE READU FOR MORE WRITES
    BACKSPACE LU
    BACKSPACE LU
    RETURN
C----- WRITE END-OF-FILE ONLY
    1 WRITE(LU) FM
*IF -DEF,IBM,5
*I RDCARD.35
*IF DEF,IBM,1
    CHARACTER*6 FMT
*/
*/      INCREASE BUFFER SIZE
*/
*/D COPMTS.25
    COMMON /SESCOM/ BUFF(33500)
*/D S2BLDM.38
    COMMON /SESCOM/ BUFF(33500)
*/D RETMAT.37
    COMMON /SESCOM/ BUFF(33500)
*/D S2SRM.28
    COMMON /SESCOM/ BUFF(33500)
*/D S2BLDL.28
    COMMON /SESCOM/ BUFF(33500) ,IBUFF(3000) ,BUFI(1000)
*/D S2UPDT.34
    COMMON /SESCOM/ BUFF(33500)

```

図A.2 PNC用HYDSESコレクションセット

```

//T201#S JOB (KOD),SESAME,MSGCLASS=X,
// MSGLEVEL=(2,0),NOTIFY=T201#,
// ATTR=(TO,C3,W3)
/*****
/** HYDSES PNC VERSION *
/** HISTORIAN/COMPILE/LINK *
/*****
//HISTOR EXEC PGM=HISTOR
//STEPLIB DD DISP=SHR,DSN=SYS9.HISTOR.LOAD
//FT05F001 DD *
HISTORIAN(P,T,C)
*LINE,78
*READ,9
*READ,10
*DEFINE IBM
*EDIT,OUTBUF,INBUF,TBFCH,RDCARD,SKIPMF,COPMTS.UPSUM,SESAME
/*
//FT06F001 DD SYSOUT=X
//FT09F001 DD DISP=SHR,DSN=T201#.HYDSES.PNCCS.FORT
//FT10F001 DD DISP=SHR,DSN=T201#.HYDSES.PNCAF.FORT
//FT21F001 DD DISP=SHR,DSN=T201#.HYDSES.HISLIB,LABEL=(,,IN)
//FT22F001 DD UNIT=WORK,SPACE=(4080,(1000,50)),DISP=NEW
//FT23F001 DD DISP=(,PASS),DSN=&&SSOR,UNIT=WORK,
// SPACE=(TRK,(50,10),RLSE),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//FT25F001 DD UNIT=WORK,SPACE=(4080,(500,50)),DISP=NEW
/*T26F001 DD DUMMY
//FT27F001 DD UNIT=WORK,SPACE=(4080,(500,50)),DISP=NEW
/***** STEP FORT =====
//FORT EXEC PGM=JZK0FORT,COND=EVEN,
// PARM='LC(78),OPT(3)'
//SYSPRINT DD SYSOUT=X
//SYSUT1 DD UNIT=SYSSQ,SPACE=(3465,(3,3)),DCB=BLKSIZE=3465
//SYSUT2 DD UNIT=SYSSQ,SPACE=(5048,(10,10))
//SYSPUNCH DD DUMMY
//SYSLIN DD DISP=(,PASS),DSN=&&SOBJ,UNIT=WORK,
// SPACE=(TRK,(50,10),RLSE),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//SYSIN DD DISP=(OLD,DELETE),DSN=&&SSOR
/**
/***** STEP LINK. =====
//LKED EXEC PGM=JQAL,REGION=500K,TIME=8,COND=(12,LT,FORT),
// PARM='LET,LIST,LINECOUNT(78)'
//SYSLIB DD DISP=SHR,DSN=SYS1.FORTLIB
//SYSUT1 DD UNIT=SYSSQ,SPACE=(1024,(200,20))
//SYSPRINT DD SYSOUT=X
//SYSLIN DD DISP=(OLD,DELETE),DSN=&&SOBJ
//SYSLMOD DD DSN=&&LOAD(UPDATE),DISP=(NEW,PASS),
// SPACE=(TRK,(30,10,1),RLSE),UNIT=WORK
/***** STEP GO =====
//UPDATE EXEC PGM=UPDATE
//STEPLIB DD DSN=&&LOAD,DISP=(OLD,DELETE)
//FT06F001 DD SYSOUT=X
//FT10F001 DD DSN=T201#.SESAME.Y8811.BCD.DATA,DISP=SHR,LABEL=(,,IN)
//FT11F001 DD DSN=T201#.SESAME.Y8811.DATA,DISP=(,CATLG),
// SPACE=(TRK,(10,10)),UNIT=DASD
//

```

図A.3 UPDATE作成/実行JCL

Appendix B PULライブラリの整備

AFDMは図形処理ルーチンとしてLANLのPUL (Plot Utility Library) を使用している。このPULはSIMMER後処理コード、T6Pに組み込まれていたものである。そこでAFDM導入に際してPULをT6Pから独立させてライブラリ形式にして再整備を行った。基本的には、LANLより送られていたPULのHISTORIANソースからHISTORIANライブラリを作成し、これに対してPNC用の修正(T6P導入の際の修正に対する若干の補正)を行い、ライブラリ形式のロードモジュールを作成した。

その後、1988年11月、AFDM Y8811版と共にPULの改訂版も送られてきた。このPUL改訂版は、前版に加えて以下のオプションが追加されていた。

(1) INTMASK ON ビット操作関数として、整数関数IAND, IOR, INOT
を使用。

OFF ビット操作関数として、実数関数AND, OR, COMPL
を使用。

(2) FOURB ON 1バイト8ビット, 1語4バイトマシン用

OFF 1バイト8ビット, 1語8バイトマシン用

(3) CGS ON LANL CGS システム用

(4) NCAR ON NCAR SCD 図形システム用

(5) DISP ON ISSCO DISPLA用

このため、INTMASK, FOURB, DISPを立てればPNC用の、PULソースプログラムができることとなった。しかしなお少々の修正が必要であり、この修正コレクションセットを

'T201#. PUL. PNCCS. FORT'

に格納した。送られてきたPUL改訂版HISTORIANソースからHISTORIANライブラリを作成し、

'T201#. PUL. HISLIB'

に格納した。これらを用いてPULライブラリを作成し、

'T201#. PUL'

に格納した。これらを用いてPUL作成JCLは、

'T201#. PUL. CNTL (HCL)'

にある。以上のPNC用コレクションセットを図B.1に、作成JCLを図B.2にそれぞれ示す。

```
*ID D8CA
*/
*/      ***** PNC CORRECTION SET FOR PLOT UTILITY LIBRARY *****
*/                        BY KOD: 88.12.02
*/
*/      CORRECT SYNTAX ERROR IN ORIGINAL PUL
*/
*/D PCON.264
*/      FE(IU+1,J)=VDT(2)
*/D DRV.9
*/      CALL VECTOR(XFR,YFR,XTO,YTO,0)
*/D WLC.12
*/      CALL RESET('ANGLE')
*/
*/I TXTFLM.25
*/IF DEF,DISP,1
*/      COMMON /AREA/ XAXIS, YAXIS
*/D TXTFLM.58
*/      CALL HEIGHT(.085)
*/D TXTFLM.60
*/      CALL MESSAG(BUF(L),NCW,0.0,(1.0-FLOAT(LPOS)/64.0)*YAXIS)
*/
*/      ADD AREA2D IN ADV
*/
*/I ADV.2
*/IF DEF,DISP,1
*/      COMMON /AREA/ XAXIS, YAXIS
*/D ADV.5,6
*/IF DEF,DISP,2
*/      CALL ENDPL(0)
*/      CALL AREA2D(XAXIS,YAXIS)
```

☒ B.1 PNC用PULコレクションセット

```

//T201#PUL JOB (PWD),PUL,MSGCLASS=X,
//  MSGLEVEL=(2,0),NOTIFY=T201#,
//  ATTR=(T1,C2,W3)
//*****
//*      PUL Y8811 PNC VERSION          *
//*      HISTORIAN/COMPILE/LINK        *
//*****
//HISTOR EXEC PGM=HISTOR
//STEPLIB DD DISP=SHR,DSN=SYS9.HISTOR.LOAD
//FT05F001 DD *
HISTORIAN(P,T,F)
*READ 9
*LINEs 78
*DEFINE DISP
*DEFINE INTMASK
*DEFINE FOURB
/*
//FT06F001 DD SYSOUT=X
//FT09F001 DD DISP=SHR,DSN=T201#.PUL.PNCCS.FORT
//FT21F001 DD DISP=SHR,DSN=T201#.PUL.HISLIB
//FT22F001 DD UNIT=WORK,SPACE=(4080,(1000,50)),DISP=NEW
//FT23F001 DD DISP=(,PASS),DSN=&&SSOR,UNIT=WORK,
//  SPACE=(TRK,(50,10),RLSE),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//FT25F001 DD UNIT=WORK,SPACE=(4080,(500,50)),DISP=NEW
//FT27F001 DD UNIT=WORK,SPACE=(4080,(500,50)),DISP=NEW
//***** STEP FORT/VP =====
//FORT EXEC PGM=JZK@FORT,COND=EVEN,
//  PARM='LC(78),OPT(3)'
//SYSPRINT DD SYSOUT=X
//SYSUT1 DD UNIT=SYSSQ,SPACE=(3465,(3,3)),DCB=BLKSIZE=3465
//SYSUT2 DD UNIT=SYSSQ,SPACE=(5048,(10,10))
//SYSPUNCH DD DUMMY
//SYSLIN DD DISP=(,PASS),DSN=&&SOBJ,UNIT=WORK,
//  SPACE=(TRK,(50,10),RLSE),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//SYSIN DD DISP=(OLD,DELETE),DSN=&&SSOR
//
//***** STEP LINK. =====
//LKED EXEC PGM=JQAL,REGION=500K,TIME=8,COND=(12,LT,FORT),
//  PARM='LET,LIST,LINECOUNT(78),ALIAS,NCAL'
//SYSLIB DD DISP=SHR,DSN=SYS1.FORTLIB
//  DD DISP=SHR,DSN=SYS9.DISLIB
//  DD DISP=SHR,DSN=SYS9.INTLIB
//SYSUT1 DD UNIT=SYSSQ,SPACE=(1024,(200,20))
//SYSPRINT DD SYSOUT=X
//SYSLIN DD DISP=(OLD,DELETE),DSN=&&SOBJ
//SYSLMOD DD DSN=T201#.PULIB(DISPLA),DISP=SHR
//

```

图 B.2 PUL 作成 JCL