

球形管板の3次元構造解析法の開発

(研究報告)

2003年12月

核燃料サイクル開発機構
大洗工学センター

本資料の全部または一部を複写・複製・転載する場合は、下記にお問い合わせください。

〒319-1184 茨城県那珂郡東海村村松4番地49

核燃料サイクル開発機構

技術展開部 技術協力課

電話：029-282-1122（代表）

ファックス：029-282-7980

電子メール：jserv@jnc.go.jp

Inquiries about copyright and reproduction should be addressed to :

Technical Cooperation Section,

Technology Management Division,

Japan Nuclear Cycle Development Institute

4-49 Muramatsu, Tokai-mura, Naka-gun, Ibaraki-Ken 319-1184, Japan

© 核燃料サイクル開発機構 (Japan Nuclear Cycle Development Institute)
2003

球形管板の3次元構造解析法の開発

高正 英樹*¹⁾、笠原 直人*²⁾、柄谷 和輝*³⁾

要 旨

実用高速炉では、経済性の要求から冷却ループの大容量化とループ数削減を指向しており、熱交換器が大型化する方向である。また、冷却材に液体金属ナトリウムを使用することから信頼性の高い2重伝熱管の採用が計画されており、その結果管溶接の不要な直管型となることから、伝熱管本数が増大する傾向にある。以上から蒸気発生器は大型の2重直管型となり、耐圧設計との兼ね合いから大型球形管板が必要になる。大型球形管板を実用化するには、複雑3次元形状に対する構造解析法の確立が必要となる。

従来の管板解析法は、多孔部を等価中実板に置き換えて平均的な応力を求め、それに集中係数を乗じることで孔縁のピーク応力を求めるものであった。球形管板では、最大応力が生じる外縁の孔廻りの異方性が強いことから、応力集中係数による評価精度が低下する。このため、孔廻りの詳細形状を考慮した3次元解析が望まれる。3次元解析は、外縁の孔廻りに限定した部分孔開きモデルとしても大規模になるため、通常の直接法ソルバでは計算が難しくなる。このため、大規模解析で計算効率の高くなる反復ソルバを開発し、FINASへ実装した。

実用化戦略調査研究で検討されている直管型球形管板を対象として、部分孔開きモデルを作成したところ100万自由度を越す大きさとなった。反復ソルバを組み込んだFINASとPCを用いて手動トリップを想定した20ステップの熱過渡弾性応力解析を行ったところ、72時間で解析が終了した。以上から球形管板に対する3次元構造解析法の適用性が確認できた。

*1) 常陽産業株式会社

*2) 大洗工学センター 要素技術開発部 構造信頼性研究グループ

*3) (株)CRCソリューションズ

The development of three-dimensional structure analysis method of the dished tubesheets.

Hideki TAKASHO^{*1)}, Naoto KASAHARA^{*2)} and Kazuteru GARATANI^{*3)}

Abstract

It is intend to adopt large scale heat exchangers in the design study of commercial fast reactors, since capacity enlargement and reduction of the cooling loops are pursued from the demand of economical competitiveness. Adoption of double-wall heat transfer tubes for reliable design for sodium coolant is also planed, in which straight type tubes without welding are used. As the result, steam generators change to double-wall straight tube types and the large-scale semi-spherical tubesheets are required from a balance with the pressure proof design. In order to realize the large-scale semi-spherical tubesheets, an analysis method for the complicated three-dimensional structures is required.

Conventional analysis models for tubesheets are equivalent solid plates with stress multipliers, which are difficult to be applied to semi-spherical tubesheets since their anisotropy. A large-scale three-dimensional analysis model is desired to take into account the effects of detailed configurations around holes. An iterative solver for efficient calculating was developed, and it was implemented to FINAS code.

A semi-spherical tubesheet model for commercial fast reactors was made with three-dimensional element, which exceeded one million degrees of freedom. FINAS code with the iterative solver analyzed thermal transient stress under a manual trip condition. 20 step calculation was carried out within 72 hours on a PC. As the result, it was confirmed that applicability of three dimensional analysis methods to semi-spherical tubesheets

*1) Joyo Industries Co. Ltd.

*2) Structure and Material Research Group, Advanced Technology Division, OEC, JNC

*3) CRC Solutions Co. Ltd

目 次

1. 緒言.....	1
2. 大規模有限要素法解析の開発.....	7
2.1 概要.....	7
2.2 反復法について.....	9
2.2.1 直接法と反復法.....	9
2.2.2 反復法の理論.....	10
2.2.3 解法の選択.....	11
2.2.3.1 共役勾配法 (CG 法).....	12
2.2.3.2 前処理付き共役勾配法.....	13
2.2.4 前処理.....	13
2.2.4.1 ポイントヤコビ.....	14
2.2.4.2 不完全コレスキー分解.....	15
2.3 FINAS への反復法組込みの設計.....	16
2.3.1 反復法の組込み方針.....	16
2.3.2 反復法の組込み方法.....	17
2.3.3 FINAS の入力データ.....	18
2.3.4 反復法組込み手順.....	18
2.4 FINAS への反復法組込み作業.....	20
2.4.1 反復法組込み作業の概要.....	20
2.4.2 SOLVE での係数行列保存.....	20
2.4.2.1 SDCOMPI での操作.....	21
2.4.3 FBS への反復法本体の組込み.....	23
2.4.3.1 FBSI での操作.....	24
2.4.4 反復ソルバ用変数領域の追加.....	24
2.4.5 共通領域 (COMMON) の追加.....	27
3. 大規模3次元有限要素解析への適用評価.....	28
3.1 反復法の性能評価-1.....	28
3.1.1 単位立方格子の弾性解析による性能評価.....	28
3.1.1.1 単位立方格子問題での記憶容量、演算量の推定.....	28
3.1.1.2 より一般的な場合での演算量の推定.....	30

3.1.1.3 FINAS による性能評価.....	31
3.1.1.4 大規模問題での留意点.....	34
3.1.1.5 参考：CPU による差.....	35
3.1.2 FINAS 実証例題 DP-14, DP-25 による性能評価.....	36
3.2 反復法の性能評価-2.....	38
3.2.1 球形管板中規模モデルによる解析.....	38
4. 直管型球形管板の3次元有限要素解析.....	44
4.1 概要.....	44
4.2 計算方法.....	44
4.2.1 熱伝導解析.....	44
4.2.2 熱応力解析.....	45
4.3 解析結果.....	45
4.3.1 熱伝導解析結果.....	45
4.3.2 熱応力解析結果.....	45
4.4 計算速度.....	46
5. 結言.....	66
6. 参考文献.....	67

図リスト

図 1.1	12Cr Steel Curved Tubesheet Structure	1
図 1.2	Flat tubesheet structure (Monju SG).....	3
図 1.3	Simplified analysis method of flat tubesheet.....	3
図 1.4	Simplified analysis method of dished tubesheet.....	4
図 1.5	Thermal transient strength test results of tubesheet.....	5
図 1.6	Revealed strain concentration mechanism around holes.....	6
図 2.1	反復法選択のお勧めを示す流れ図.....	11
図 2.2	反復ソルバ追加の概念図	17
図 2.3	三角分解過程：TRIA への組込み	23
図 2.4	前進後退代入過程：FBSI への組込み	24
図 3.1	単位立方格子モデル	28
図 3.2	分割方法	31
図 3.3	全 CPU 時間の比較	33
図 3.4	ソルバでの処理時間の比較	33
図 3.5	FINAS 実証例題.....	36
図 3.6	球形管板中規模モデル 全体形状.....	41
図 3.7	球形管板中規模モデル 管部詳細.....	41
図 3.8	球形管板中規模モデル 熱伝導解析結果 (BCS-Metis, JNC による)	42
図 3.9	球形管板中規模モデル 熱伝導解析結果 (ポイントヤコビ-CG 法)	42
図 3.10	球形管板中規模モデル 熱応力解析結果 (BCS-Metis)	43
図 3.11	球形管板中規模モデル 熱応力解析結果 (ポイントヤコビ-CG 法)	43
図 4.1	直管型 SG (全体図)	48
図 4.2	直管型 SG (出口側管板)	49
図 4.3	直管型 SG 球形管板メッシュ図 (全体)	49
図 4.4	直管型 SG 球形管板メッシュ図 (最外縁部詳細)	50
図 4.5	直管型 SG 球形管板メッシュ図 (中央部詳細)	50
図 4.6	SG 出口蒸気温度履歴 (手動トリップ)	51
図 4.7	等価物性領域の内部発熱量算出法.....	51
図 4.8	負荷条件	52
図 4.9	拘束条件	52
図 4.10	リガメント効率による等価物性値 (ASME Sec III A-8000)	53
図 4.11	温度分布 (t=0 秒)	53
図 4.12	温度分布 (t=30 秒)	54
図 4.13	温度分布 (t=120 秒)	54

図 4.1.4	温度分布 (t=200 秒)	55
図 4.1.5	温度分布 (t=280 秒)	55
図 4.1.6	温度分布 (t=500 秒)	56
図 4.1.7	相当応力分布 (内圧負荷のみ、t=0 秒)	56
図 4.1.8	相当応力分布 (最大応力発生時点、t=400 秒)	57
図 4.1.9	外縁部の相当応力分布 (1) (最大応力発生時点、t=400 秒)	57
図 4.2.0	外縁部の相当応力分布 (2) (最大応力発生時点、t=400 秒)	58
図 4.2.1	中央部の相当応力分布 (1) (最大応力発生時点、t=400 秒)	58
図 4.2.2	中央部の相当応力分布 (2) (最大応力発生時点、t=400 秒)	59
図 4.2.3	相当応力分布 (内圧負荷のみ、t=0 秒)	59
図 4.2.4	相当応力分布 (t=30 秒)	60
図 4.2.5	相当応力分布 (t=120 秒)	60
図 4.2.6	相当応力分布 (t=200 秒)	61
図 4.2.7	相当応力分布 (t=280 秒)	61
図 4.2.8	相当応力分布 (t=500 秒)	62
図 4.2.9	球形管板大規模モデルの熱伝導解析 (最終ステップの温度分布、BCS-Metis)	63
図 4.3.0	球形管板大規模モデルの熱伝導解析 (最終ステップの温度分布、点ヤコビ-CG 法)	63

表リスト

表 2.1	反復ソルバ用の変数領域 (1)	26
表 2.2	反復ソルバ用の変数領域 (2)	27
表 3.1	1次元、2次元、3次元的な分割とメモリ・演算量が比例する基数	31
表 3.2	反復ソルバでの処理時間 (対角スケーリング、CG法)	32
表 3.3	直接ソルバでの処理時間 (マルチフロントアル法、MMDリナンバ)	32
表 3.4	直接ソルバでの処理時間 (FINAS固有ソルバ)	32
表 3.5	反復ソルバでの処理時間	35
表 3.6	DP-14での処理時間 (自由度数: 5,815)	37
表 3.7	DP-25での処理時間 (自由度数: 2,839)	37
表 3.8	球形管板中規模モデルの概要	39
表 3.9	球形管板中規模モデル熱伝導解析 (自由度数: 41,164、1ステップ)	40
表 3.10	球形管板中規模モデル応力解析 (自由度数: 123,492、1ステップ)	40
表 4.1	直管型球形管板モデルの概要	64
表 4.2	熱伝導解析の物性値	64
表 4.3	熱伝導解析に用いた熱伝達係数	64
表 4.4	熱応力解析で用いた材料物性値	64
表 4.5	球形管板大規模モデル熱伝導解析の処理時間 (自由度数: 342,830、19ステップ)	64
表 4.6	球形管板大規模モデル熱伝導解析での反復回数	65
表 4.7	球形管板大規模モデル熱応力解析の処理時間 (自由度数: 1,028,490、20ステップ)	65

1. 緒言

実用高速炉では、経済性の要求から冷却ループの大容量化とループ数削減を指向しており、熱交換器が大型化する方向である[1]。また、冷却材に液体金属ナトリウムを使用することから信頼性の高い2重伝熱管の採用が計画されており、その結果管溶接の不要な直管型となることから、伝熱管本数が増大する傾向にある。

さらに、実用高速炉は出力に対する冷却材インベントリが実証炉より小さく熱過渡条件が厳しくなることから、材料に耐熱特性に優れ良熱伝導の12Cr鋼の採用を計画している。以上から蒸気発生器は図1.1に示すような大型の2重直管型となり、耐圧設計との兼ね合いから12Cr大型球形管板が必要になる。12Cr系大型球形管板を実用化するには、複雑3次元形状に対する構造解析法と、12Cr系大型鍛鋼品の熱過渡強度評価法の確立が必要になる。

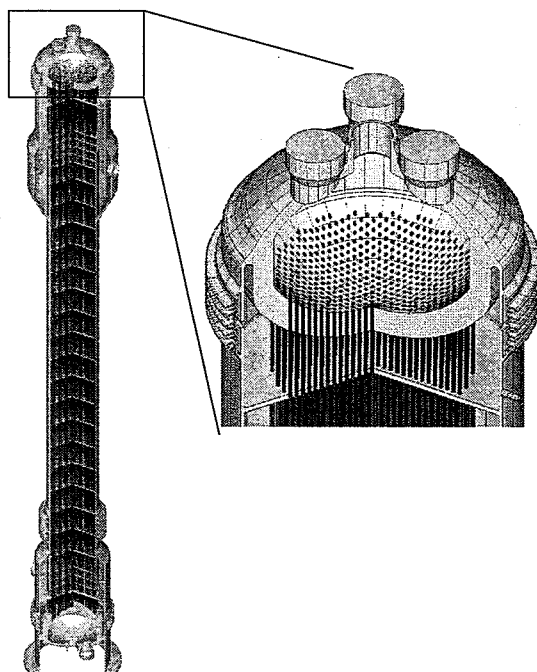


図 1.1 12Cr Steel Curved Tubesheet Structure

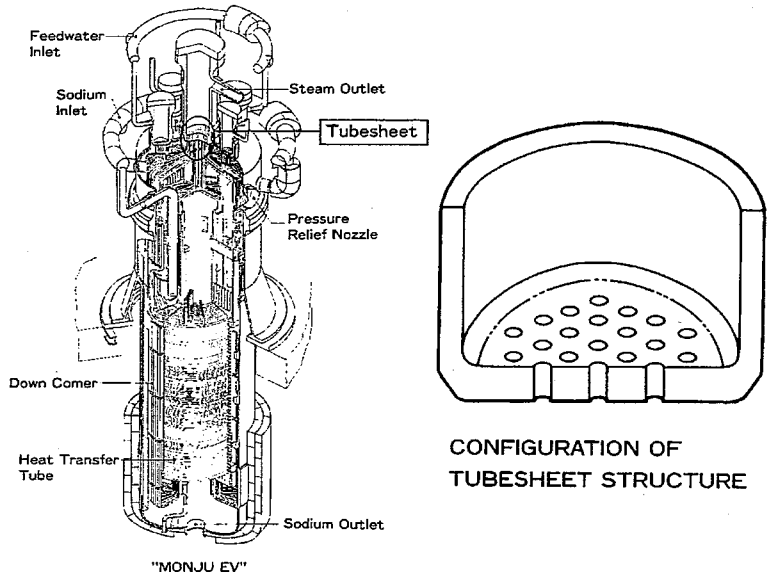
管構構造は従来の平管板であっても、図 1.2 に示すように多孔領域とそれを囲むリムおよび支持構造物とからなる複雑な 3 次元構造であるため、そのままの形状で解析することは現実的でなく設計評価には専用の簡易解析法が必要とされてきた。

軽水炉の分野では管板の簡易解析法として、ASME Boiler & Pressure Vessel Code Sec. III 中の Appendix A-8000 [2] が基本的に参照されている。本方法は O' Donnell らにより提案されたもので、図 1.3 に示すように解析の簡易化のため多孔領域を等価な中実板でモデル化し、等価中実板の解析から得られる公称応力 (nominal stress) に応力集中係数を乗じることで孔回りの局所的応力を評価する弾性領域における解析法 [3] である。

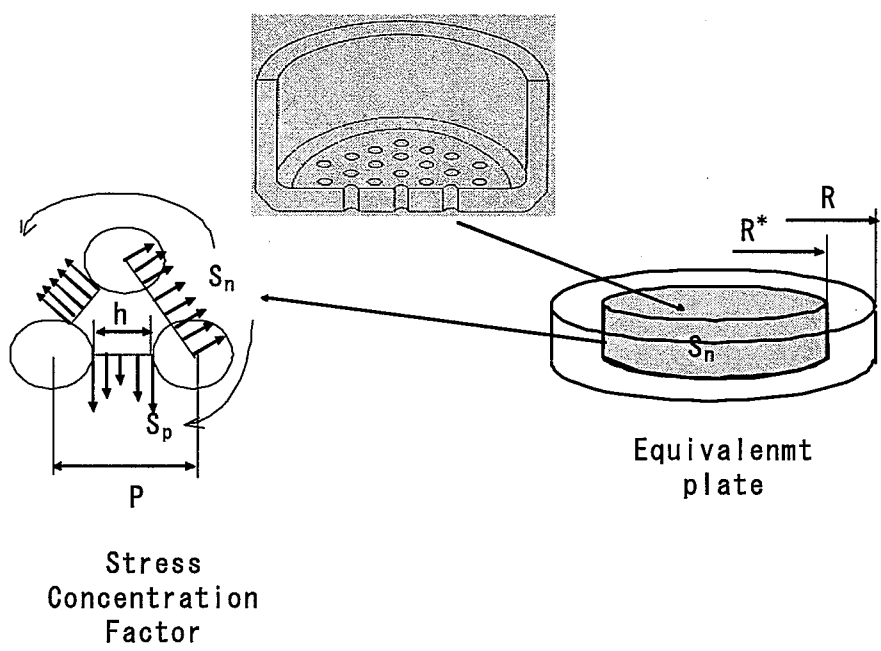
これに対し、材料の弾塑性クリープ挙動が問題となる高温で運転される高速炉の管板を解析するには、非弾性解析法が必要となる。このためサイクル機構が中心となり、弾塑性解析用の等価物性値と孔周りの弾塑性クリープひずみ集中係数とからなる高温設計用の簡易解析法 [4] [5] [6] が提案された。

A-8000 と同様のアプローチを球形管板に適用しようとする、2 つの問題が生じる。等価中実板に関して、球形管板に対して伝熱管が面外方向に垂直 (球中心から放射状) に差し込まれる場合は、中実板はほぼ等方均一と考えられるが、図 1.4 のように伝熱管が平行に差し込まれた場合は、管板の周辺部が非均一で異方性を持つことになる。孔周り集中係数に関しても、孔形状が楕円になることから特有の値を持つことになる。

以上から、球形管板に対しては専用の簡易解析法を開発するか、または FEM による大規模 3 次元解析法を実用化する必要がある。



☒ 1.2 Flat tubesheet structure (Monju SG)



☒ 1.3 Simplified analysis method of flat tubesheet

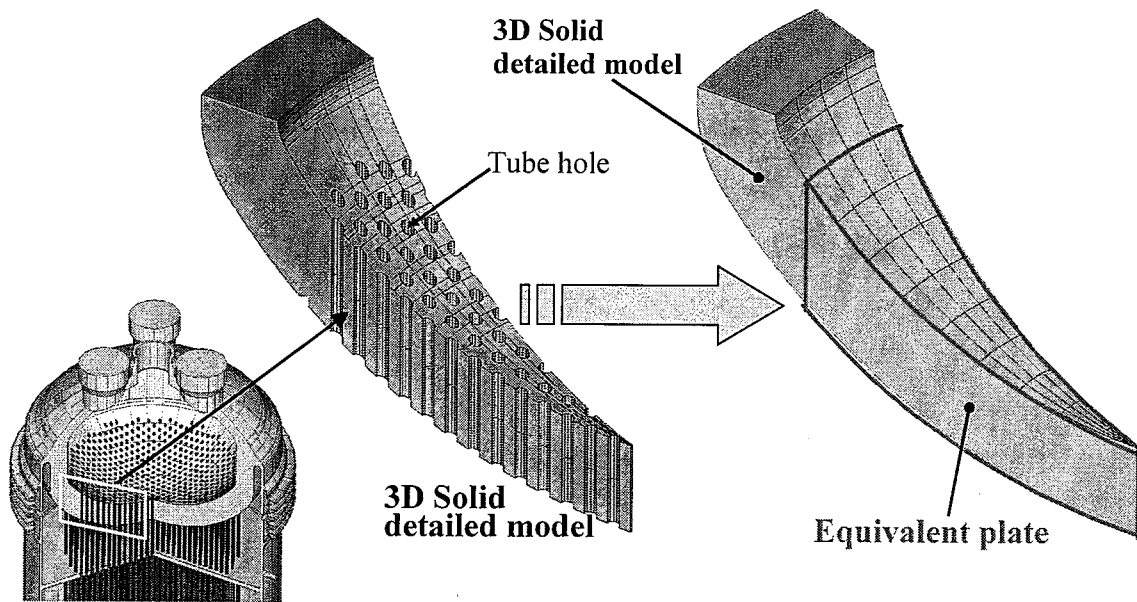


図 1.4 Simplified analysis method of dished tubesheet

原子力プラントの構造設計基準は破損様式を意識した「解析による設計」を基本方針としているため、管板構造に関しても防止すべき破損モードを把握することが設計法開発の上で重要となる。これに対し、管板の多孔部は形状が複雑なため解析的に破損形態を予測することは難しい。

このため実証炉設計において稠密管板を採用する際に、その破損モード把握を目的として、SUS304 鋼多孔板構造を盛り込んだ供試体を製作し、構造物強度確性試験施設 (TTS) を用いて実機を上回る厳しい過渡熱応力を繰り返し加えることにより強度試験を実施した。その結果、多孔板リガメント部の熱応力による破損データを得ることに成功し [7]、図 1.5 に示すように多孔板には孔縁を起点とした微細なき裂と、孔間の最小断面を貫通するき裂の、2つの破損形態が存在することを確認した。この結果に基づき、図 1.6 に示すように孔間距離と材料の応力-ひずみ特性に依存して変化する応力集中メカニズムを解明し、それを考慮した設計法を開発した [5]。

以上の経験から、形状の特異な球形管板に関しては、新たな破損モードの有無を確認する必要があると考えられる。

さらに 12Cr 鋼は耐圧設計を主体とする火力機器用に開発されてきたが、球形管板では繰り返し熱応力に対する破損防止を図る必要がある。12Cr 鋼はステンレス鋼に比較

して降伏応力が高いことから、孔周りのひずみ集中形態がピークひずみ集中型になる可能性がある。また、延性が低いことから孔間貫通型の膜ひずみ集中に対する許容値は低下する。特に、厚肉鍛鋼品から製作される管板ではその傾向が顕著と考えられている。

以上から、12Cr鋼管板の破損形態はステンレス鋼と異なる可能性が考えられ、耐熱応力設計法を確立するためには破損様式と限界強度の把握が必要となる。

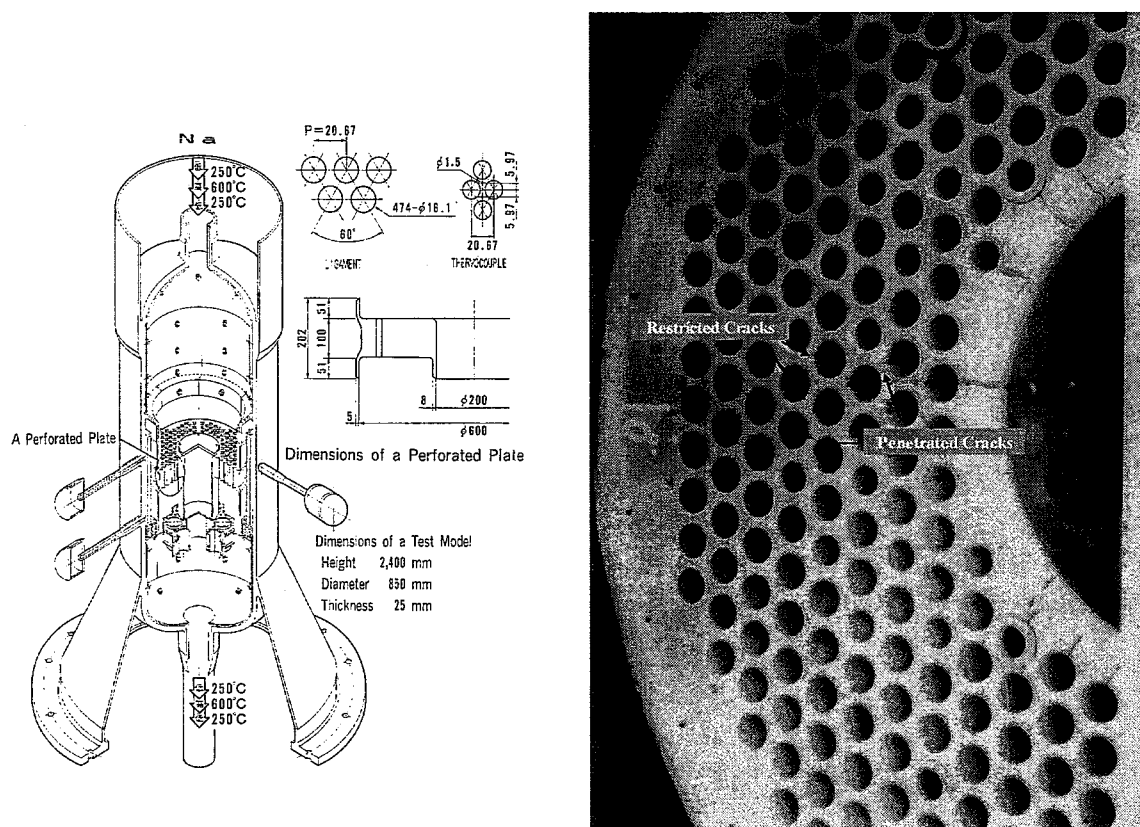


図 1.5 Thermal transient strength test results of tubesheet

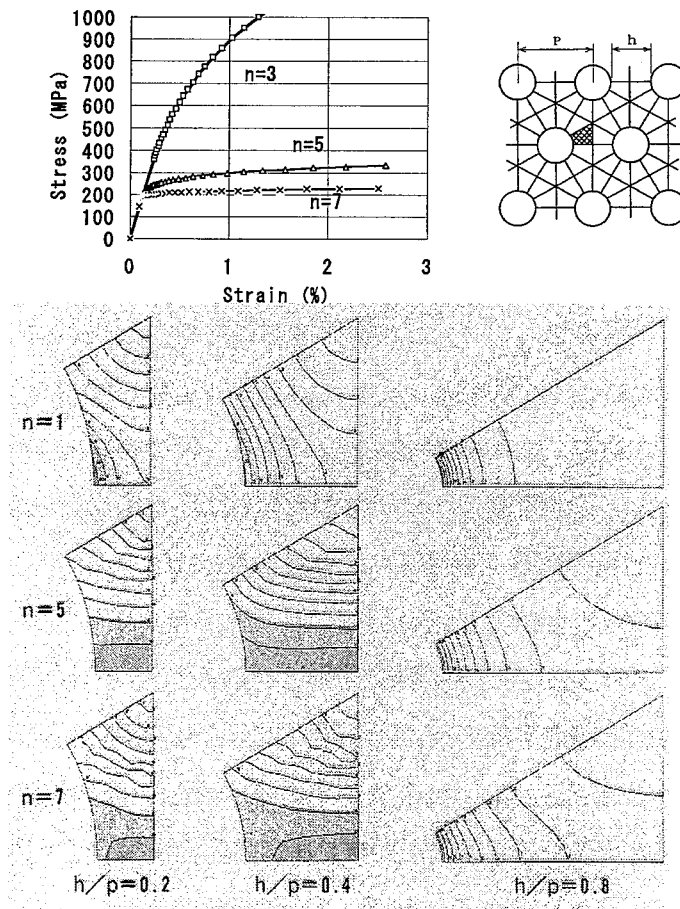


図 1.6 Revealed strain concentration mechanism around holes

これまで説明してきた背景と理由から、12Cr 鋼大型球形管板の構造健全性評価手法の確立を目的として、3次元構造解析法と熱過渡強度評価法の開発を進める。3次元構造解析法については、実用化戦略調査研究フェーズ2（～H17）において、成立性を見通すために最低限必要な弾性解析法（簡易解析法と3次元FEM解析法）を開発する。本報告書では、3次元FEM解析法の開発状況について説明する。

2. 大規模有限要素法解析の開発

2.1 概要

球形管板を精度良く解析するには、孔廻りの詳細形状を考慮した3次元解析が望まれる。孔廻りの詳細形状を考慮した3次元解析は、対象を外縁の孔廻りに限定した部分孔開きモデルとしても大規模なものになる。計算機の処理能力が飛躍的に向上したとはいえ、このような3次元モデルに対するシミュレーションは容易ではなく、現状では規模の大きさに関して、物理的、経済的な限界が存在する。特に有限要素法を用いた計算機シミュレーションで大規模解析を実施する場合には、利用可能な記憶装置あるいは補助記憶装置の容量、または演算時間が問題の規模を制限する。

この問題を解決する方法として並列計算機の利用も始まっているが、まだ実用的なレベルで利用できるソフトウェアの整備がなされていないこと、入力データや出力結果の処理についても並列化を考慮する必要があるために、これらが容易に試みられるほど並列計算が普及しているとは言えないのが実情である。また、近年は安価なPCがかつての大型計算機と同等な性能を示し、PC クラスタ等の並列計算機が出始めている。しかし、小規模な並列化を試みる程度であれば、たとえPCであっても複数台の購入により結局は高価となるクラスタ型計算機を導入する利点はそれほど見出せない。

一方、従来、有限要素法による熱あるいは応力解析では、その演算量の大部分を占める連立一次方程式の解法（ソルバ）として直接法が用いられてきた。直接法は、常に安定して解が得られ、有限要素法で解くべきスパース・マトリックス（疎行列）においては、数万～数十万自由度までは高速である。しかし、数十万自由度以上の大規模問題になると、直接法は反復法に較べて演算時間が急激に増加するために、反復法が有利になってくる。この他にも反復法は、スパース・マトリックスの場合、係数行列の非零項のみを記憶するために記憶容量の点で有利である。このような理由で、一般に小～中規模問題スパース・マトリックスでは直接法が有利であるが、数十万自由度を超える大規模問題では反復法が有利となる。また、反復法の欠点である安定性に関して、近年、多くの研究がその前処理について実施されており、適切な前処理を施すことによって、従来、反復法の適用が難しいといわれていた分野でも、安定した解が得られるようになってきた。

汎用非線形有限要素法解析システム FINAS の大規模問題に対する適用性を向上させ

るために、連立一次方程式の解法として、直接法の他に新たに反復法の組込みを行い、大規模問題についての実用的な解析を実施できるようにする。

(1) FINAS への反復法組込み

(2) 反復法の性能の評価

(1) FINAS への反復法組込み

FINAS への反復法組込みでは、FINAS の静的解析・熱伝導解析機能において反復法が適用できるように機能の拡張を行う。また、反復法の手法としては、静的解析・熱伝導解析で現れる正定値対称マトリックスにおいて、最も性能のよい共役勾配法（CG 法）とし、さらに収束性を向上させるため、ポイントヤコビ法等の前処理を試行する。

(2) 反復法の性能の評価

反復法の性能の評価では、FINAS に組込んだ反復法により、球形管板の熱伝導解析と、これによって得られた温度分布を用いた静的熱応力解析を実施する。また、一方では、これまでの直接法による解析も実施することによって、結果の妥当性を検証し、反復ソルバの性能を評価する。

2.2 反復法について

2.2.1 直接法と反復法

自然現象あるいは工学現象を記述する偏微分方程式を、有限要素法によって離散化し、数値的に解を得る場合、最終的に以下のような連立一次方程式を解くことになる。

$$Ax = b \quad (2-1)$$

このとき、問題の規模が大きくなればなるほど、有限要素法の全計算時間に占める連立一次方程式を解くための時間が大きくなる。よって、より安定で効率の良い連立一次方程式の解法（以下ソルバと呼ぶ）を使うことができれば、解析の経済性を改善することができる。

従来、有限要素法、特に構造解析の分野では、ソルバとして直接法が用いられてきた。直接法はガウスの消去法に基づくもので、有限要素法ではマトリックスが大規模で係数項が疎であること（スパース性）を利用し、バンド法、スカイライン法、ウェーブフロント法、マルチフロンタル法が用いられる。直接法は、演算回数が既知であり、必ず正解が得られる（安定している）ため、信頼性の高いソルバとして、この分野で長い実績と不動の地位を保ってきた。

これに対して反復法は、 x の近似ベクトルを反復計算により正解に近づけていくもので、大規模問題では、直接法より記憶容量と演算量に関して直接法より有利とされている（3.1.1.1 節参照）。しかし、収束計算によって解を得るため、理論上は必ず正解が得られるという保証はあっても、実際には収束しない場合もある。この点から、それほど問題規模が大きくならない構造解析の分野で反復法はあまり利用されることが無かった。

一方、近年の大幅な計算機性能の向上と低価格化は、有限要素法で扱う問題の規模を常に大きくし続けている。このような環境の基、様々な分野で大規模問題に有効なソルバとしての反復法に期待が集まっており、収束性を改善させる前処置手法の研究が盛んである。また、反復法の利点は、直接法の消去過程や前進/後退代入過程で現れるマトリックス全体への依存性が少ないために、並列計算に有利なことも挙げられる。よって、今後さらに計算機の処理能力が向上するなら、構造解析の分野でも反復法が主流になる

ことも考えられる。

2.2.2 反復法の理論

反復法とは、連立一次方程式の未知変数ベクトルに対する近似ベクトルを仮定し、これを逐次的に真の解に反復収束させる手法である。反復法には様々な手法が提案されているが、これらは定常的手法 (stationary) と非定常的手法 (non stationary) という2種類に分類できる。

例えば、連立一次方程式 $Ax = b$ を以下の形に変形し、

$$Ax = Sx + g \quad (2-2)$$

反復法では、初期ベクトル x_0 から出発して以下の漸化式により反復的に解く。

$$x_{k+1} = Sx_k + g \quad (2-3)$$

このとき、上式が、以下のように書ける場合、この反復法を定常的手法と呼ぶ。

$$x_{k+1} = x_k + H(b - Ax_k) = x_k + Hr_k \quad (2-4)$$

この方法は、既に得られている x_k の値を右辺に代入すれば x_{k+1} が決まるので、陽解法とも呼ばれている。つまり、反復ごとに変化する情報を計算に取り込まないという意味で定常的手法と呼ばれる。定常的手法は、歴史が古く、理解しプログラムを作るのは簡単であるが、現在は効率的と言えない場合が多い。この種の解法としては、ヤコビ法、ガウス-ザイデル法、逐次過剰緩和法 (SOR 法) が挙げられる。

これに対して、非定常的手法は反復毎に変化する情報を計算に取り込むという点で定常的方法と異なる。これらは、比較的最近開発されたもので、手法としては理解し難いが、一般的に定常的手法より効率的である。非定常的手法は、その多くが直交するベクトル列を作り出すというアイデアに基づいており、定数は残差ベクトルの内積、あるいは反復法で生成される他のベクトルの内積より計算される。非定常的手法としては、共役勾配法 (CG 法)、双共役勾配法 (BiCG 法)、2乗共役勾配法 (CGS 法)、安定化双共役勾

配法 (BiCGSTAB 法)、一般化最小残差法 (GMRES 法)、2 乗共役勾配法 (CGS 法) 等が挙げられる。

このように、反復法には様々な手法が提案されており、今後もさらに新しい手法が出てくると考えられる。しかし、効率の良い計算を行うためには、係数行列の性質に従って最適な手法を選択し、場合によっては前処理行列を用い、収束性の改善を図ることによって、安定した効率のよい方法を用いることが反復法を使う鍵となる。

2.2.3 解法の選択

前節で述べたように反復法には様々な解法があるが、これらは対象となる係数行列の性質によって、最適な方法を選ぶ必要がある。例えば、文献[9]には反復法選択のお勧めを示す流れ図が掲載されている。この手法に従えば、今回対象となる熱伝導解析および応力解析とも、係数行列は、正定値対称行列となり、またどちらの解析でも一般に固有値の範囲は既知ではない。よって、この流れ図に従うなら、共役勾配法 (CG 法) が最もよい方法であることになる。

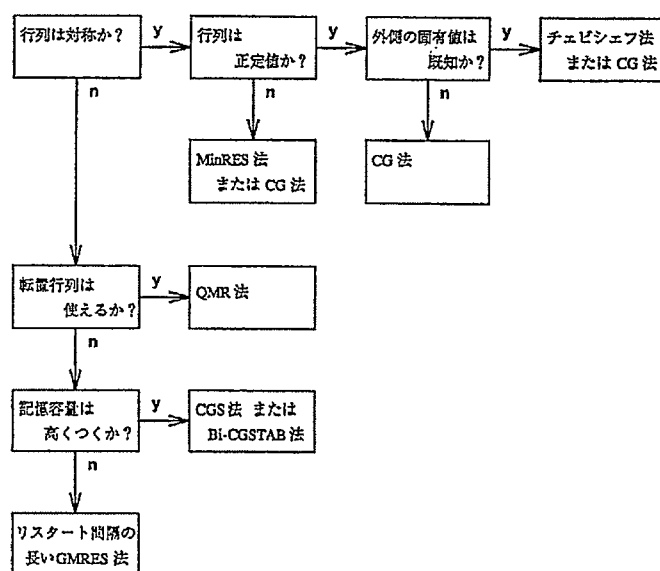


図2.1 反復法選択のおすすめを示す流れ図

また、米国ローレンスリバモア国立研究所による SLAP (Sparse Linear Algebra Package) Ver 2.0 の SLAPDOC にも”どの手法を使うべきか”という項がある。ここでも、”

もし係数行列が対称であり、さらに正定値ならば、共役勾配法を選べとある。この他にも、これまでの経験から言って、現状では、正定値対称行列については、共役勾配法 (CG 法) が最もよい方法であることに間違いは無い。そして、収束性の改善のために必要に応じてなんらかの前処置を選ぶことになる。

2.2.3.1 共役勾配法 (CG 法)

共役勾配法 (CG 法) は、正定値対称行列を係数とする連立一次方程式に対して効果的な解法で、非定常的な解法の中では最も古いもののひとつである。この解法の導出過程は省略するが、対称行列についての汎関数の最小値問題を逐次近似法で解くことから導びかれる。共役勾配法では、反復の残差ベクトルとして汎関数の直交な勾配ベクトルの列が生成される。反復過程においては、係数行列の他にこれらの残差ベクトル、反復解ベクトル、探索方向ベクトルなどの少数のベクトルを記憶するだけでよいために、記憶容量的にも有利である。

以下に、 $Ax = b$ を解くための共役勾配法のアルゴリズムを示す。ここに A は係数行列、 b は右辺項ベクトル、 x は (反復) 解ベクトル、 p は探索方向ベクトル、 r は残差ベクトルであり、 α, β はスカラー量、 ε は収束判定基準、また、下付き指標は反復数、ベクトル 2 項の括弧は内積を表す。

共役勾配法のアルゴリズム

```

 $x_0$  is an initial guess,  $r_0 = b - Ax_0$ , set  $\beta_{-1} = 0$ ,
for  $n=0, 1, \dots$  until  $\|r_n\| \leq \varepsilon \|b\|$  do :
begin
   $p_n = r_n + \beta_{n-1} p_{n-1}$ ,
   $\alpha_n = (r_n, r_n) / (p_n, Ap_n)$ ,
   $x_{n+1} = x_n + \alpha_n p_n$ ,
   $r_{n+1} = r_n - \alpha_n A p_n$ ,
   $\beta_n = (r_{n+1}, r_{n+1}) / (r_n, r_n)$ ,
end

```

このアルゴリズムにおいて、 α は $(r_n, A^{-1}r_n)$ を最小にするように、 β は p_n と Ap_n が直

交する（もしくは同値な条件である r_n と r_{n-1} が直交する）ように決めている。従って、この手法では、残差ベクトル列が直交するので、残差は高々、未知数回の反復によって零になる、つまり、収束性が保証されていることになる。

2.2.3.2 前処理付き共役勾配法

共役勾配法において、収束性を改善するために前処理行列として M を用いる場合のアルゴリズムを以下に示す。ただし、この共役勾配法の根拠となっている、対称行列についての汎関数の最小値問題としての意味を満たすために、前処理行列として M も正定値対称行列とする必要がある。

前処理付き共役勾配法のアルゴリズム

```

 $x_0$  is an initial guess,  $r_0 = b - Ax_0$ , set  $\beta_{-1} = 0$ ,
for  $n=0, 1, \dots$  until  $\|r_n\| \leq \epsilon \|b\|$  do :
begin
 $p_n = M^{-1}r_n + \beta_{n-1}p_{n-1}$ ,
 $\alpha_n = (M^{-1}r_n, r_n) / (p_n, Ap_n)$ ,
 $x_{n+1} = x_n + \alpha_n p_n$ ,
 $r_{n+1} = r_n - \alpha_n A p_n$ ,
 $\beta_n = (M^{-1}r_{n+1}, r_{n+1}) / (M^{-1}r_n, r_n)$ ,
end

```

2.2.4 前処理

反復法が収束する速さは係数行列の固有値分布（スペクトル）に強く依存する。よって係数行列を変換することによって、より都合のよい固有値分布に変換するような行列を用いることが有効となる場合がある。この変換行列のことを「前処理行列」と呼ぶ。前処理行列によっては反復法の収束性が改良される場合が多く、極端な場合、前処理行列なしでは反復法が収束しないこともある。

例えば、連立一次方程式として、 $Ax = b$ が与えられたとき、前処理行列 M の逆行列を作用させて得られる以下の式を解いてもよい。このとき前処理行列は、元の係数行列 A に較べて $M^{-1}A$ の固有値分布が、収束性を改善できることを目的に決定する。

$$M^{-1}Ax = M^{-1}b \quad (2-5)$$

上の式から明らかなように、 $M=A$ であれば、 $M^{-1}=A^{-1}$ であるから、前処理行列を作用させるだけで（反復計算を実施することなしに）、正解を得ることができる。しかし、このような前処理を行うためには、結局、前処理行列、つまり係数行列の逆行列を必要とする。例えば、係数行列が正定値対称であれば、この過程は直接法でコレスキー分解を行うのに等しく、直説法を採用することに等価となる。このような考え方を適用するならば、前処理は係数行列の逆行列を求める過程を付加することによって、直接的な解法を反復法に持ち込んでいると考えることもでき、以下に説明する対角スケールリングや不完全コレスキー分解も、近似的な逆行列を求める方法として理解できる。

前処理を導入することは、前処理行列（またはその逆行列）を求めるためと、反復の演算過程で、演算回数が増加するから、これらは前処理を導入する効果（収束性の改善）との比較によって（トレードオフ）、最適な前処理を選ぶことになる。つまり、前処理を用いない場合に較べて、前処理と反復法のトータルとしての演算量が、低下するような手法を選択することが有効である。

2.2.4.1 ポイントヤコビ

先に示したように反復法で前処理を行うためには、前処理行列の逆行列を求める必要がある。よって、簡単に逆行列を得るために前処理行列を対角項のみとし、これを係数行列に等しくしたのがポイントヤコビ前処理である。この手法は対角スケールリングとも呼ばれ、最も簡単で効率的な前処理であり、実用的には対角優位の場合に効果を発揮する。（ある条件の基では、ポイントヤコビ前処理が全ての対角形式の前処理において条件数を小さくする意味において最適であることが証明されている）また、この前処理は対角項のみを記憶すればよいので、記憶容量の点からも有利である。

ここに係数行列の要素を a_{ij} 、前処理行列の要素 m_{ij} としたときに、ポイントヤコビの前処理行列は以下の式で与えられる。

$$m_{ij} = \begin{cases} a_{ij} & (i = j) \\ 0 & (i \neq j) \end{cases} \quad (2-6)$$

2.2.4.2 不完全コレスキー分解

前処理の大きな系統として、三角分解を行うものがある。ここに係数行列が正定値対称であれば、コレスキー分解、非対称であれば三角分解(LU分解)を行うことになる。ただし、三角分解の過程において、非零の係数行列要素を持つ部分に値が入ることになる(fill-in)が、これを一定の法則に従って制限することによって不完全な分解を行うのがこの手法である。例えば、不完全コレスキー分解より一般的な不完全LU分解はILUと表記されるが、ILU(0)のように括弧内に fill-in の項数を示す。ここに(0)は fill-in を考慮しない場合であり、係数行列で零の要素を持つ場所は前処理行列の逆行列でも同様に零とすることを意味する。つまり、不完全コレスキー分解では、対角要素として以下の式を用いることになる。

$$a_{ii} = a_{ii} - \sum_{v=1}^{i-1} \sum_{k=1}^{v-1} a_{vk} a_{kv} / a_{vv}^{(v)} \quad (2-7)$$

このような不完全な分解過程の問題点は、正定値性の保存ができない場合があることである。つまり、係数行列が正定値であっても不完全な分解後の行列は正定値になると限らない。この場合は正値対称行列であっても不完全コレスキー分解が破綻してしまうため(分解過程で対角項が負になる)、不完全LU分解(対角項をごまかして分解を続ける方法も提案されている)や他の前処理を用いる必要が出てくる。

2.3 FINAS への反復法組込みの設計

2.3.1 反復法の組込み方針

(i) FINAS 本体と独立した形で組込む。

反復法ソルバと FINAS のインターフェースを明確にし、反復法ソルバは FINAS と独立した形で組込む。このために、反復法ソルバのための変数領域は、Fortran-90 の allocate 文で動的に確保し、FINAS 側との変数の受渡しは、値の代入（ポインタ渡しは使わない）を基本とする。

(ii) 反復法を利用可能な解法は、静的応力解析と熱伝導解析とする。

FINAS の直接法ソルバでは、三角分解後の係数行列を記憶しており、これによって、多数の右辺項がある場合、効率的な演算ができるようになっている。これに対して、反復法では、各右辺項に対して反復計算を行う必要があり、右辺項の数だけ反復計算を実行する必要が生じる。このような観点から、今回の反復ソルバを有効とするのは、静的応力解析と熱伝導解析とし、静的応力解析においても多荷重処理は含まない。

(iii) SLATEC の利用

反復法ソルバの解法部分は、信頼性と効率を考慮して SLATEC を使用する。SLATEC は、米国のロスアラモス国立研究所の W. Robert Boland、ローレンスリバモア国立研究所の Kirby W. Fong、Tokihiko Suyehiro、サンディア国立研究所の Thomas H. Jefferson, Lee Walton らによって開発された疎行列の線形演算に関する共通数学ライブラリ (CML: Common Mathematical Library) で、パブリックドメインのソフトウェアである。SLATEC の中には、SLAP (Sparse Linear Algebra Package) や BLAS (Basic Linear Algebra Subprograms) が含まれており、信頼性のある数値計算を効率的に行うことができる。

2.3.2 反復法の組込み方法

FINAS では静的応力解析機能、熱伝導解析機能ともに、ソルバ部分は同様な手順に従っている。つまり、ファンクションモジュール TRIA で三角分解を行い、その後 SOLVE で前進後退代入を行うことにより、連立方程式の解を得る。ただし、これらソルバの呼び出しは各解法によって一度とは限らないので、各々についての処理が必要である。また、これら TRIA, SOLVE の主要な入出力である(下三角)全体マトリックス、右辺ベクトル、未知量ベクトルはデータブロックの形で与えられる。よって、これら TRIA, SOLVE の過程を反復ソルバに置き換えれば良いのだが、多荷重処理のように三角分解後のマトリックスを多数の右辺項に作用させる場合等に注意が必要となる。この手順を図に示せば以下のようなになる。

- ・ファンクションモジュール TRIA を拡張し、反復法が選択されていた場合、係数行列を次節に示す方式でメモリ上の配列に保存する。
- ・ファンクションモジュール SOLVE を拡張し、反復法が選択されていた場合、右辺項ベクトルをセットし、係数行列により反復計算を開始する。

なお、反復法、前処理のためのルーチンは、これら TRIA, SOLVE から、呼び出す。

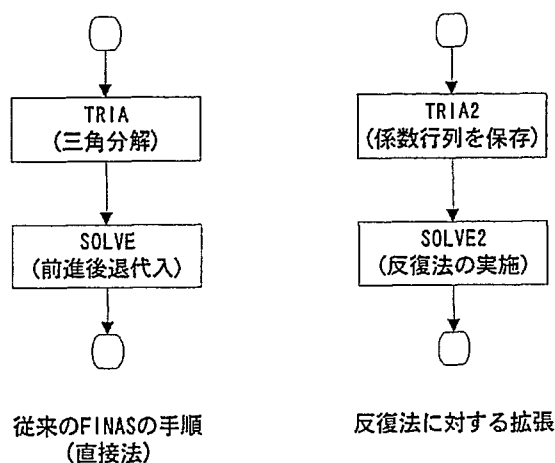


図2.2 反復ソルバ追加の概念図

2.3.3 FINAS の入力データ

反復ソルバの指定にはこれまでのリナンバーアルゴリズム選択のためのデータ (V15.0) に以下の項目を追加する。

iii) 反復ソルバの選択のためのデータ

	10	20	30	40	50	60	70	80
B ₁₂	ISOLV		ime ipm		imax	itol		ipr

ISOLV データは、反復ソルバを選択するためのカードである。このカードが入力されない場合は、連立方程式の解法として FINAS のデフォルトである直接法ソルバが選択される。ISOLV カードでは、21 から 25 カラムの ime に反復法の手法、26 から 30 カラムの ipm で前処理の手法を選択する。また、反復法における最大反復回数と収束判定基準は imax、itol で指定し、ipr では、収束状況のログファイルの出力を選択する。

反復法が適用可能なのは、熱伝導解析機能と静的応力解析機能に限定されている。

カードの種類	カラム	タイプ	変数	内 容	無記入の場合
B ₉	6-15	英 字	—	ISOLV と記入	*
	21-25	英 字	ime	反復法の手法 CG : CG 法	CG
	26-30	英 字	ipm	前処理手法 NONE : 前処理無し DS : 対角スケーリング (ポイントヤコビ) IC : 不完全コレスキー分解 ILU : 不完全 LU 分解	DS
	36-40	整 数	imax	計算打ち切り反復回数	未知数
	41-50	正実数	itol	反復法の手法	10 ⁻⁷
	66-70	英 字	ipr	収束状況のログ出力 LOG : 収束ログ出力 NLOG : 出力しない	NLOG

2.3.4 反復法組込み手順

FINAS への反復ソルバ組込みにおいては、ソルバが本来独立して動作するのが一般的であること、SLATEC、その他利用できる反復法ソルバが存在することを考慮し、以下の手順を踏んで組込み作業を実施した。なお、作業の実施においては、このように段階的な組込み作業を行うことによって、誤りの発生を特定でき、致命的な誤りの発生が防げるようになる。

(i) ファイル入力によるソルバの動作確認

まず、反復ソルバを作成し、これに係数行列と右辺項をファイルで与えることによってソルバ部分の動作を確認する。ここに、ソルバの入力となる係数行列等のファイルは、FINAS 以外のプログラムで作成したものをを用いる。

(ii) FINAS の係数行列・右辺項をファイル出力し、ソルバの動作を確認①

次に、FINAS の係数行列と右辺ベクトルを一度ファイルに出力し、これを別に動かすソルバプログラムの入力とすることによって、FINAS と同様の解が得られることを確認する。

(iii) FINAS の係数行列・右辺項をファイル出力し、ソルバの動作を確認②

先の手順に同様であるが、FINAS 本体とソルバ部分を同時にコンパイルし、一つの実行形式として解が得られることを確認する。この時点では、FINAS 本来のものと今回組込む反復法の2つソルバが動作している。

(iv) FINAS の係数行列・右辺項をファイル出力し、ソルバの動作を確認③

これまでは、FINAS の解ベクトルとしては、FINAS に組込まれた値を用いてきたが、これを組込みソルバの値で置き換える(必要に応じてファイルによる入力からメモリ上のコピーへと手順を踏む)。その後、応力等、状態変数の計算が正常に行われることを確認する。

(v) FINAS への反復ソルバの組込み

反復ソルバが動作するときは、FINAS 本体のソルバが動作しないようにする。また、係数行列、右辺項、解ベクトルをメモリ上で引数として渡し、組込みが正常に行われたことを解析結果として確認する。

(vi) 入出力の整備

今回の組込みでは、入力としては、コントロールデータによる制御、出力としては、収束状況のログと実行サマリのみが対象になっている。これら入出力部分の整備を行う。

(vii) ベンチマーク解析による性能評価

ベンチマーク解析を実施し、ソルバの性能を評価する。評価用の問題としては、まずは最も簡単で、自由度数を任意に変更することのできる単位立方格子の弾性解析を扱う。その後、順次、各種の問題についてベンチマーク解析を実施し、計算時間、反復回数、メモリ使用量等の検討を行う。

2.4 FINAS への反復法組込み作業

2.4.1 反復法組込み作業の概要

FINAS で現在用いられているソルバ(ここで言うソルバとは連立一次方程式を解く部分で固有値ソルバは含まない)は、以下の2段階より成り、各々以下に示すファンクションモジュールで構成されている。

① 三角分解：(TRIA)

係数行列(構造問題では全体剛性行列)を三角分解(対称行列ではコレスキー分解)する。

② 求解過程：(SOLVE)

境界条件を考慮して解を求める過程である。ソルバ側から見た主要な部分は、前進後退代入であり、三角分解後の係数行列と右辺ベクトルについて前進後退代入を実施し、解を求める。

従って、反復ソルバもこの部分に接続することになる。実際の操作としては、TRIAで係数行列をメモリ上の保存し、SOLVEで実際に反復法を実行して解を得る。よって、直接法では、TRIA(三角分解)の操作の方がSOLVE(前進後退代入)よりはるかに演算量が大きい。反復法の実際の操作は、SOLVEで行われるため、TRIAよりSOLVEの演算量が大きくなる。

2.4.2 SOLVE での係数行列保存

三角分解を行うファンクションモジュールはTRIAであるが、実際の三角分解の操作は、TRIAから呼ばれるSDCOMPで行われており、このルーチンを中心に考えることになる。今回対象としている、静的応力解析機能と熱伝導解析機能では、係数行列が正定値対称になるため、SDCOMPでは、コレスキー分解を行うことになる。SDCOMPはV15.0のBCSソルバ(ポーイングによるマルチ フロントアル ソルバ)の追加によって、BCSソルバ用となり、従来のSDCOMPはSDCOMPOとしてSDCOMPから呼ばれている。よって、反復ソルバ用の係数行列を保存するルーチンも同様に、SDCOMPより呼び出すものとし、これをSDCOMPI(三角分解をするわけではないので、decompを付けるのも変だが、名称上の整合性を保つため、このようにした)とする。

〈TRIA のサブルーチン関係〉

TRIA ファンクションモジュール

SDCOMP 三角分解：マルチフロントルソルバ用

SDCOMP0 三角分解：従来のソルバ用 (Old)

V15.0 で追加済

SDCOMPI 反復ソルバ用の係数行列格納 (Iterative)

V16.0 で追加

2.4.2.1 SDCOMPI での操作

SDCOMPI での操作は、マルチフロントルソルバ用の SDCOMP を参考にすることができる。FINAS は、係数行列の l-set (未知数の次元) の非零成分を圧縮した形式で、FFCP (Finas File Control Package) で提供される仮想ファイルとして保存している。ここに、マルチフロントルソルバの例を挙げるなら、SDCOMP での操作は以下のようになる。また、ここに xds から始まるサブルーチンはマルチフロントルソルバ用のユーティリティールーチンである。

〈マルチフロントルソルバ SDCOMP の操作手順〉

xdslin: マルチフロントルソルバの初期化を行う。

FFCP の係数行列(A)をメモリ上に展開する。

sdmin: 係数行列の構造をチェックする。(引数: IFLAG=0)

sdlif: 係数行列の構造を決める。

xdslor: リナンバを行う (Metis の場合は xdslom)

xdslsf: シンボリックファクトライゼーション

xdslmc: コアの最適化

再び FFCP の係数行列(A)をメモリ上に展開する。

sdmin: 係数行列をマルチフロントルソルバにセットする。(引数: IFLAG=1)

xdslvf: 係数行列のセットを完了する。

xdslfa: 不完全コレスキー分解の実施。

xdslsr: 統計情報を得る。

getwkr: 三角分解後の係数行列(L)を FFCP で書き込む。

さて、l-set の係数行列 KLL は上三角行列の非零成分が行毎に格納されている。ここ

では、一定の書式(係数行列の値を記憶する実数の1次元配列と各要素の列番号及び、最初の非零項を記憶する整数型配列2本)に従って記憶すればよい。しかし、FINASのKLLの格納方式では、係数行列の非零項の割合があるのみで、その数が記憶されていないため、反復法においてもマルチフロントルソルバ同様、配列の大きさを決めるためにKLLをスキャンする必要がある。つまり、反復法でもマルチフロントルソルバ同様の手順を踏むため、KLLを2度走査する必要が生じる。以下に反復法での処置手順を示す。係数行列はFFCPへ出力するのではなく、メモリ上に展開するため、最後にFFCPへ書き込む手順が不要となるが、ATOMの整合性をとるために、ダミーの(L)として、長さが1の単位行列を書き込んでいる。また、今回反復ソルバ用のルーチンは、サブルーチン名の前にis_を先行させた。ただし、sadminiのみは、sadminに習ったため、このルールを踏襲していない。

〈反復ソルバSDCOMP1の操作手順〉

is_init: 反復ソルバの初期化を行い、配列が確保されているかを調べる。

FFCPの係数行列(A)をメモリ上に展開する。

sadmini: 係数行列の構造をチェックする。(引数: IFLAG=0)

is_lil 係数行列非零項の行番号、列番号をセットする。

is_lif: 係数行列等の配列領域を確保する。

(今回、リナンバは考慮しない)

再びFFCPの係数行列(A)をメモリ上に展開する。

is_lvi: 係数行列セットのための指標を初期化する。

sadmini: 係数行列をマルチフロントルソルバにセットする。(引数:

IFLAG=1)

is_lv1 係数行列の非零項の値をセットする。

is_lvf: 係数行列のセットを完了する。

(is_save_m:係数行列をファイルに出力する。) デバッグ用
仮の(L)を書き込む、(L)は1次元の単位行列とした。

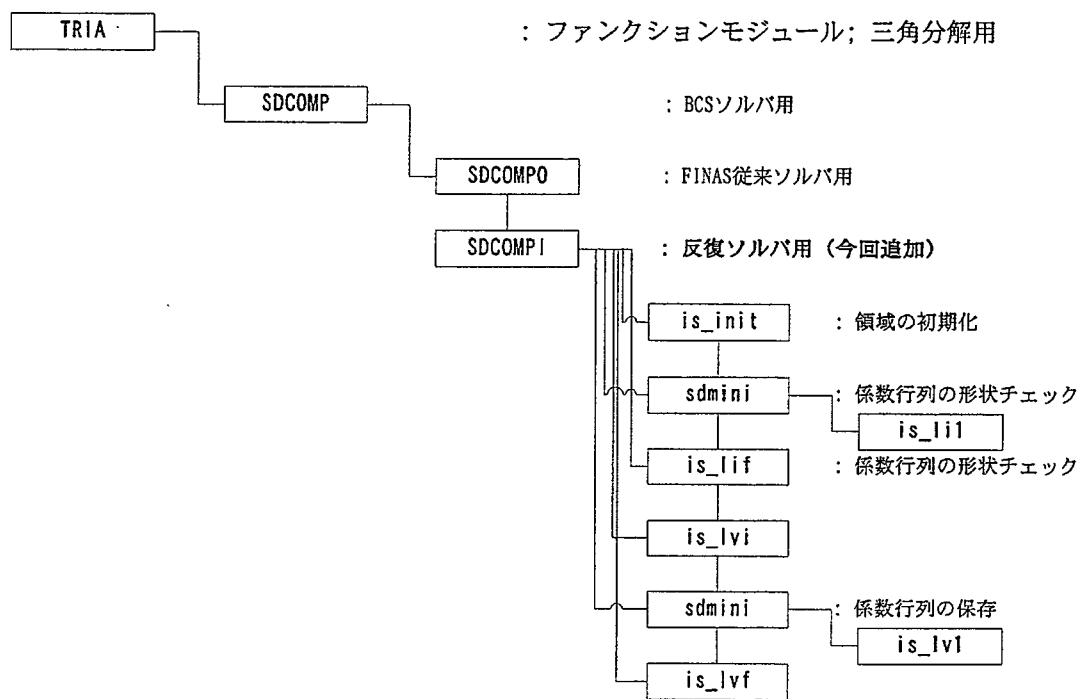


図2.3 三角分解過程:TRIA への組み込み

2.4.3 FBS への反復法本体の組み込み

三角分解の操作で明らかなように、FINAS で連立一次方程式を解く部分は、最終的な未知自由度である l-set について実施される。よって、ファンクションモジュール: SOLVE においても実際に前進後退代入をしている部分である FBS を中心に考えればよい。

FBS は、文字通り前進後退代入 (Forward-Backward Substitution) を実行しており、三角分解後の係数行列 (下三角行列) :L と右辺ベクトル:B を入力として、前進後退代入の操作を行い、解ベクトル:X を出力する。これら、SOLVE におけるサブルーチンの関係を以下に示す。FBS は V15.0 のマルチフロンタルソルバの追加によって、マルチフロンタルソルバ用となり、従来の FBS は FBS0 として FBS から呼ばれる。よって、反復ソルバを呼び出すルーチンも同様に、FBS より呼び出すものとし、これを FBSI (前進後退代入ではないので、FBS を付けるのも変だが、他との関係もあり、このようにした。これは SDCOMPI に同様である)

<SOLVE のサブルーチン関係>

SOLVE ファンクションモジュール
 FBS 前進後退代入: マルチフロンタルソルバ用

- FBS0 前進後退代入：従来のソルバ用 (Old) V15.0 で追加済
- FBSI 反復ソルバ実行用の最上位ルーチン (Iterative) V16.0 で追加

2.4.3.1 FBSI での操作

反復法用のルーチン FBSI では、以下の操作を行うことになる。これらは、以下のサブルーチンを使って実現する。また、FBSI では、これらのサブルーチンを有効にするために、モジュール `m_is` を use する。

- i) 右辺ベクトルを $Z(i)$ に展開する。 is_rh
- ii) 反復ソルバを呼び、解ベクトルを求める is_pcg
- iii) 解ベクトルを $Z(i)$ に保存する。 is_u

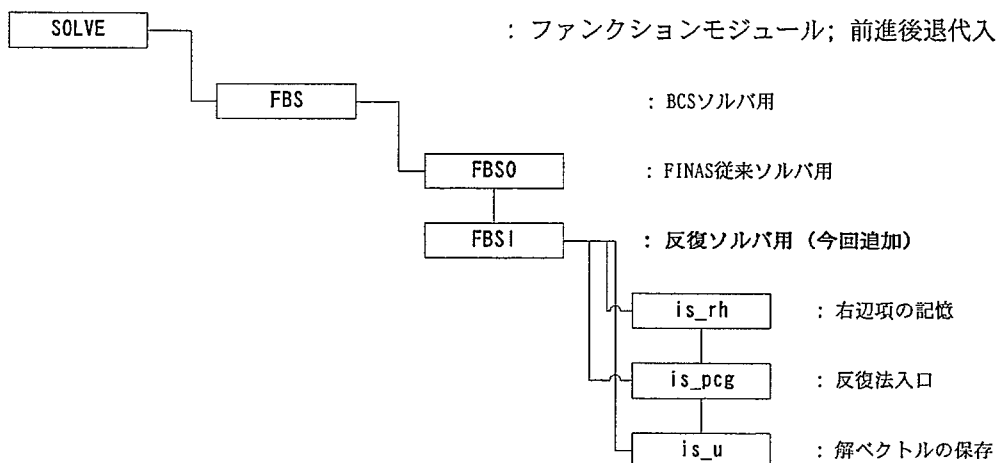


図2.4 前進後退代入過程:FBSI への組込み

2.4.4 反復ソルバ用変数領域の追加

今回の組込みでは、反復ソルバ用の変数領域(係数行列、右辺項、前処理等のワーク領域等)は、FINAS のオープンコア領域を使わず、fortran-90 の allocate(アロケート)文によって、別領域に動的に確保するようにした。この方法は、一般にメモリが無駄になりがちであるが、今回の組込みでは、これらの領域がメモリ上に確保できるとの仮定のもとに行っており、FINAS 本体は少ないメモリでも動作可能となっているため、実用上問題は無い。

実際の組込み操作においては、モジュール `m_is` を定義し、この中のモジュール変数として反復ソルバ用の変数領域を定義し、さらにモジュール `m_is` の内部サブルーチンとして、反復法の入口となる `is_pcg` 等を定義している。従って、FINAS の組込みにおいては、実際にこれらのルーチンが呼び出される `SDCOMPI` と `FBSI` のみで `m_is` を use すればよい。また、`m_is` に定義した反復ソルバ用の変数領域は、`private` 宣言をしているため、内部サブルーチンからしか参照できないような仕様にしている。このように、反復法の領域と FINAS の領域は完全に分離できるように設計されており、変数等も全てコピーして使うように留意した。これらの変数の内容については、モジュール `m_is` の先頭に記述しているが、これを表 2-1 に示す。

```

-----
* variables for solver interface
  n      (int., in)  : number of unknowns
  nelt   (int., in)  : length of 1-d array ia, ja, a
  isym   (int., in)  : matrix symmetric key (1; sym., 0; not.)
  itol   (int., out) : tolerance key
  itmax  (int., in)  : max. of iteration count
  iter   (int., out) : iteration count
  ierr   (int., out) : err key (0; normal)
  iunit  (int., in)  : convergence log print unit
  lenw   (int., in)  : array length of rwork
  leniw  (int., in)  : array length of iwork
  tol    (d.p., in)  : convergence judge tolerance
  err    (d.p., out) : estimated error of result
  ia(nelt) (int., in) : column index
  ja(nelt) (int., in) : row index
  a(nelt) (d.p., in)  : coefficient matrix (lower triangle)
  x(n)    (d.p., inout) : initial and solution vector
  b(n)    (d.p., in)  : right-hand vector
-----
  integer, private :: n      , nelt, isym , itol, itmax,
&                  iter , ierr, iunit, lenw, leniw
  double precision, private :: tol , err
  integer, private, dimension(:), allocatable
&                  :: ia, ja, iwork
  double precision, private, dimension(:), allocatable
&                  :: b, x, a, rwork
-----
* variables for count terms
  column_count(n) (int.): column counter
-----
  integer, private, dimension(:), allocatable :: column_count
-----
* variables for save counter
  pos      (int.): 1-d array position pointer
  i_last   (int.): last column i
  j_last   (int.): last column j
-----
  integer, private :: pos , i_last, j_last
-----
* variables for allocation
  mem_len (int.): allocated array memory length in words
  n_int   (int.): integer type variable length
  n_real  (int.): real type variable length
  key_init (log.): initialize iterative solver key
  key_alloc (log.): area allocation key
-----
  integer, private :: mem_len = 0
  integer, parameter, private :: n_int = 1
  integer, parameter, private :: n_real = 2
-----
* key flags
  key_init (log.): initialize iterative solver key
  key_alloc (log.): area allocation key
-----
  logical, private :: key_init = .false.
  logical, private :: key_alloc = .false.
-----
* parameters for debug
  key_debug (log.): debug key
  key_file_out (log.): matrix and vector file out key
-----
  logical, parameter, private :: key_debug = .true.
  logical, parameter, private :: key_file_out = .false.
  logical, parameter, private :: key_file_out = .true.
  logical, parameter, private :: key_file_out = .false.
=====

```

表2.1 反復ソルバ用の変数領域(1)

2.4.5 共通領域(COMMON)の追加

先の反復ソルバ用の変数領域とは別に、FINAS 側とのインターフェースとして、反復ソルバの制御データをコモン領域として確保している。これらの値は、初期値が ITESLB で設定された後、コントロールデータを参照して定義される。ISLOG は、-1 以外であれば、ISYS (51) の値、ISITE、ISMEM は反復ソルバ経過後に実際の反復回数とメモリ使用量が設定される。

```

.....
* control data for iterative solver
      written by by k.garatani (dec.2002)

      istol (d-pre. real): tolerance for convergence
      ismeth (integer  ): method of iterative solver
                        where -1; not iterative solver
                        1; c.g. method
      ispre (integer  ): method for precondition
                        where 0; no precondition
                        1; diagonal scaling
      isimax (integer  ): max. count for iteration
      islog (integer  ): conv. log output file code (-1; no print)
      isite (integer  ): actual iteration count at solver
      ismem (integer  ): memory used in iter. solver [byte]
.....

```

表2.2 反復ソルバ用の変数領域(2)

3. 大規模 3次元有限要素解析への適用評価

3.1 反復法の性能評価-1

ここでは、組込んだソルバの性能評価をするためいくつかのベンチマーク問題について、検証解析を実施した。なお、解析にはPC(Pentium-4 2.53GHz)を用い、解の収束判定基準としては、残差の相対ノルムを 10^{-7} 以下とした。

3.1.1 単位立方格子の弾性解析による性能評価

単位立方格子の弾性解析は、図 3.1 に示す長さが 1.0 の立方体の X 方向に一様な力を作用させ、その変位、応力を弾性解析によって求めるものである。このモデルは、メッシュ分割のための一辺の節点数を変化させることによって、様々な規模の問題を作成することができ、データの作成が容易なことから、様々な大規模問題のベンチマークとして採用している。

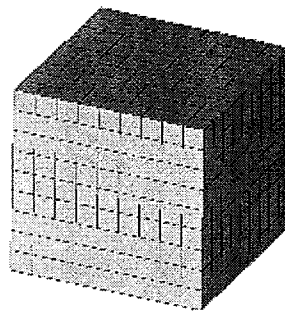


図 3.1 単位立方格子モデル

3.1.1.1 単位立方格子問題での記憶容量、演算量の推定

単位立方格子は、形状が単純なことから、計算での演算量を容易に推定できる。図 3.1 に示すように、一辺の節点分割数を n とし、X, Y, Z 軸について同様なメッシュ分割を行うとするなら全ての演算量等の特性はこの n のみで記述できる。

1 辺の節点分割数 :	n
1 辺の要素分割数 :	$n-1$
全節点数 :	n^3
全要素数 :	$(n-1)^3$

未知数数： $3 \times n^3$ (1 節点 u, v, w の 3 方向の変位)

・直接法での演算量と使用メモリの推定

この問題では、係数行列がバンドマトリックスとなる。バンドマトリックスは、直接法での演算量が、(未知数)×(バンド幅)²に比例すると言われている。結局、この問題では、直接法での演算量が n の 7 乗、または未知数の $7/3$ 乗に比例する。

バンド幅： $2 \times (3n)^2$

直接法の演算量： $\alpha' d \times (3 \times n^3) \times \{2 \times (3n)^2\}^2$ ($\alpha' d$ は比例定数)

または、 $\alpha_d \times n^7$ (α_d は比例定数)

バンドマトリックスを直接法で解く場合、Fill-in のため、対角項から最外非零要素の内側を演算のために記憶する必要がある。ここでは、簡単のためにマトリックスの対称性と端部での分布を考えないとすると、係数行列の項数は、(未知数)×(バンド幅)となる。ここに、1 変数のバイト数として 8 バイト(倍精度実数)を仮定すると、以下のように必要なメモリ量が n の 5 乗、または未知数の $5/3$ 乗に比例する

直接法の必要メモリ量 $3 \times n^3 \times (3n)^2 \times (1 \text{ 変数のバイト数: } 8)$

または、 $216 \times n^5$

・反復法(CG法)での演算量と使用メモリの推定

反復法での主要な演算は、係数行列と未知ベクトルの乗算過程である。反復法では、係数行列の非零項のみ、演算を実施すればよいが、1 自由度あたりの非零項は、要素の接続によって変化する。今回のように 8 節点六面体要素を用いる場合は、1 節点の 8 個の要素が接続し、これらの要素の節点分だけ接続関係が発生する。よって一反復あたりの演算量は(未知数)×(1 自由度あたりの非零項)に比例し、以下の式で与えられる。

1 自由度あたりの非零項： 3×27 (8 節点六面体要素の場合)

1 反復あたりの演算量： $\alpha' \cdot 3 \times n^3 \times 3 \times 27$

このような問題の反復回数は、経験的に一辺の節点分割数に比例することが知られて

いる（理論的背景としては固有値の比である条件数がこの関係を持つ）。よって、反復数での演算量は以下の式となる。よって、反復法での演算量が n の 4 乗、または未知数の $4/3$ 乗に比例する。

1 自由度あたりの非零項： 3×27 （8 節点六面体要素の場合）

反復法での演算量： $\alpha'_i \times 3 \times n^3 \times 3 \times 27 \times n$

または、 $\alpha_i n^4$

さらに、メモリ量についても非零項のみを記憶すればよいから、以下のようになり、約自由度数の 1,000 倍のメモリを必要とすることになる。ただし、ここでは、1 変数あたりに必要なメモリとして実数型変数と指標としての整数型変数の 12 バイトを仮定した。

反復法での必要メモリ量 $3 \times n^3 \times 3 \times 27 \times 12$ × (1 変数のバイト数： 12)

または、 $972 \times n^3 \approx 1000 \times 3 \times n^3$

また、有限要素として、20 節点の六面体を用いるなら、1 節点に接続される要素の数は 8 つであるが、節点数は 93 個となる。よって、8 節点の場合に較べて約 3.5 倍のメモリが必要となる。

3.1.1.2 より一般的な場合での演算量の推定

以上、単位立方格子モデルについて、直接法、反復法での演算量と必要メモリを検討した。単位立方格子モデルは、3 次元的な分割であるが、図 3-2 に示すように、2 次元及び 1 次元的な分割とすることもでき、同様な検討を行うことができる。このときの演算量と必要メモリの関係を表 5-1 に示す。表より明らかのように問題規模が大きくなる、つまり n が大きくなった場合は、1 次元的問題では直接法が有利で、3 次元的問題では、反復法が有利となる。これは、1 次元的問題では、バンド幅は小さいが反復法での収束性が劣り、逆に 3 次元的問題ではバンド幅が大きくなるが、反復法での収束性に優れることによる。

ただし、大規模問題において、1 次元的な分割をするとは考え難いので（実用的には 2.5 次元程度と思われる）、一般に言われているように、大規模問題ほど、反復法が有

利となる。

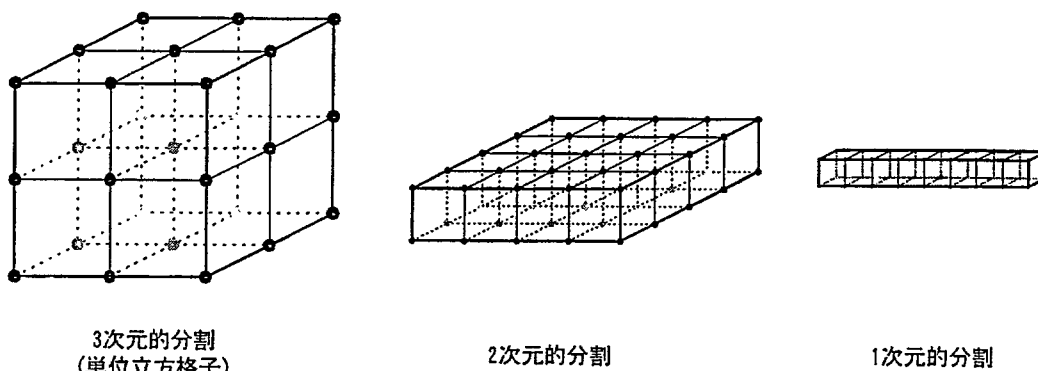


図3.2 分割方法

表3.1 1次元、2次元、3次元的な分割とメモリ・演算量が比例する基数

	手法	1次元的分割	2次元的分割	3次元的分割
演算量	直接法	n^3	n^5	n^7
	反復法	n^6	n^5	n^4
メモリ	直接法	n^3	n^4	n^5
	反復法	n^3	n^3	n^3
評価		直接法有利	両者拮抗	反復法有利

n は1辺の節点数

3.1.1.3 FINAS による性能評価

以下の表に、FINAS に組込んだ反復ソルバと従来のウェーブフロント法 (FINAS 固有のソルバ)、マルチフロンタル法 (ポーイングより供給されたソルバ、マルチフロンタル) の比較を示す。これらの計算は P-4 (2.53GHz)、メモリ 1GB の PC (OS: Windows-2000) で、FINAS のオープンコアメモリを 64MB (PC でのデフォルト) として実行したものである。

これらの図表から、明らかなように、反復法は、直接法より有利であり、さらに問題規模が大きくなるに従って、その差が大きくなることが解る。ただし、直接法ではメモリ領域等の工夫によって、ある程度の性能改善が見込まれる。

表3.2 反復ソルバでの処理時間（対角スケーリング、CG法）

n	節点数	要素数	未知自由度数	ディスク [Block]	合計時間 [Sec.]	ソルバ 時間 [sec.]	反復法 の時間 [sec.]	反復 数	メモリ [MB]
10	1,331	1,000	3,630	2,580	0.5	0.2	0.2	95	2.0
15	4,096	3,375	11,520	8,574	2.0	1.1	0.8	144	6.8
20	9,261	8,000	26,460	20,169	5.5	3.5	2.7	191	16.2
25	17,576	15,625	50,700	39,945	13.2	8.5	7.1	239	31.6
30	29,791	27,000	86,490	67,657	26.2	17.0	14.5	287	54.7
35	46,656	42,875	136,080	109,118	48.4	31.0	27.1	336	86.9
40	68,921	64,000	201,720	159,922	81.2	51.1	45.3	383	129.8
45	97,336	91,125	285,660	227,492	132.0	81.2	72.7	431	184.7
50	125,000	117,649	367,500	297,520	179.4	113.0	103.4	470	238.5
60	226,981	216,000	669,780	545,360	406.9	247.7	228.3	576	438.5
70	357,911	343,000	1,058,610	865,184	779.2	459.0	427.2	672	696.8
80	531,441	512,000	1,574,640	1,290,552	1,380.9	779.9	731.6	768	1,040.4

表3.3 直接ソルバでの処理時間（マルチフロンタル法、MMDリナンバ）

n	節点数	要素数	未知自由度数	ディスク [Block]	合計時間 [Sec.]	ソルバ時 間[sec.]
10	1,331	1,000	3,630	2,580	2.3	2.0
15	4,096	3,375	11,520	8,574	30.6	29.7
20	9,261	8,000	26,460	20,169	198.9	196.8
25	17,576	15,625	50,700	39,945	881.5	876.3
30	29,791	27,000	86,490	67,657	3,032.2	3,022.5

表3.4 直接ソルバでの処理時間（FINAS固有ソルバ）

n	節点数	要素数	未知自由度数	ディスク [Block]	合計時間 [Sec.]	ソルバ時 間[sec.]
10	1,331	1,000	3,630	2,580	1.5	1.2
15	4,096	3,375	11,520	8,574	21.2	20.4
20	9,261	8,000	26,460	20,169	145.6	143.6
30	29,791	27,000	86,490	67,657	2,503.8	2,494.1
35	46,656	42,875	136,080	109,118	8,648.6	8,631.0

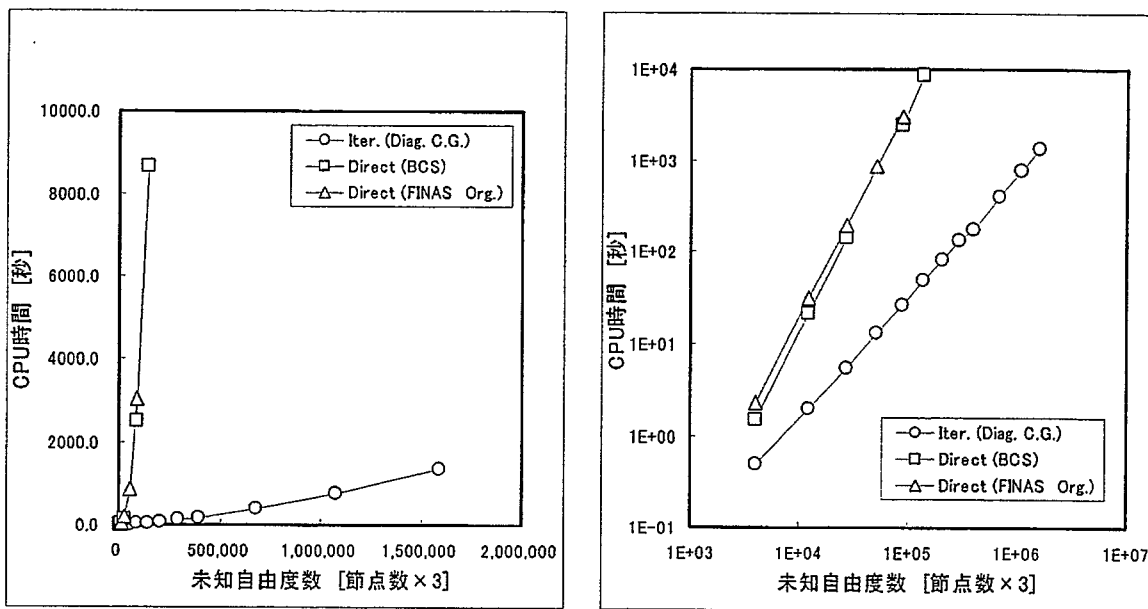


図3.3 全 CPU 時間の比較

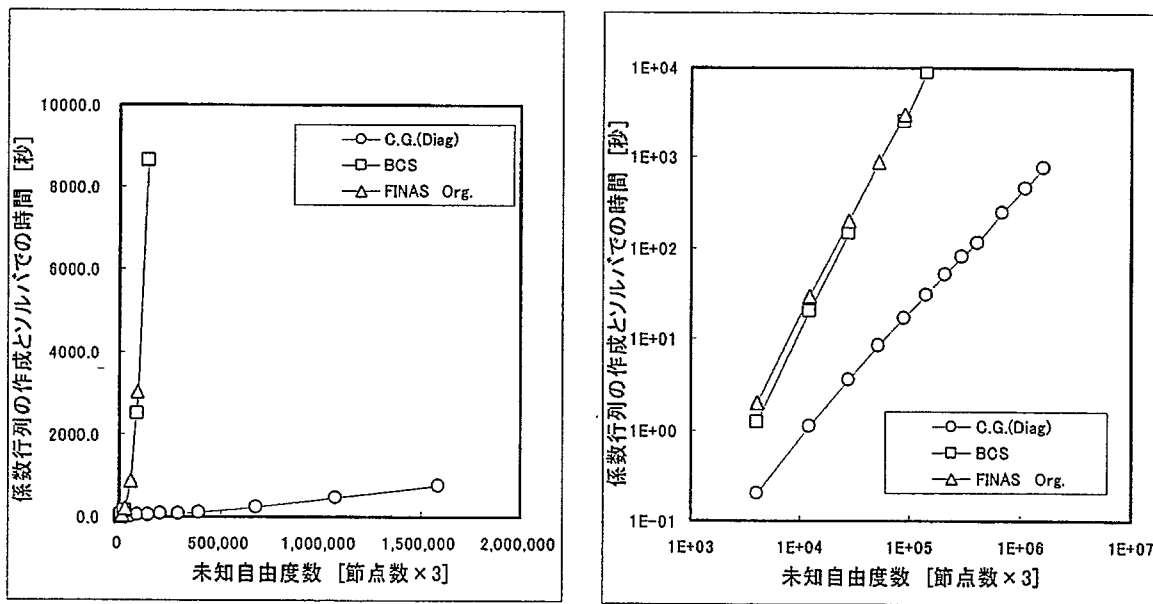


図3.4 ソルバでの処理時間の比較

3.1.1.4 大規模問題での留意点

今回のような大規模問題を解析するにあたっては、デフォルトでの FINAS の設定のままではいくつかの不都合が生じた。ここでは、これらの内容と対処方法についてまとめておく。

① 入力データの列不足 → WIDEDATA の指定

FINAS のデフォルトの入力データ形式では、整数型として 5 列が割当てられているため、10 万以上の節点、要素番号を定義できなくなる (99999 以上は書けないため)。

この対策は、V15.0 で整備された入力データの拡張書式を用いる。操作としては、入力データの始め (FINAS カードの前) に WIDEDATA カードを挿入する。

② FFCP バッファサイズの拡張 → SISTEM 1/8193 の指定

この問題は、処理系に依存する可能性があるが、 $n=50\sim 60$ (約 10 万～20 万節点) 以上では、以下のエラーが発生する。これは、ワークファイルのレコード数があふれたことを意味している。

```
forrtl: severe (25): record number outside range, unit 43, file C:\isoh\cub_comp2\WORK13
```

この対策としては、FFCP のバッファサイズの拡張が望ましい。デフォルトのバッファサイズは、1025 であるが、これを SYSTEM カードにより、例えば 8 倍の 8193 にする。なお、バッファサイズの値としては、2 のべき乗+1 が推奨されている。

以下に、①、②に対応した FINAS 入力データ例を示す。

```
WIDEDATA
SYSTEM    1/8193
FINAS
```

以下、FINAS の通常の入力データ

③ FFCP レコードカウントの問題→V16.0 の利用

さらに問題規模を大きくしていくと、 $n=80$ (約 50 万節点) を超えた処で、ファンクションモジュール EMSALL にて、以下のエラーが発生した。このエラーについては FFCP の

レコードカウント制限の問題が原因になっている可能性がある。この場合、挙動が予測できないため、エラーの出方も一定ではない。ただし、V16.0ではFFCPのレコードカウント部が16ビットから32ビットに拡張されており、この問題は発生しない。また、V16.0を用いるなら、②におけるSYSTEMカードの指定も不要である。

0*** SYSTEM FATAL MESSAGE 8063,

ELEMENT DOF TABLE INCONSISTANT WITH /ESTCOM/ -- NDOF= 24, NDOFX= 21

④ FNSEDITの問題→ポストから温度データを作成するプログラムを作成

現在、入力データの拡張書式(WIDE DATA)を用いた場合、リスタートファイルの動作が保証されていない。よって、従来行われてきたように熱伝導解析結果のリスタートファイルからFNSEDITにより、温度データを出力させることができない。しかし、節点温度の情報は、ポストファイルにも出力されており、ポストファイルを用いた方がファイル容量も少なく、容易にデータの作成が可能となる。そこで、今回、ポストファイルから温度データを出力するプログラムP2TEMPを作成し、温度データを作成した。

3.1.1.5 参考：CPUによる差

今回は、PCのCPUによる違いも検討してみた。この結果を表3.5に示すが、反復法のみの部分での反復法の比率は、P-4 2.53GHzとAthlon 1800+で2倍以上となる。一般の問題では、これほどの差は出ないため、P-4は反復ソルバに向いていると言えるかもしれない。

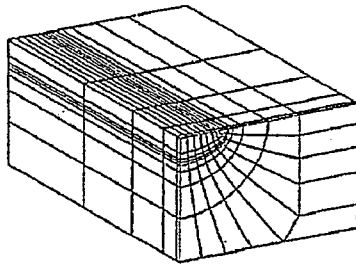
表3.5 反復ソルバでの処理時間

n	節点数	Athlon-XP 1800+ (1.54GHz)			Pentium-4 2.53GHz			Pen4とAthlonの比率
		合計時間[Sec.]	ソルバ[sec.]	反復法のみ[sec.]	合計時間[Sec.]	ソルバ[sec.]	反復法のみ[sec.]	
10	1,331	0.8	0.5	0.4	0.5	0.2	0.2	2.11
15	4,096	3.3	2.4	2.0	2.0	1.1	0.8	1.80
20	9,261	9.8	7.4	6.5	5.5	3.5	2.7	1.87
30	29,791	50.1	37.9	34.9	26.2	17.0	14.5	2.05
40	68,921	164.7	119.0	111.4	81.2	51.1	45.3	2.18
45	97,336	267.1	187.5	176.5	132.0	81.2	72.7	2.17

3.1.2 FINAS 実証例題 DP-14, DP-25 による性能評価

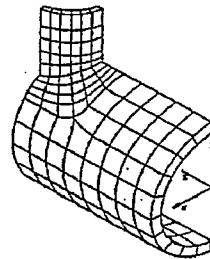
FINAS には、検証例題の他に、実証例題として、DP (Demonstration Problem) が整備されている。この中で、大規模問題として、使われてきたのが、DP-14 および DP-25 である。これらは、どちらも 20 節点六面体要素により 3 次元の弾性解析を行うもので、現在では、もう大規模とは言えないが、当時はそれなりに大規模な問題であった。例えば、CRAY X-MP/1 (235MFLOPS) では、DP-14 と DP-25 の解析に、各々 45.9 秒、10.7 秒を要している。

計算結果を、表 3.6, 7 に示す。問題規模が小さいせいか、反復法と直接法は同程度～直接法が有利といった処である。ここに、FINAS オリジナルとは、FINAS のデフォルトであるウェーブフロント法ソルバであり、BCS (MMD) は V14.0 で導入したボーイング社からのソルバ (マルチフロンタル法) に MMD (Multiple Minimum Degree) オーダリングを施したもの、BCS (Metis) は同じくボーイング社のソルバに Metis によるオーダリングを行ったものでどれも直接法である。また、不完全コレスキー分解による前処理 (IC) は、正定値性が保たれなくなり (消去過程で対角項が負になった) 破綻した。このため、不完全 LU 分解 (ILU (0), ILU と表記) による前処理と本来非対称行列に対する手法である、双共役勾配法 (BiCG 法) を用いている。



DP-14 三次元亀裂解析

要素数: 380 (HEX20), 節点数: 2,048, 未知自由度数: 5,815



DP-25 パイプ結合部解析

要素数: 125 (HEX20), 節点数: 1,013, 未知自由度数: 2,839

図3.5 FINAS 実証例題

表3.6 DP-14 での処理時間 (自由度数: 5,815)

手 法	合計時間[Sec.]	ソルバ [sec.]	ディスク [Block]	反復法の時間 [sec.]	反復数	メモリ [MB]
FINAS オリジナル	35.0	33.9	14,776			
BCS (MMD)	6.6	5.6	11,123			
BCS (Metis)	4.7	3.7	10,165			
C.G.法(ポイントヤコビ)	9.8	8.7	4,891	8.2	1,144	6.51
C.G.法(IC)				Break Down		
Bi-C.G.法(ILU)	7.7	6.6	4,891	6.2	165	16.02

表3.7 DP-25 での処理時間 (自由度数: 2,839)

手 法	合計時間[Sec.]	ソルバ [sec.]	ディスク [Block]	反復法の時間 [sec.]	反復数	メモリ [MB]
FINAS オリジナル	0.6	0.3	1,710			
BCS (MMD)	1.0	0.7	1,710			
BCS (Metis)	1.0	0.7	1,710			
C.G.法(ポイントヤコビ)	3.9	3.7	1,710	3.6	1,294	2.46
C.G.法(IC)				Break Down		
Bi-C.G.法(ILU)	1.7	1.4	1,710	1.3	86	6.04

3.2 反復法の性能評価-2

さらに実用的な問題において、組込んだソルバの性能評価をするため、球形管板の熱伝送および応力解析についての解析と性能評価を実施した。解析は、41,164 節点よりなる中規模モデルについて実施した。また、これらの計算には CPU として、Pentium-4 2.53MHz を持つ PC を用い、収束判定基準として、残差ノルムを 10^{-7} 以下とした。

3.2.1 球形管板中規模モデルによる解析

球形管板の中規模モデルは、4 万節点を超えるモデルではあるが、従来の FINAS によっても解析可能な規模である。この解析は、ソルバでの処理時間の概要を知る目的で行ったので、計算は 1 ステップのみとした。解析には、20 節点の六面体要素を用い、熱伝導解析としては、非定常、応力解析としては、熱弾性問題を扱っている。表 3.8 にモデルの概要を、図 3.6,7 にモデルの形状を示す。

まずは、当然のことであるが、反復法によっても直接法と同様の解が得られることを確認した。熱伝導解析の結果得られた温度分布を Metis アルゴリズムによるマルチフロントアル法 (BCS ソルバ) とポイントヤコビ前処理による CG 法を比較して図 3.8,9 に示す。図では、全く同じに見えるが、出力リストで見ると、温度の最終桁 (7 桁目) がたまに異なる程度の差しかない。これは、収束判定の基準値を 10^{-7} としているので妥当な精度と言える。次に、同様に熱応力解析の結果を図 3.10,11 に示す。こちらも、図では全く同様で、出力リストでも、変位の最終桁 (6 桁目) が 1~2 程度異なるだけであり、反復ソルバが正常に動作していることが確認された。

表 3.9 に熱伝導解析での計算時間を示す。表中の手法の記述は 3.1.2 節に同様である。まず、直接法では V14.0 で組込んだソルバ (ボーイングによるマルチフロントアル法) が、FINAS の実行速度を大幅に改善させていることが解る。また、リナンバーアルゴリズムでは MMD より Metis が優れている。反復法では、ポイントヤコビ前処理による CG 法が、直接法で最速である Metis アルゴリズムによるマルチフロントアル法より、ソルバの処理時間で倍以上速く、反復法を採用したことの効果が現れている。また、反復法でも、前処理として不完全コレスキー分解 (IC) や不完全 LU 分解を採用すると、最速であったポイントヤコビ前処理に較べて反復数は減少するが、処理時間としては増加してしまうという場合がある。いずれにせよ、この問題では、ポイントヤコビ前処理による CG と不完全 LU 分解による Bi-CG 法が Metis によるリナンバーアルゴリズムを採用したマルチ

フロントアル法より速く、反復法によって直接法の処理時間を改善できることを示した。

次に、表 3.10 には応力解析での計算時間を示す。応力解析でも、直接法ではマルチフロントアル法がソルバの性能を大幅に改善したことが解る。しかし、反復法では、ポイントヤコビ前処理のCG法においてもソルバにおける処理時間が、Metis リナンバのマルチフロントアル法に較べて倍程度の計算時間を要していることが解る。このように、反復法がそれほどの性能を発揮できないのは、応力解析では、熱伝導解析に較べて収束性に問題があるためと考えられる。これは、同じポイントヤコビ前処理において、熱伝導での反復回数、464 に較べ、応力解析では、7,087 と 17 倍程度に増加している。この反復数の増加は、未知数の増加量である 3 倍に較べて極端に大きい(3.1.1.3 節によれば、同系列での反復回数の増加は、未知数の 1/3 乗に比例するので、未知数が 3 倍で反復回数は高々 1.5 倍程度と推定できる)。熱伝導解析は本来、ポアソン方程式を解く問題であるために、対角優位でポイントヤコビ法の前処理が良く効くが、構造の問題はこれより収束性が劣ることが知られている。これほどの収束回数の増加があれば、反復法の利点が生かされておらず、今後、他の前処理手法を試行することを含めて前処理方法を工夫する必要がある。

表3.8 球形管板中規模モデルの概要

項 目		規 模
節点数		41,164
要素数	熱伝導解析	10,904
	応力解析	8,640
自由度数	熱伝導解析	41,164
	応力解析	123,492
解析ステップ数	熱伝導解析	1
	応力解析	1

表3.9 球形管板中規模モデル熱伝導解析（自由度数：41,164、1ステップ）

手 法	合計時間 [Sec.]	ソルバ [sec.]	ディスク [Block]	反復法 の時間 [sec.]	反復数	メモリ [MB]
FINAS オリジナル	791.2	784.7	5,788			
BCS (MMD)	67.5	61.2	5,788			
BCS (Metis)	32.8	26.5	5,788			
C.G.法(前処理無し)	51.6	45.1	5,788	43.5	1,873	18.7
C.G.法(ポイントヤコビ)	18.8	12.4	5,788	10.7	464	18.7
C.G.法(IC)	67.1	60.8	5,788	59.1	94	31.2
Bi-C.G.法(ILU)	17.8	11.5	5,788	9.7	94	45.0

表3.10 球形管板中規模モデル応力解析（自由度数：123,492、1ステップ）

手 法	合計時間 [Sec.]	ソルバ [sec.]	ディスク [Block]	反復法 の時間 [sec.]	反復数	メモリ [MB]
FINAS オリジナル	19,759.7	19,735.5	102,283			
BCS (MMD)	1,744.3	1,720.1	102,283			
BCS (Metis)	567.9	544.0	102,283			
C.G.法(前処理無し)	3,047.5	3,023.5	102,283	3,014.5	17,769	148.3
C.G.法(ポイントヤコビ)	1,238.6	1,214.9	102,283	1,205.7	7,087	148.3
C.G.法(IC)	1,681.3	1,657.3	102,283	1,648.3	2,355	255.2
Bi-C.G.法(ILU)	2,483.0	2,459.3	102,283	2,450.3	3,183	365.8

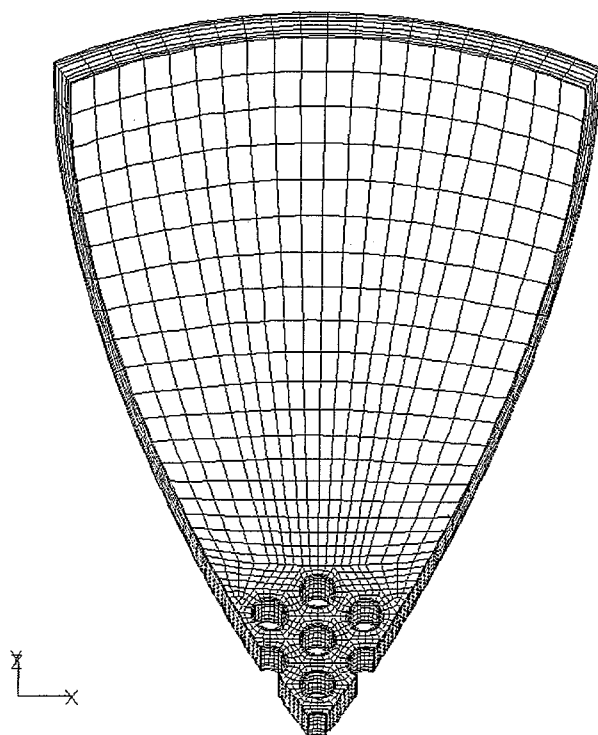


図 3.6 球形管板中規模モデル 全体形状

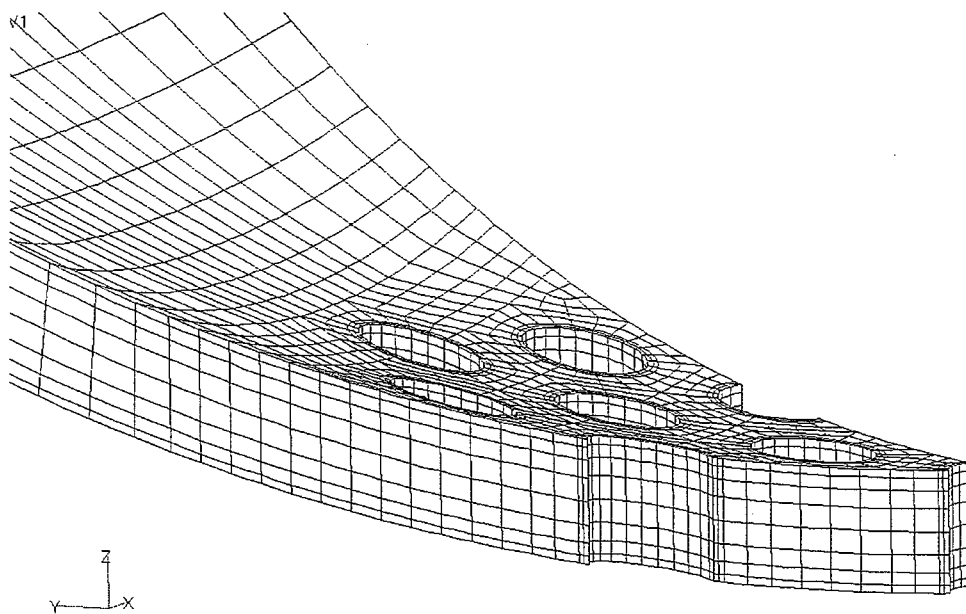


図 3.7 球形管板中規模モデル 管部詳細

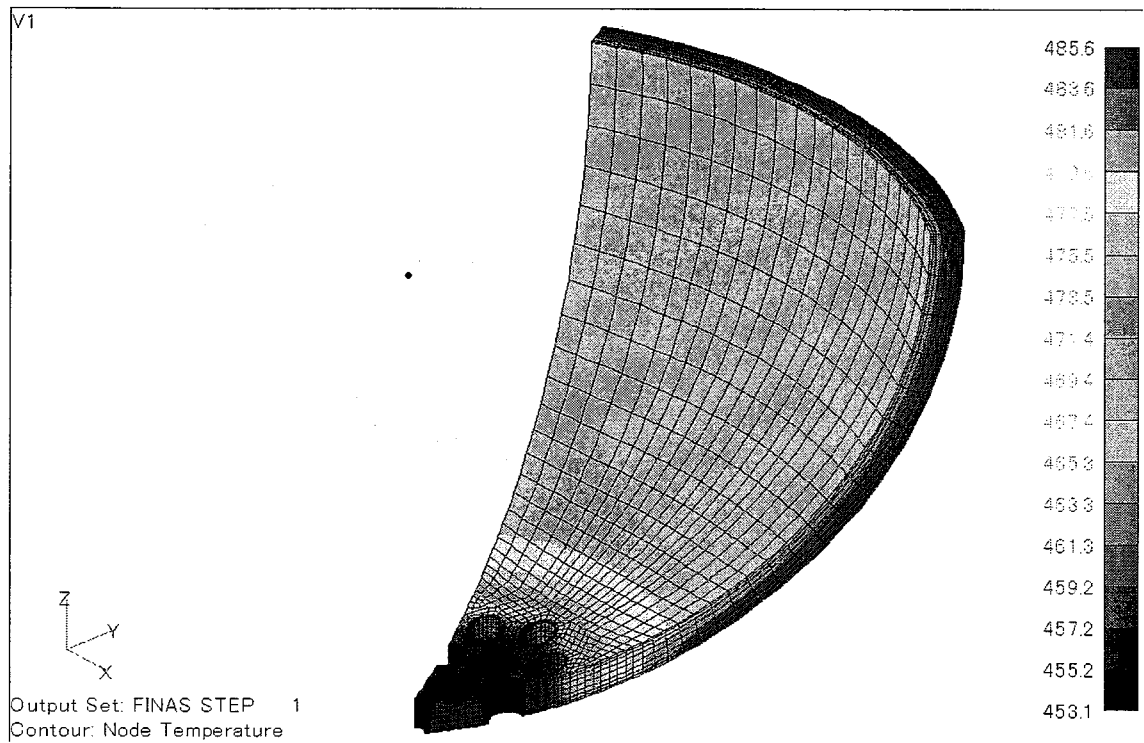


図 3.8 球形管板中規模モデル 熱伝導解析結果(BCS-Metis, JNC による)

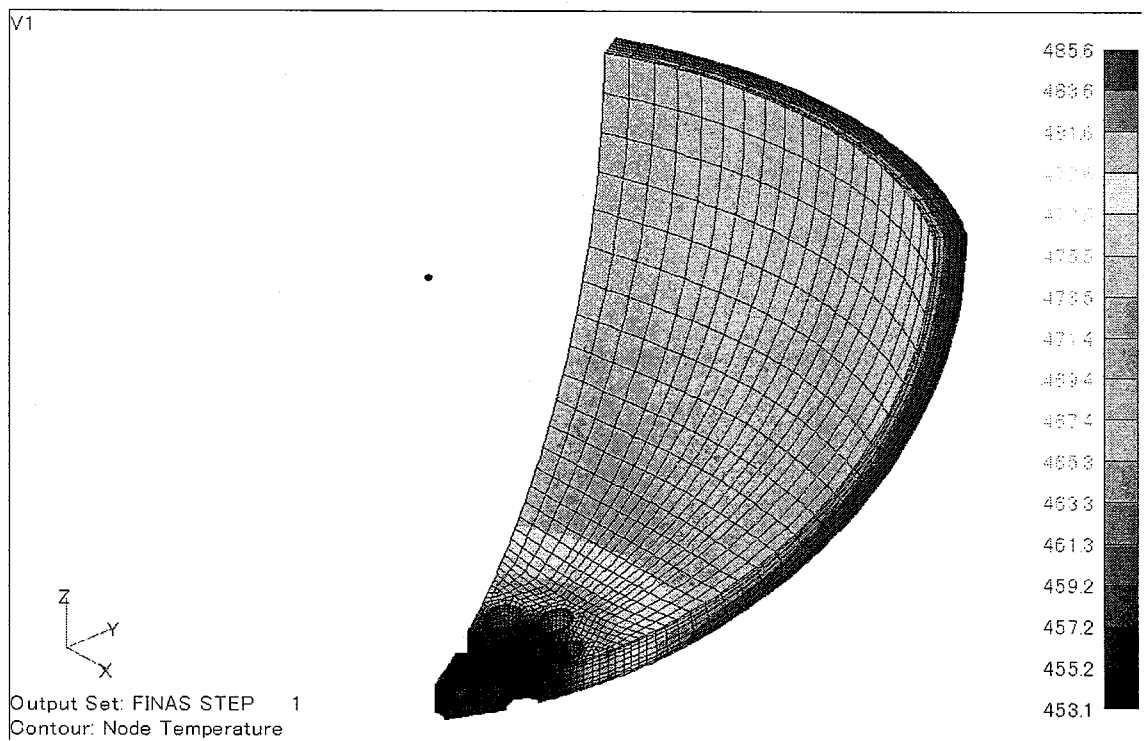


図 3.9 球形管板中規模モデル 熱伝導解析結果(ポイントヤコビ-CG 法)

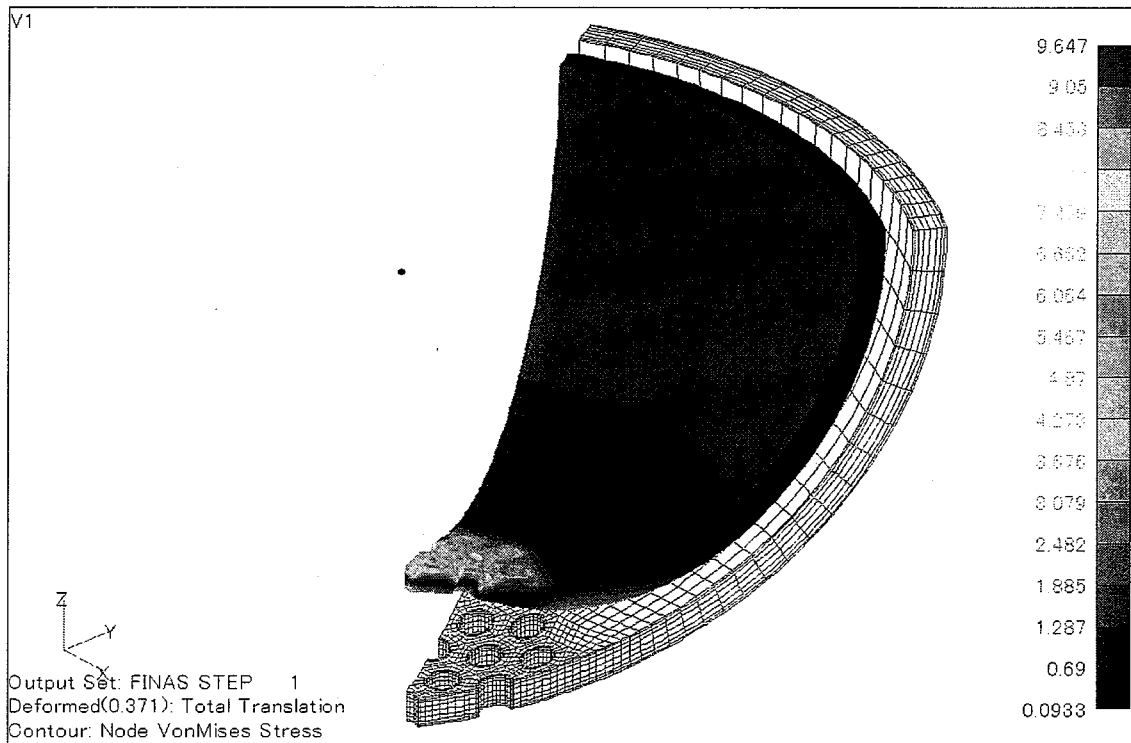


図 3.1 0 球形管板中規模モデル 熱応力解析結果(BCS-Metis)

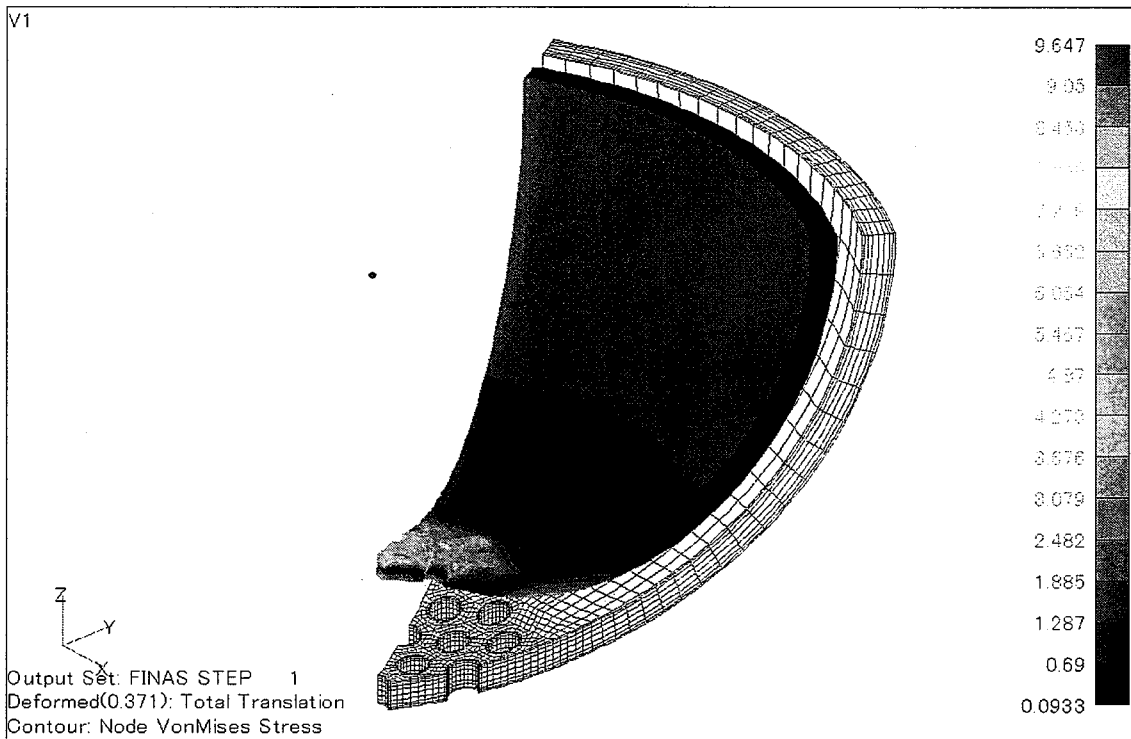


図 3.1 1 球形管板中規模モデル 熱応力解析結果(ポイントヤコビ-CG 法)

4. 直管型球形管板の3次元有限要素解析

4.1 概要

部分孔開きモデルと反復ソルバによる3次元構造解析法の適用性を調べるため、実用化戦略調査研究で検討されている直管型球形管板を対象とした熱過渡解析を行う。直管型SGの全体図を図4.1に、上部管板部（SG出口側）を図4.2に示す。直管型球形管板は、球殻に多数の伝熱管が鉛直に差し込まれた形状を有しており、それらは正三角配列に並ぶ。そこで、従来の管板解析法は、多孔部を等価中実板に置き換えて平均的な応力を求め、それに集中係数を乗じることで孔縁のピーク応力を求めるものであった[2]。しかし、直管型球形管板では外縁の孔周りの異方性が強いことから、従来法では評価精度が低下するため、孔廻りの詳細形状を考慮した3次元解析が望まれる。本計算では、運転中に生じる熱過渡時に応力が集中すると想定される伝熱管部分を有するSG出口側にスポットをあて、管板の一部を3次元ソリッド要素でモデル化し、手動トリップを想定した熱過渡弾性解析を行った。

4.2 計算方法

直管型球形管板の形状の対称性を考慮すると中心角30度のセクタモデルで解析できる。さらに数千本の伝熱管群のすべてをモデル化すると、現実的に不可能な計算規模になるため、中央と外側の孔のみを詳細にモデル化し、その中間的挙動を示すと考えられる両者に囲まれた領域を平均的に等価な中実板で置き換えた部分孔開モデルを作成した。また、詳細モデル化した伝熱管群は、等価領域や胴部の境界の影響をなくすため孔の配置は3列分設けた。解析に用いた有限要素モデルを図4.3～4.5に示す。計算には大規模解析を可能とする反復ソルバを適用したFINAS Ver. 16を使用し、20節点の六面体2次要素と8節点の熱伝達要素を用いた。メッシュモデルの規模を表4.1に示す。また、3次元モデルによる熱伝導解析、および熱応力解析の固有の解析手法については以下の節に記す。

4.2.1 熱伝導解析

負荷条件として手動トリップを想定し、その負荷履歴を図4.6に示す。熱伝導解析において、管板内面、および伝熱管内面に熱伝達要素を介して構造物に熱過渡を与える。また、等価物性を用いた領域には等価な熱容量値と内部発熱を与える。ここで、等価物性領域で与える内部発熱量の算出方法を記す。まず初めに六角柱で考えると複雑になる

ので、簡易的に六角柱に孔の開いた伝熱管を含む構造体を孔あき円筒モデルに、等価領域を円筒モデルの孔を除いた円柱モデルにそれぞれ置き換える。それから円筒モデルのほうを軸対称モデルで作成し、伝熱管内部に相当する部分に熱伝達率を介して熱過渡温度履歴を与えると、流体から構造物に蓄積された単位体積当りの熱量 Q が得られる。円筒モデルに伝わる全熱量 $Q V$ と円柱モデルに存在する全熱量 $Q' V'$ は同じであるので、円柱モデルに発生する単位体積あたりの熱量 Q' は $Q (V/V')$ で表される。熱伝導解析において、ここで得られた Q' を内部発熱として等価領域に与える。さらに、材料物性の面から見ると等価モデルは母材と比較して熱容量が変化する。等価モデルの熱容量は母材の熱容量(= (比熱) × (密度))の体積比、すなわち等価領域の比熱、もしくは密度と体積比の積を用いると等価領域での熱容量を表現することができる(図4.7)。上記のことを考慮し、母材部と等価領域の熱伝導解析に用いた物性値を表4.2に示す。熱伝達率は管板内表面に $460\text{W/m}^2\cdot\text{K}$ 、伝熱管内表面にあたる管板の孔の内側には $6,640\text{W/m}^2\cdot\text{K}$ を用いた(表4.3)。

4.2.2 熱応力解析

熱応力解析では、管板内面、伝熱管内面に 17.25MPa の一定な内圧を負荷し、熱伝導解析で得られた温度履歴を与えた。拘束条件はモデルの対称性を考慮した。解析に用いた負荷条件、境界条件を図4.8,9に示す。

物性値は12Cr鋼を温度依存性を考慮しない 500°C 一定の値を用いた。孔部の機械的性質はASME Sec III A-8000[2](図4.10)の $h/P=0.5$ の値を用いると、母材部とのヤング率比0.51、ポアソン比0.3となる。解析に用いた材料物性値を表4.3,4に示す。

4.3 解析結果

4.3.1 熱伝導解析結果

熱伝導解析結果の温度コンター図をみると、手動トリップによって伝熱管部分が急激な温度降下があるのが分かる。また伝熱管以外の胴の部分は内表面だけ温度降下が表れるが、板厚内部にはほとんど伝播しない。また、伝熱管の部分と胴にあたる部分の境界に急激な温度勾配が発生している。

4.3.2 熱応力解析結果

熱応力解析結果を見ると、最大応力が発生する時点である図4.18~22から、伝熱管の管板板厚方向の応力分布に注目すると、管板の外表面から内表面に向かって応力が高

くなる傾向がある。また、伝熱管の管板中央部は同心円状に応力が分布しているが、外縁部は断面が楕円形状になることから、伝熱管孔部分の中央に向かう部分が特に鋭角になることから高い応力集中が発生しているのが分かる。総じて手動トリップ時に直管型球形管板に発生する熱過渡応力は、管板の外面より内面に、中央部より外縁部に高い応力が発生しやすいということが分かった。また図 4.23~28 から、全体的な挙動は伝熱管の部分の急激な温度低下に伴って、この部分が内側に向かって収縮する変形が生じる。

4.4 計算速度

計算に使用した PC は Pentium-4 3.06GHz の CPU を用いている。また、Intel E7205 チップセットで、メモリバンド幅は 4.2GB/sec である。ただし、OS は Windows-2000 とし、Hyper-threading 機能は OFF の状態での計測結果である (Hyper-threading を ON にすると 6%程度速度が低下するため)。

球形管板の解析では前章の検討より、直接法では Metis アルゴリズムによるリナンバを用いたボーイングによるマルチフロントル法 (BCS)、反復法ではポイントヤコビ前処理を用いた CG 法が最も高速であることが解っている。よってこの解析では、これら 2 つの方法により、解析時間等を比較する。

まず、両者による 19 ステップの解析後に得られた温度分布を図 4.29, 30 に示す。当然なら、直接法と反復法の結果は図面上全く同じであり、出力リスト上でも反復法の収束判定基準 (10^{-7}) 以上の有意な差はみられない。解析に要した処理時間については、表 4.5 に直接法、反復法を比較して示す。反復法 (ポイントヤコビ前処理-CG 法) は直接法 (Metis リナンバ-マルチフロントル法 (BCS)) に比べ、ソルバで 4 倍以上速く、合計時間でも 3.5 倍程度速い。ただし、反復法の収束状況 (表 4.6) を見ると、始めは 257 反復で収束しているものが、最終的には 390 反復と約 5 割程度反復回数が増加している。このようなことが起こる理由は、温度分布の履歴から明らかなように、始めは熱があまり伝わってが、時間が経過するに従って複雑な温度分布が形成されるためと思われる。なお、本計算においては、熱容量マトリックスとして FINAS の規定値である整合熱容量マトリックスを用いた。

前章同様、熱応力解析では、反復法の収束性に問題があって反復数が極端に大きくなるため、直接法に比較して約 2 倍の処理時間を要し、反復法の優位を示すことができなかった。また、表 4.7 には CPU 時間を示しているが、経過時間としては、直接法が 127,548

秒、反復法が 259,876 秒で、約 6 割程度の増加となる。

このように、今回は応力解析においては大規模問題で反復法の優位性を示すことができなかったが、この原因が収束性にあることにより、前処理の工夫によって改善できるものとする。

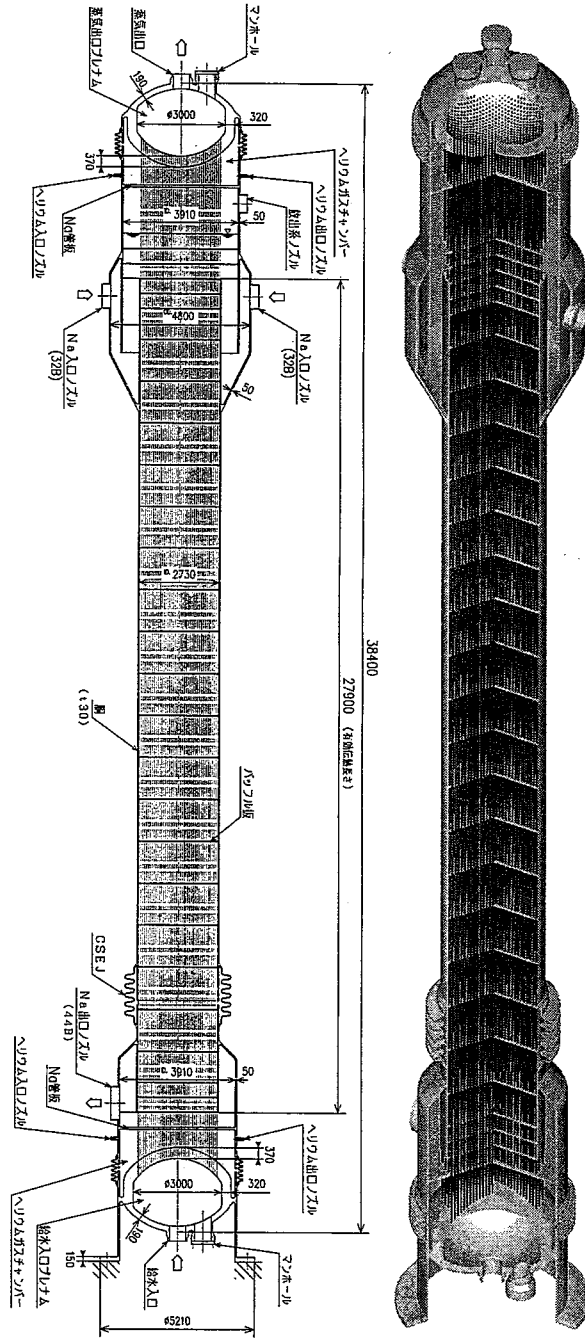


図 4.1 直管型SG(全体図)

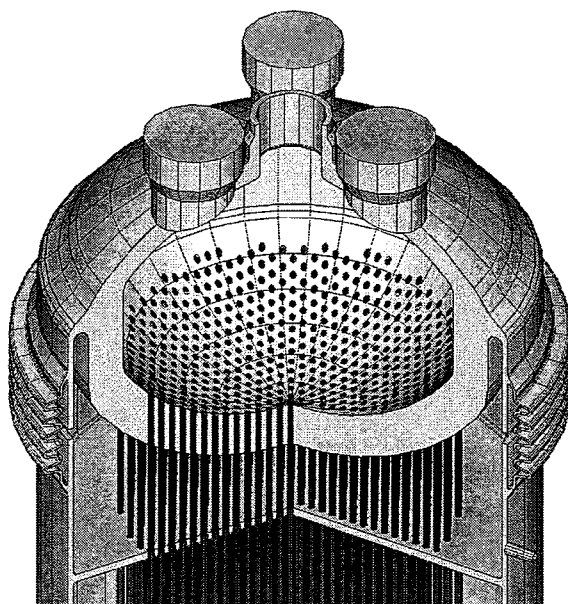


図 4.2 直管型SG(出口側管板)

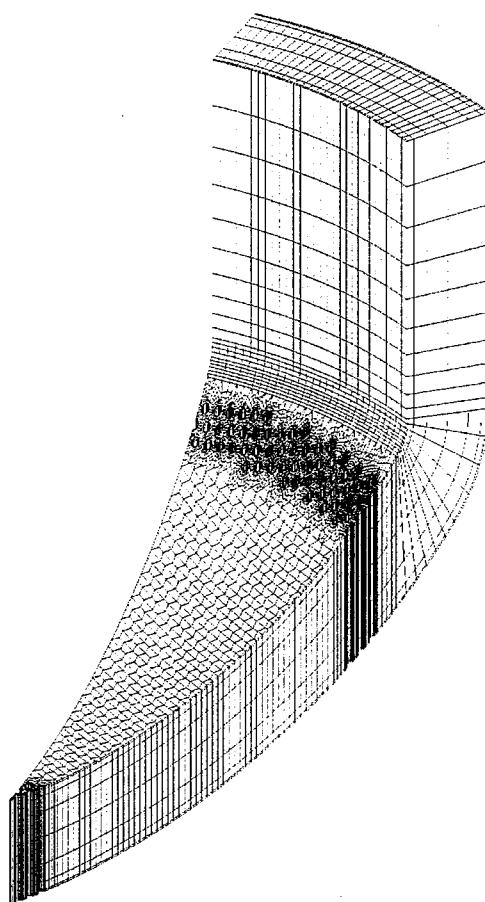


図 4.3 直管型SG球形管板メッシュ図(全体)

V1
L2

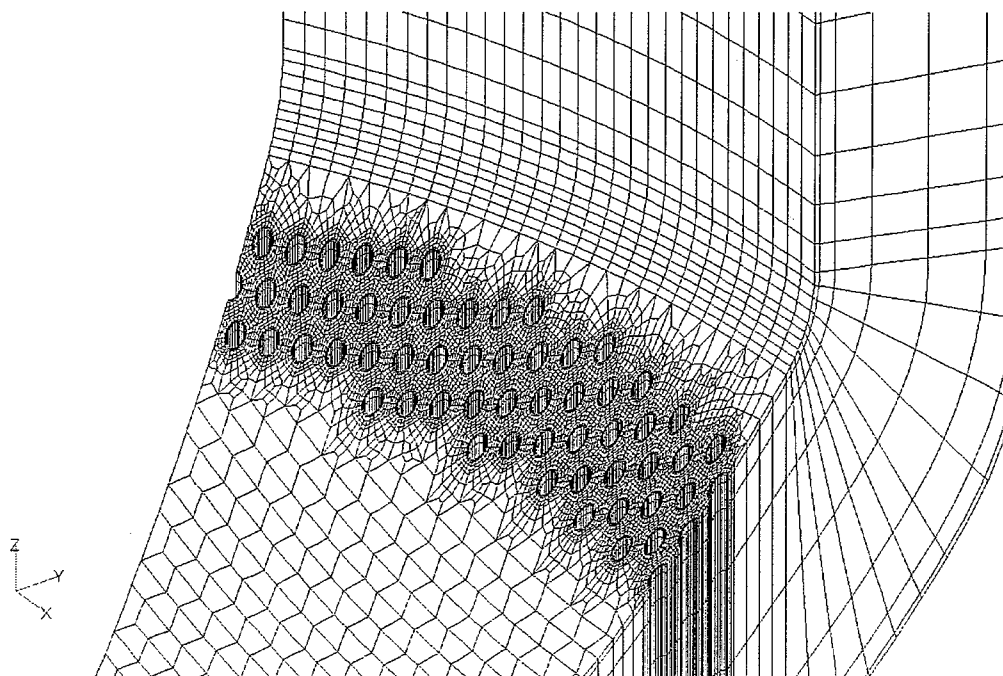


図 4.4 直管型SG球形管板メッシュ図(最外縁部詳細)

V1
L2

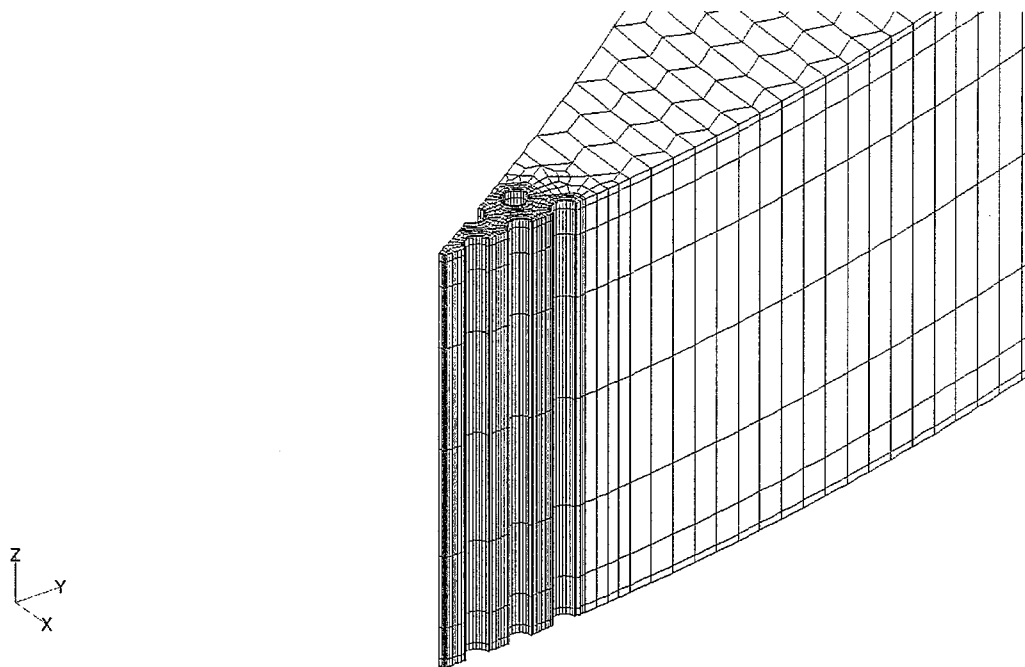


図 4.5 直管型SG球形管板メッシュ図(中央部詳細)

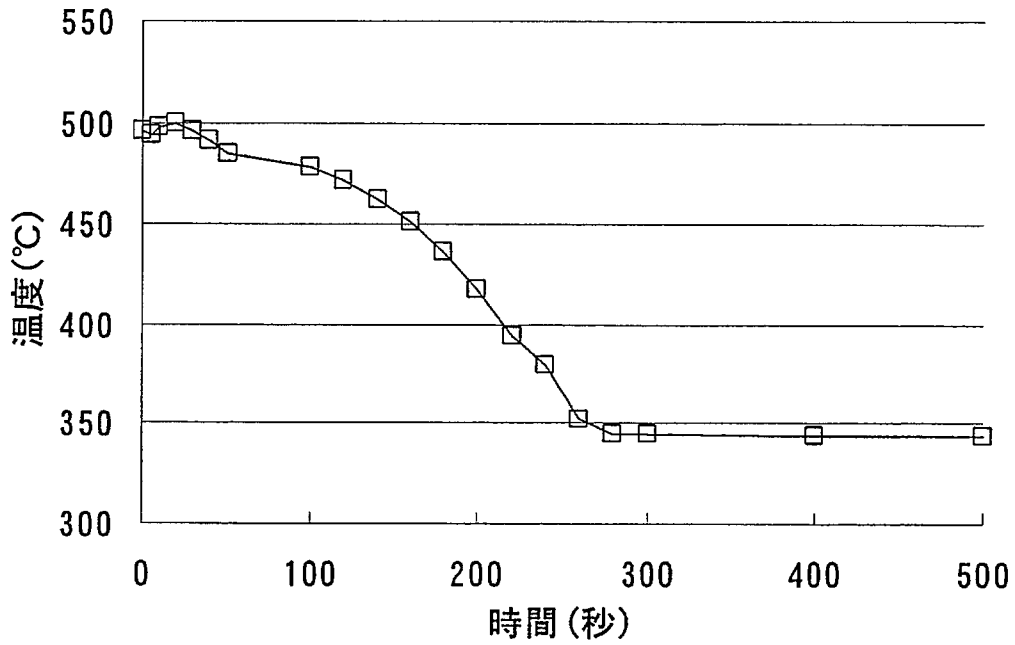


図4.6 SG出口蒸気温度履歴(手動トリップ)

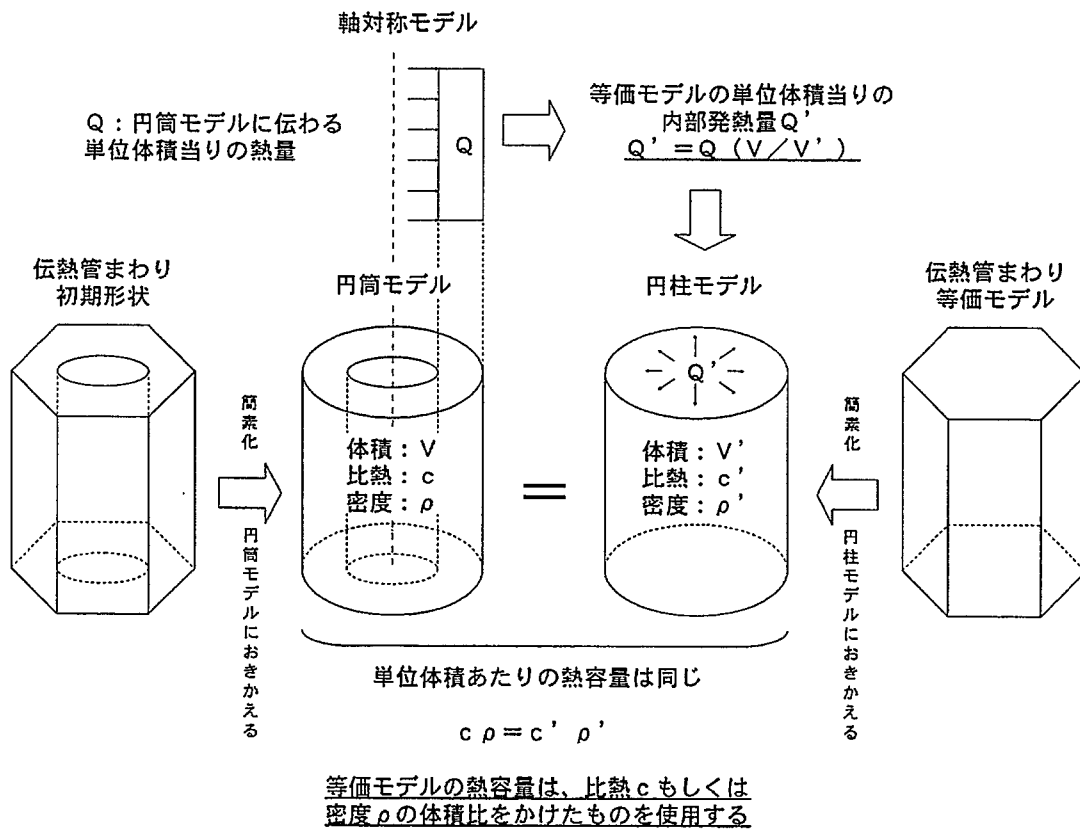


図4.7 等価物性領域の内部発熱量算出法

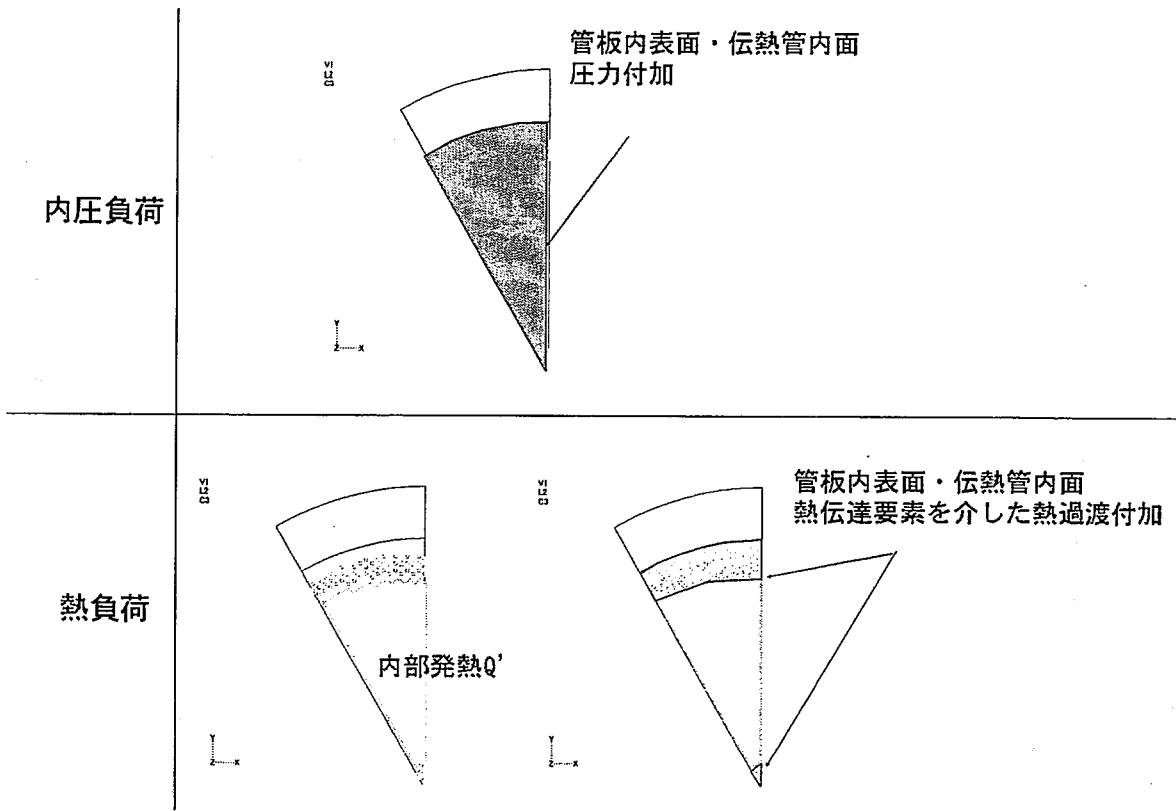


図4.8 負荷条件

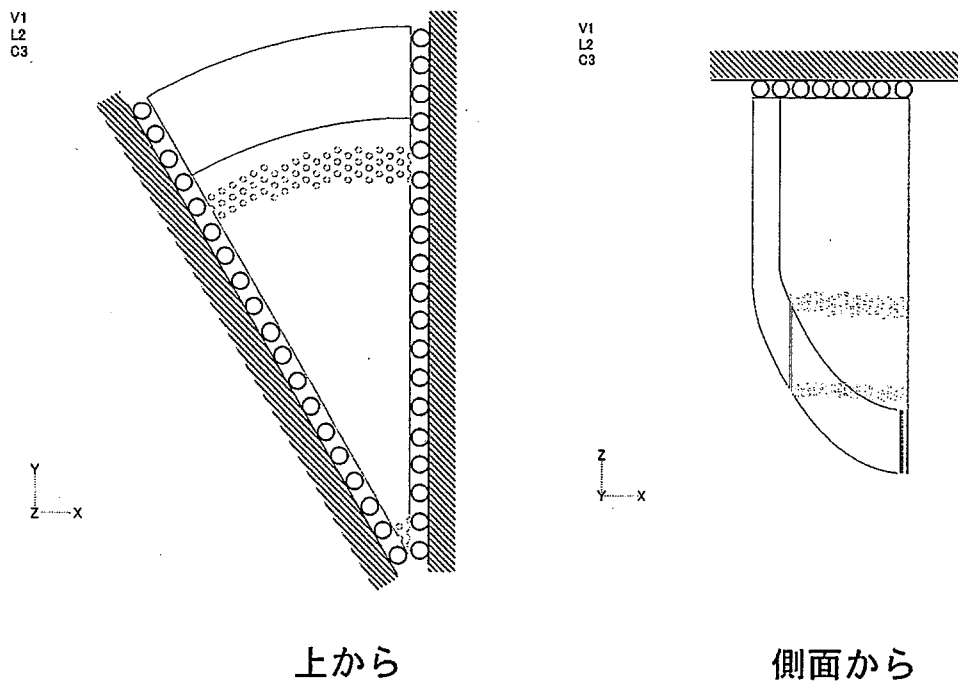


図4.9 拘束条件

Fig. A-8131-1

2001 SECTION III, DIVISION 1 — APPENDICES

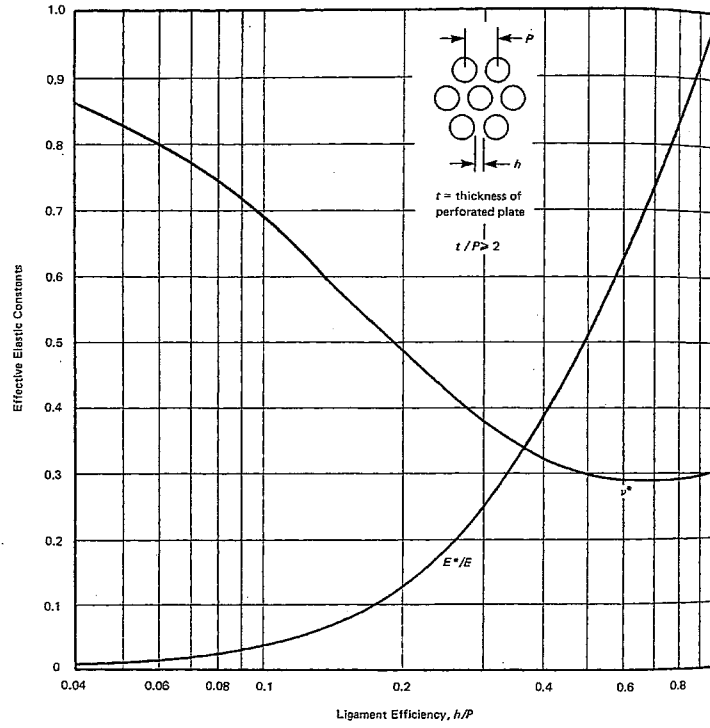


FIG. A-8131-1

図 4.1 0 リガメント効率による等価物性値 (ASME Sec III A-8000)

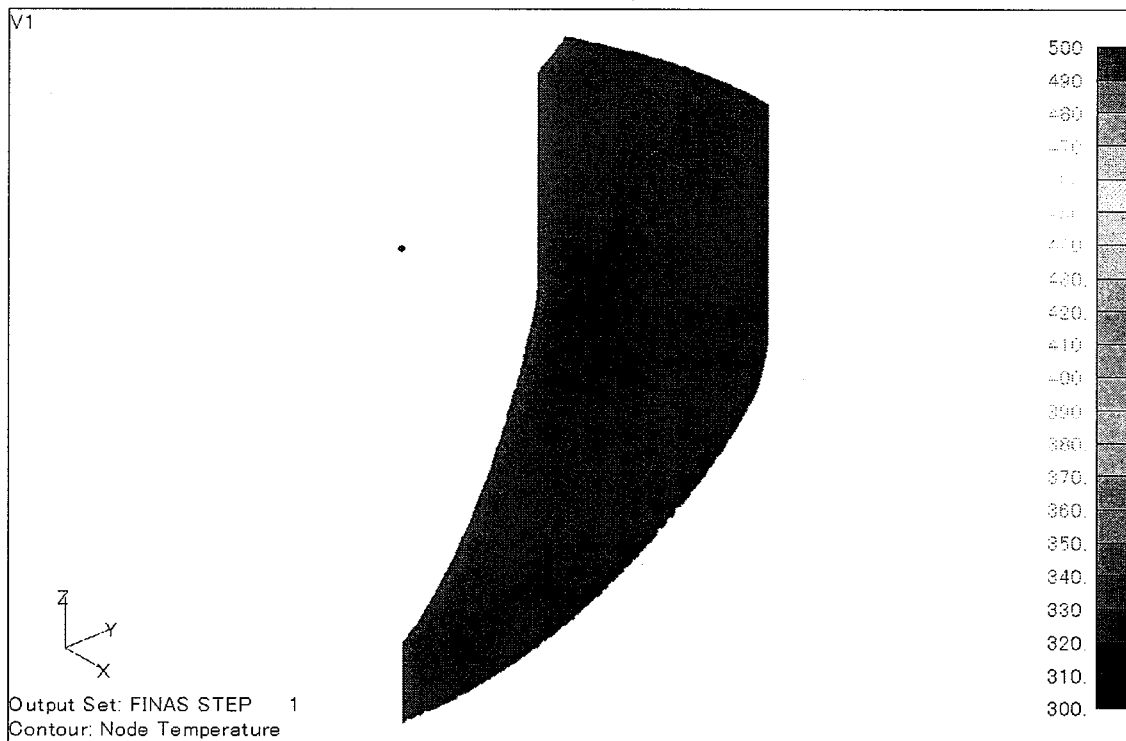


図 4.1 1 温度分布 (t=0 秒)

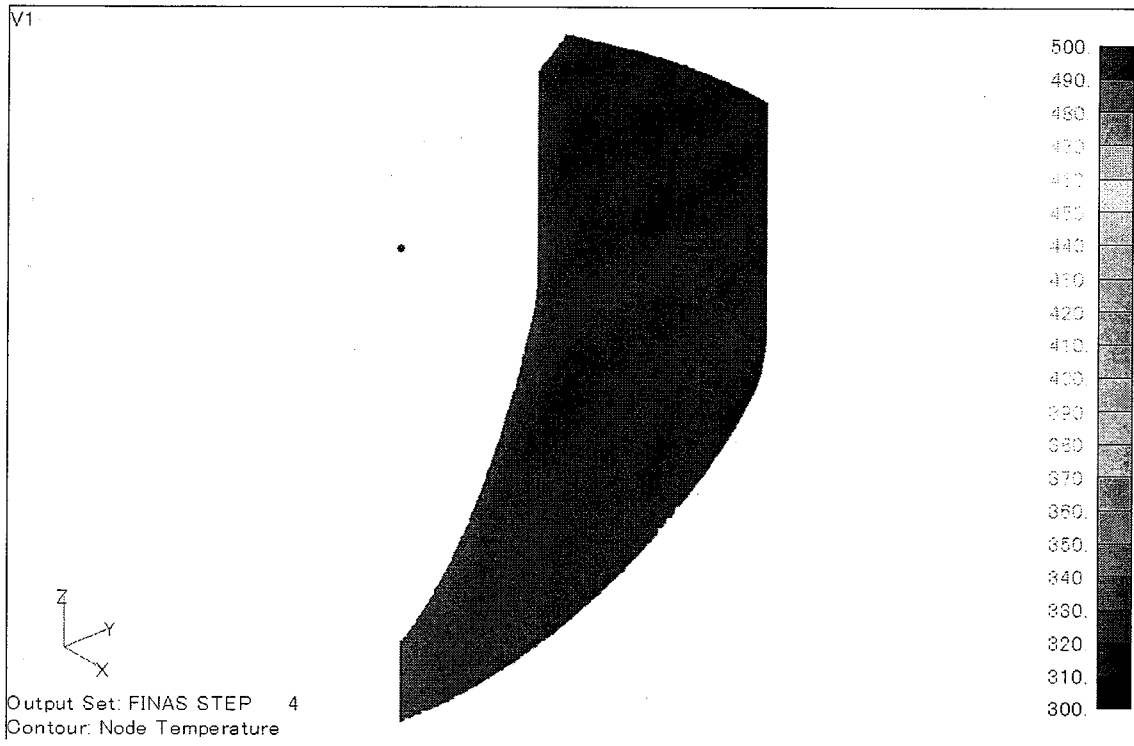


图 4.1 2 温度分布(t=30 秒)

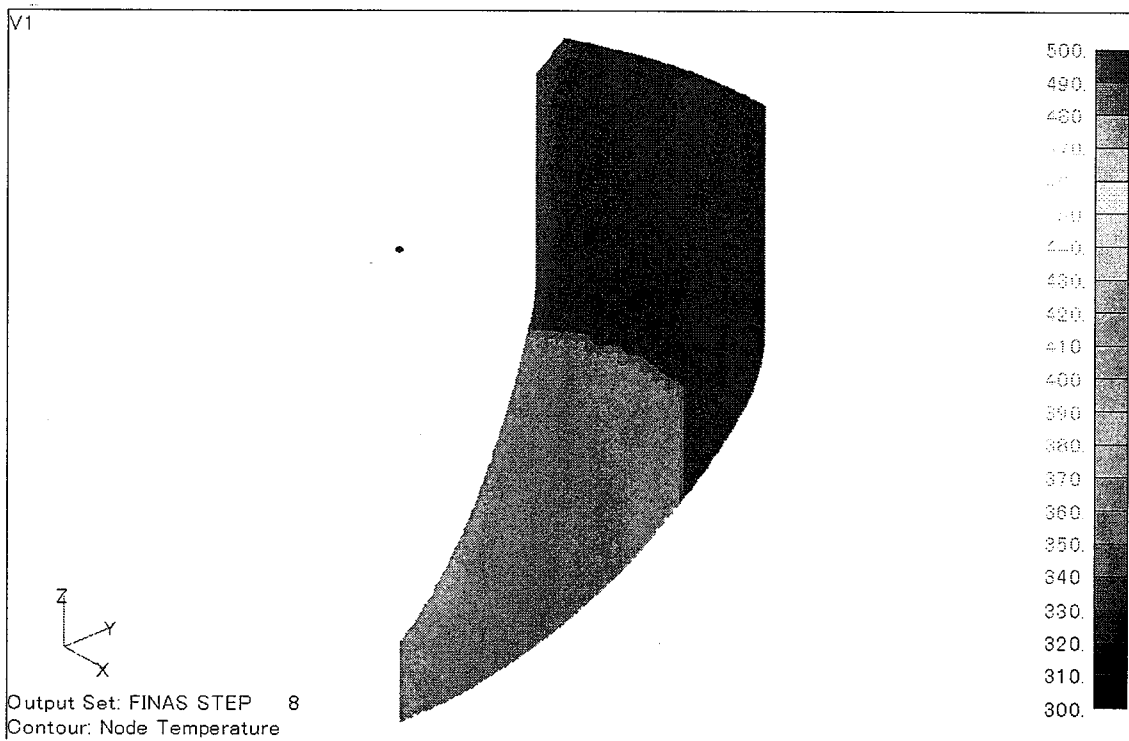


图 4.1 3 温度分布(t=120 秒)

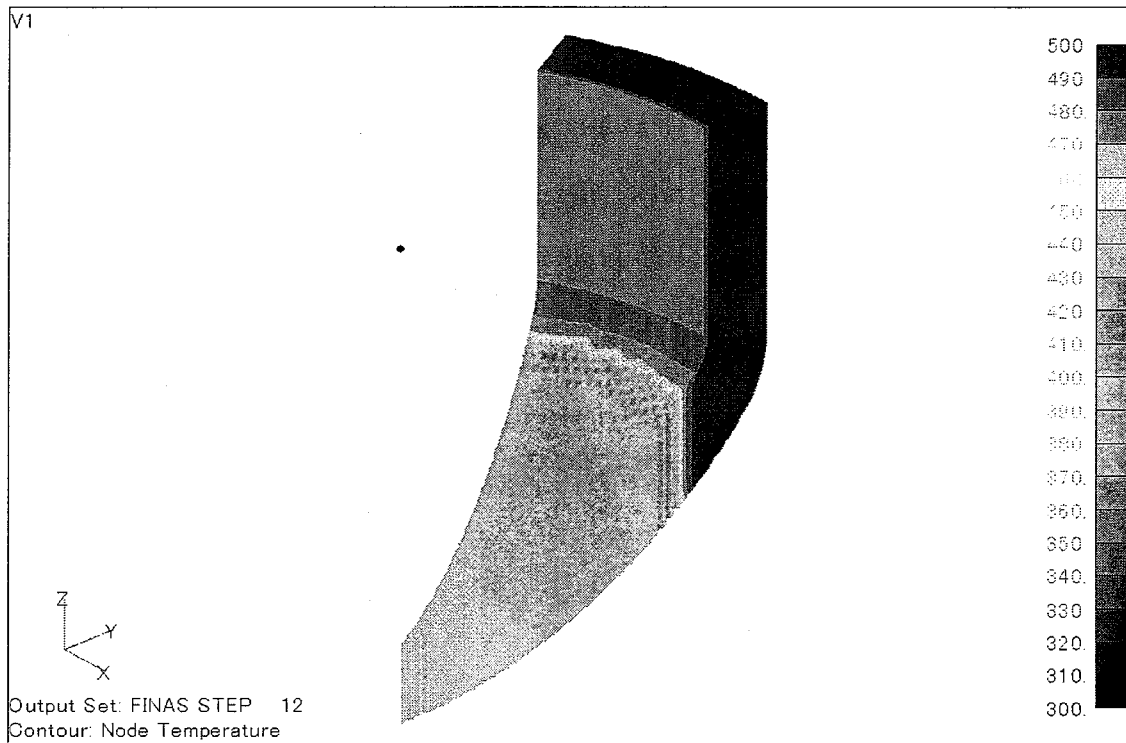


图 4.1 4 温度分布(t=200 秒)

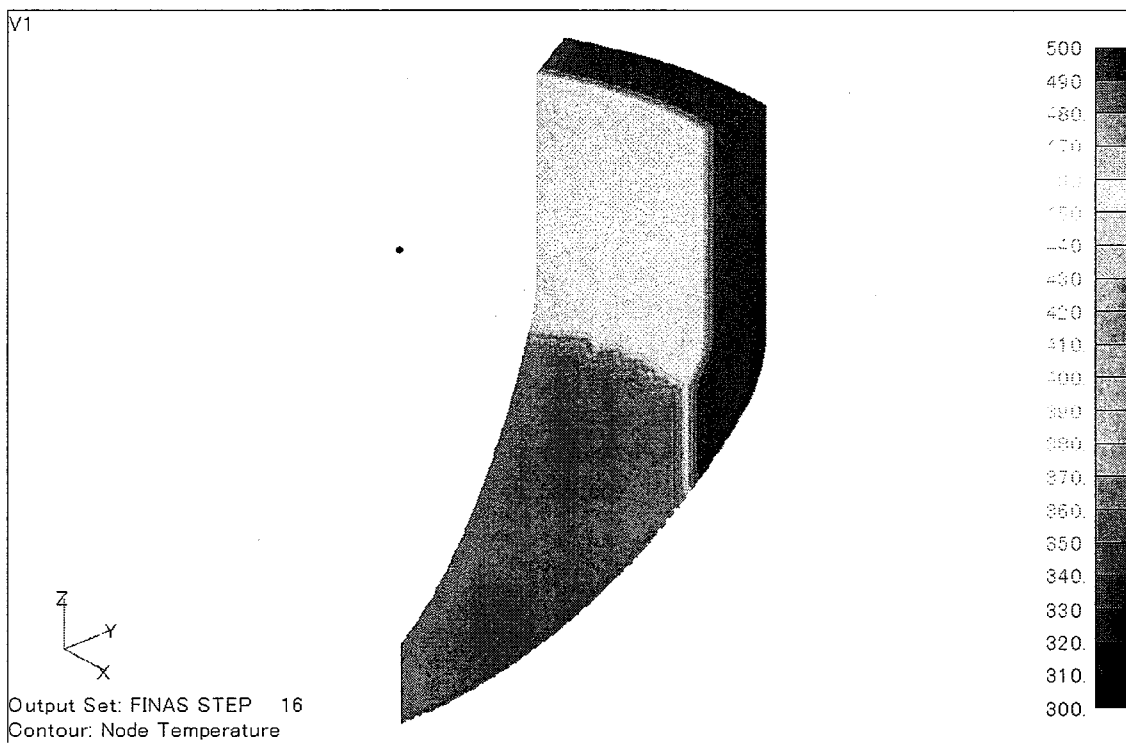


图 4.1 5 温度分布(t=280 秒)

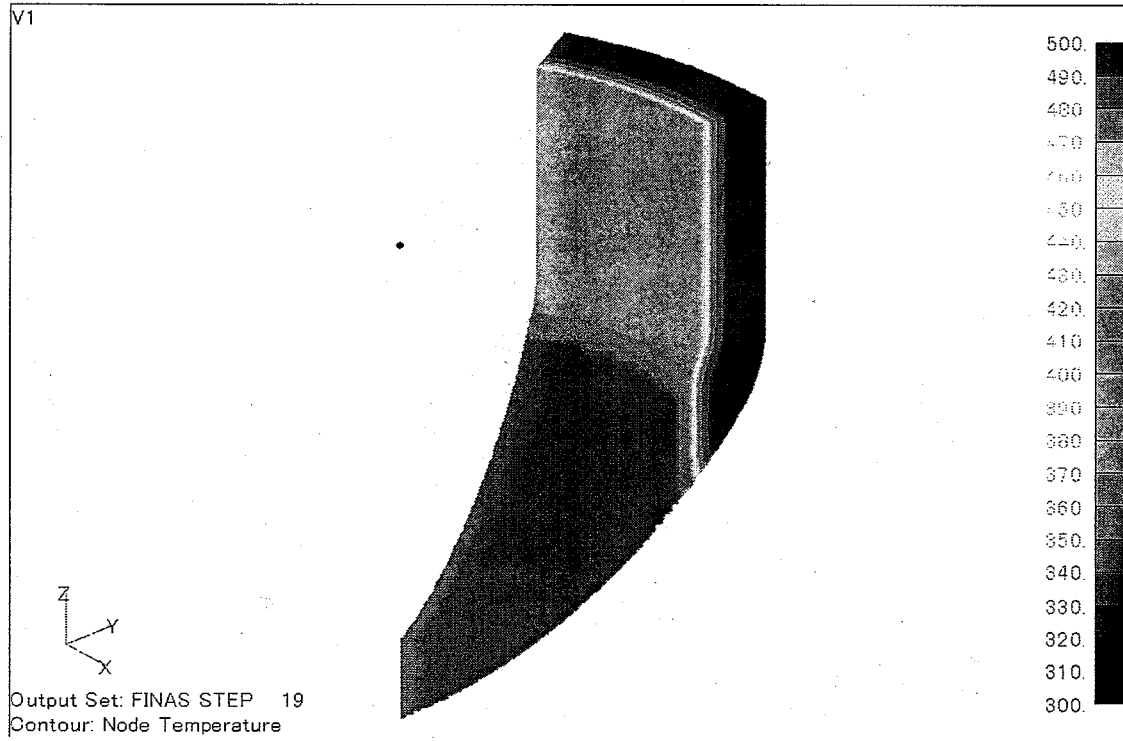


図 4.1 6 温度分布 (t=500 秒)

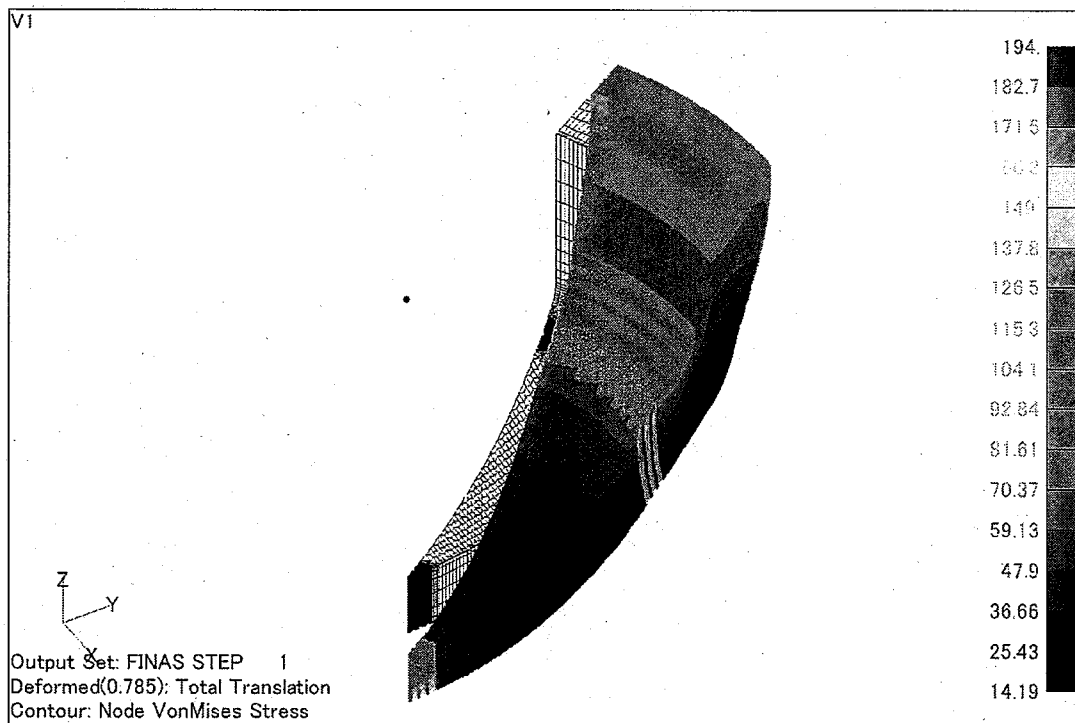


図 4.1 7 相当応力分布 (内圧負荷のみ、t=0 秒)

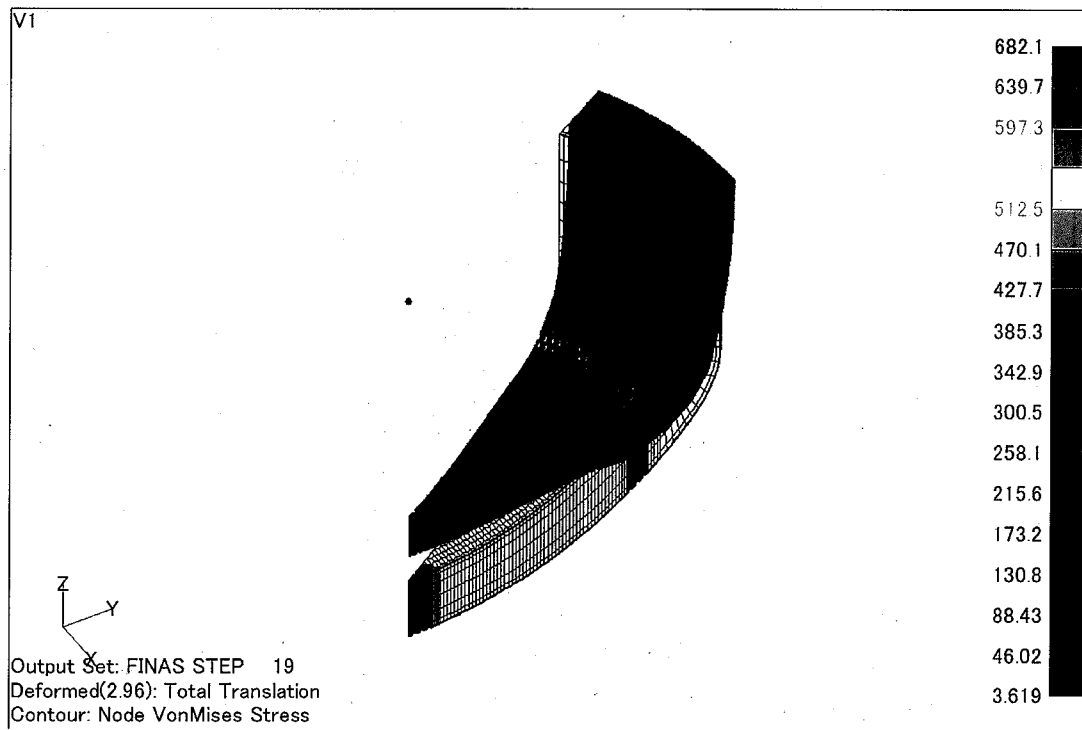


图4. 18 相当応力分布(最大応力発生時点、t=400 秒)

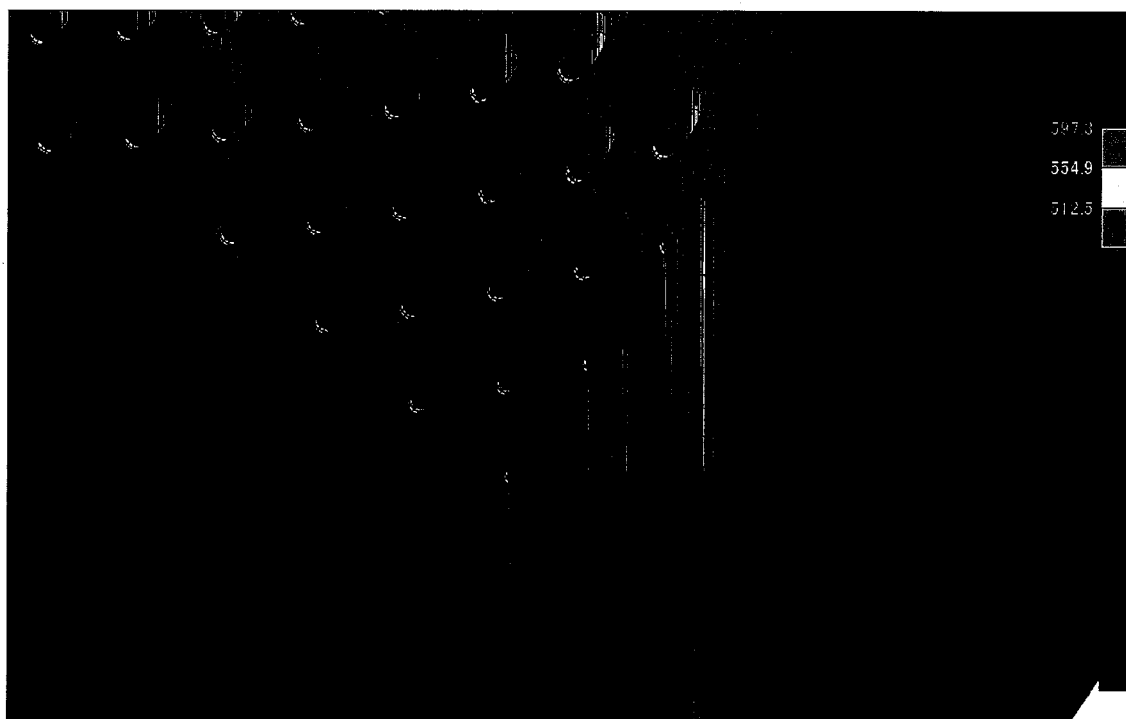


图4. 19 外縁部の相当応力分布(1)(最大応力発生時点、t=400 秒)

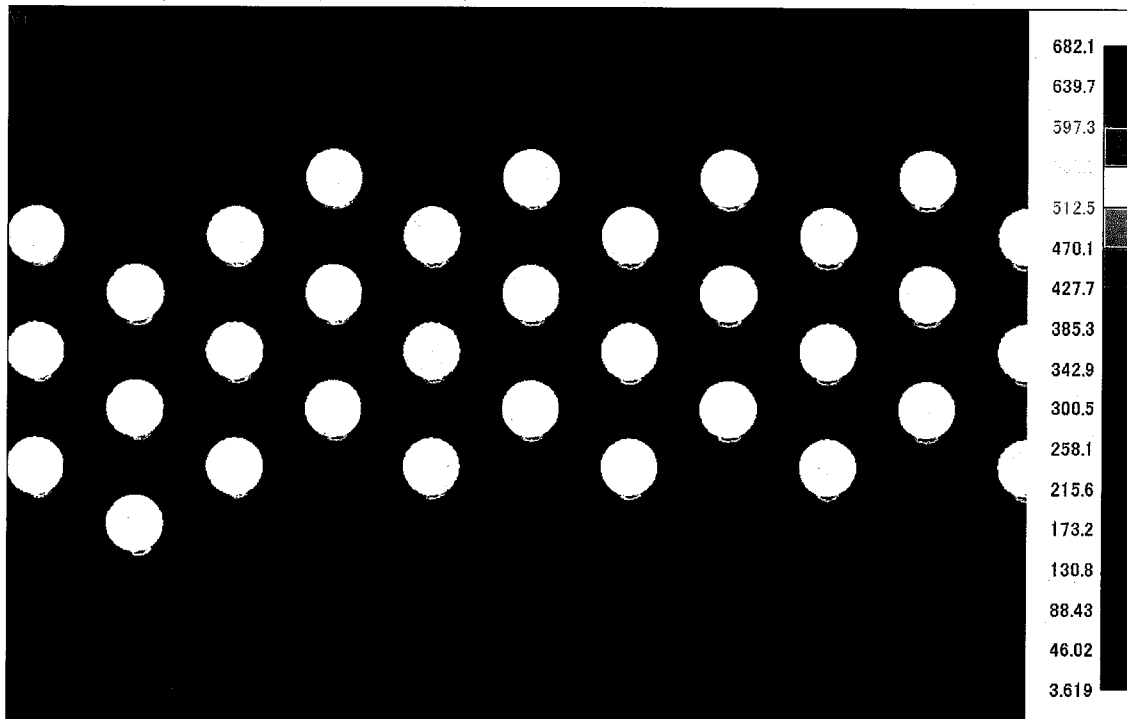


図4. 20 外縁部の相当応力分布(2)(最大応力発生時点、t=400 秒)

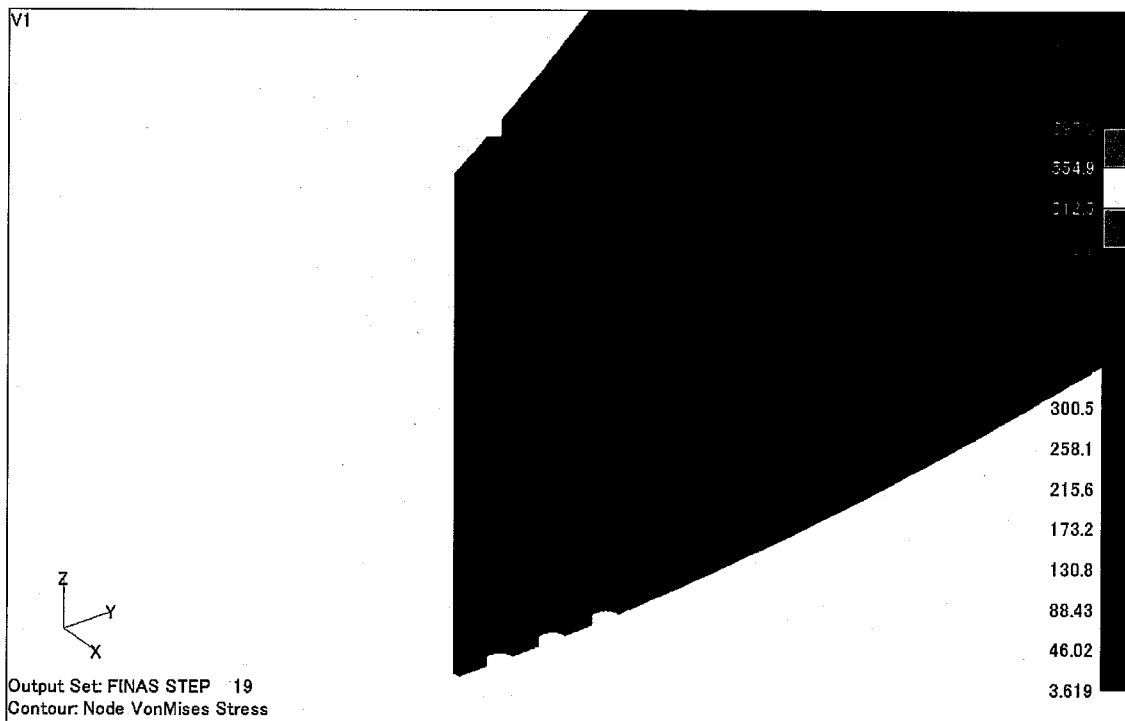


図4. 21 中央部の相当応力分布(1)(最大応力発生時点、t=400 秒)

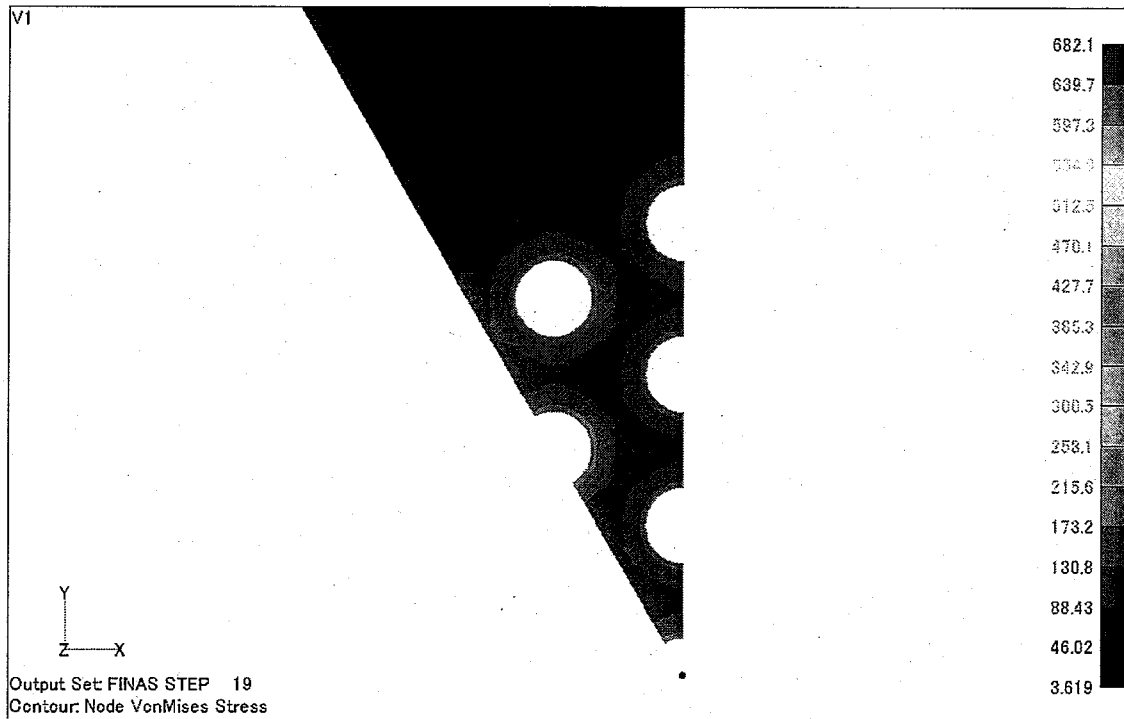


図 4.2 2 中央部の相当応力分布(2)(最大応力発生時点、t=400 秒)

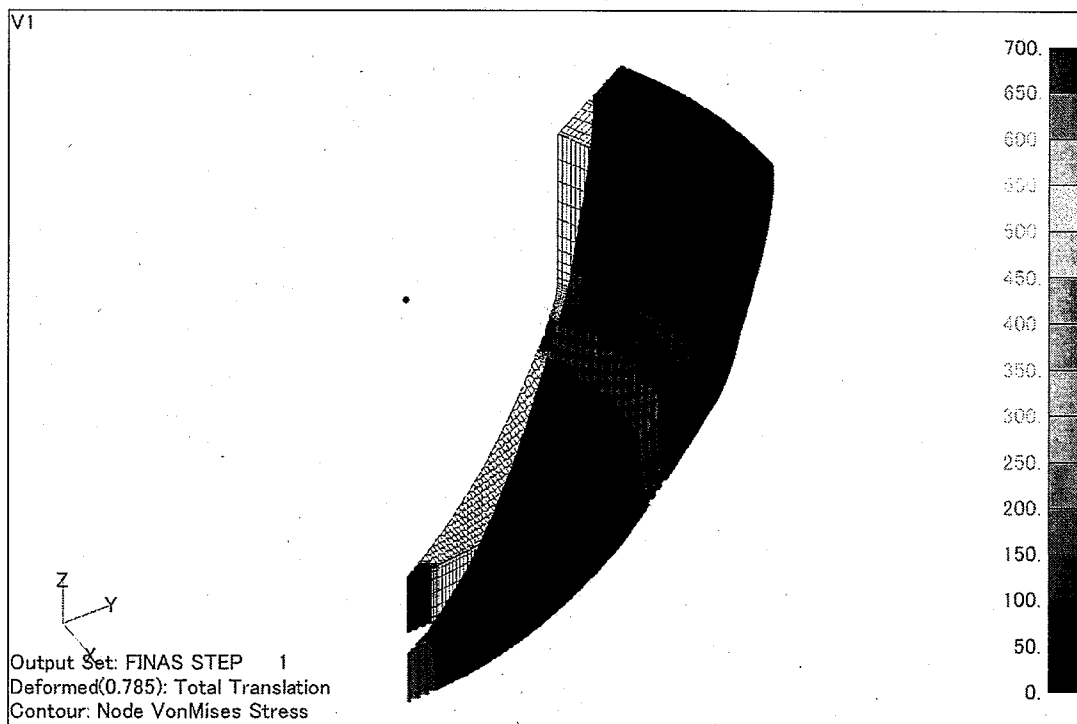


図 4.2 3 相当応力分布(内圧負荷のみ、t=0 秒)

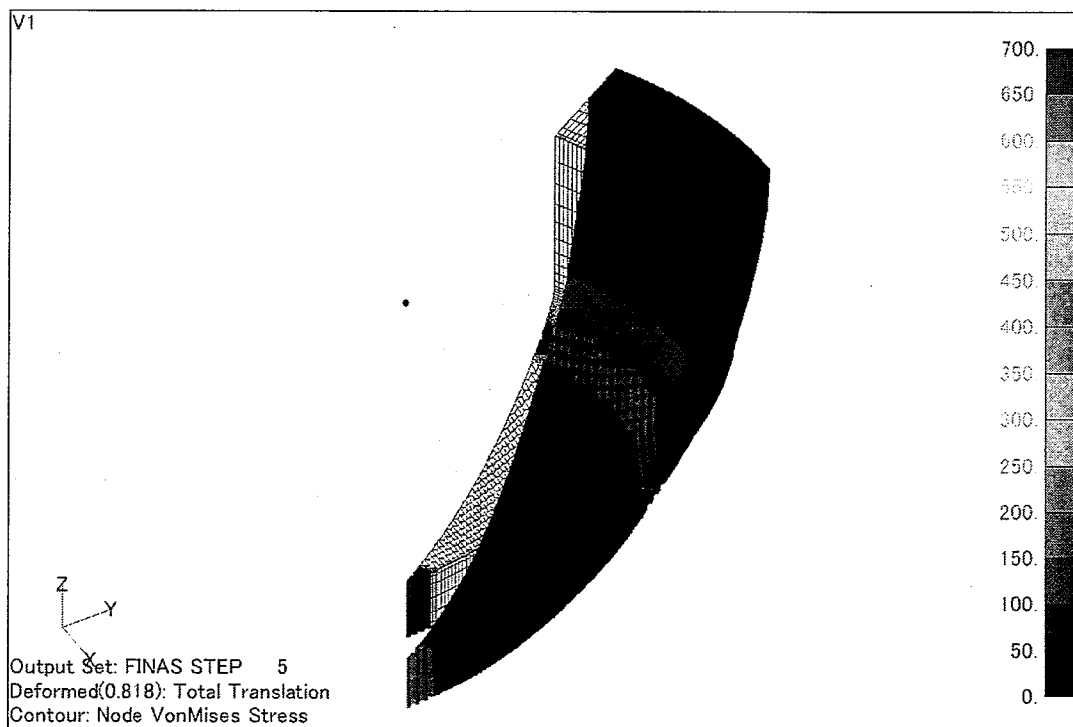


图 4.2 4 相当应力分布 (t=30 秒)

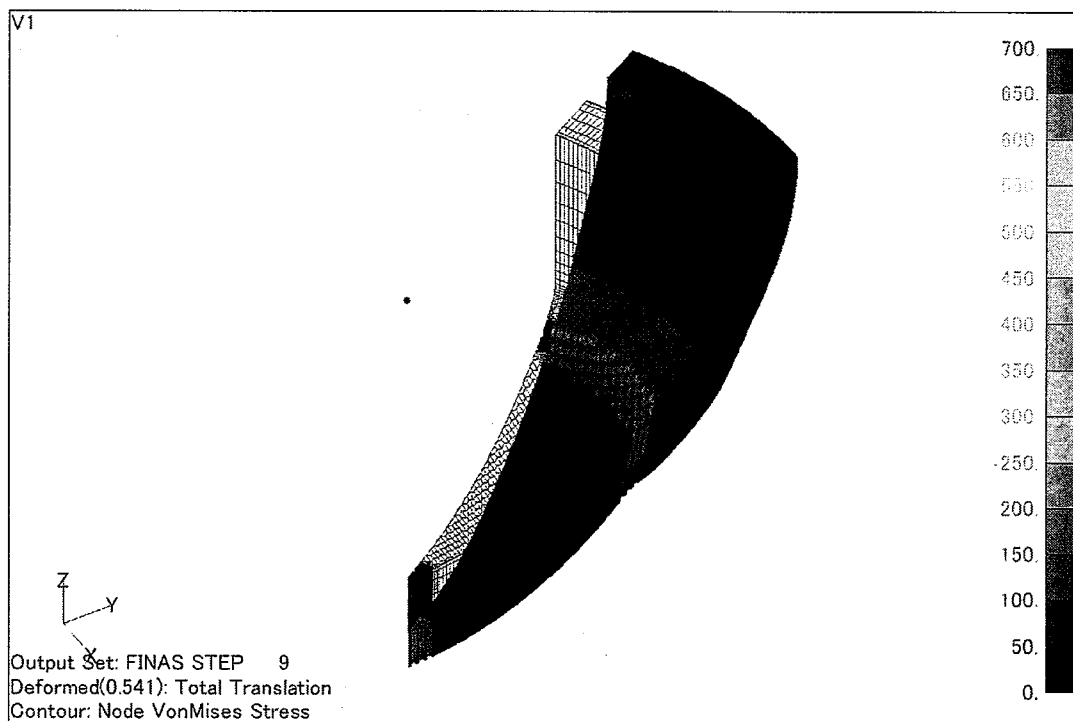


图 4.2 5 相当应力分布 (t=120 秒)

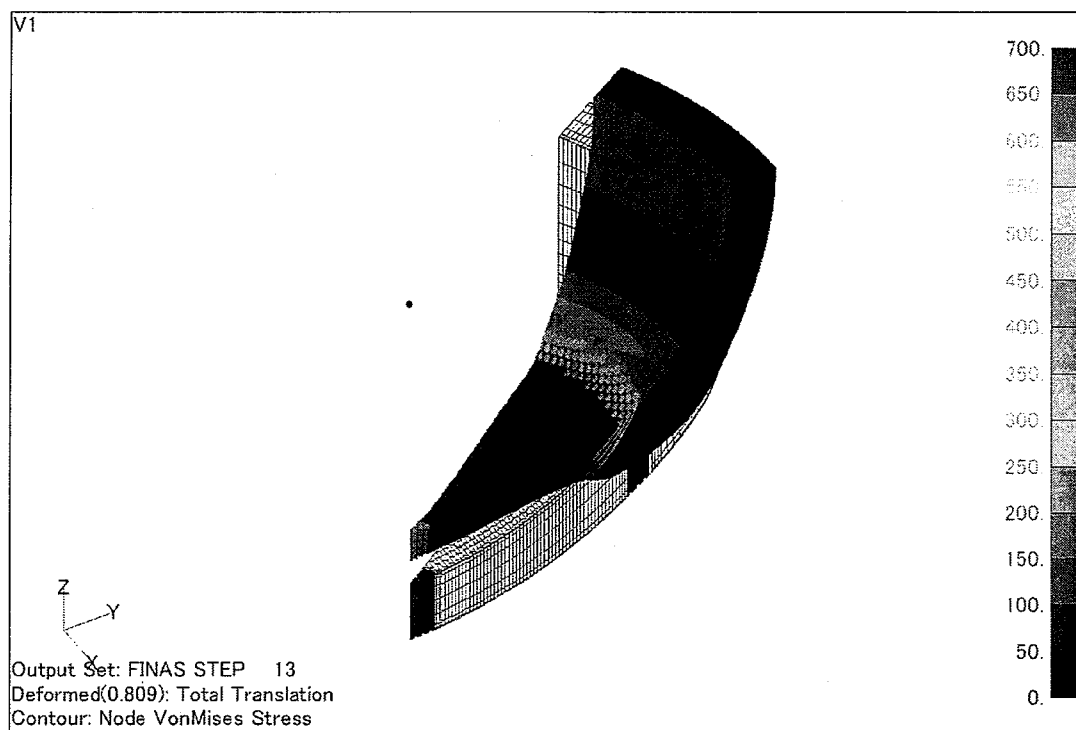


图 4.2 6 相当应力分布 (t=200 秒)

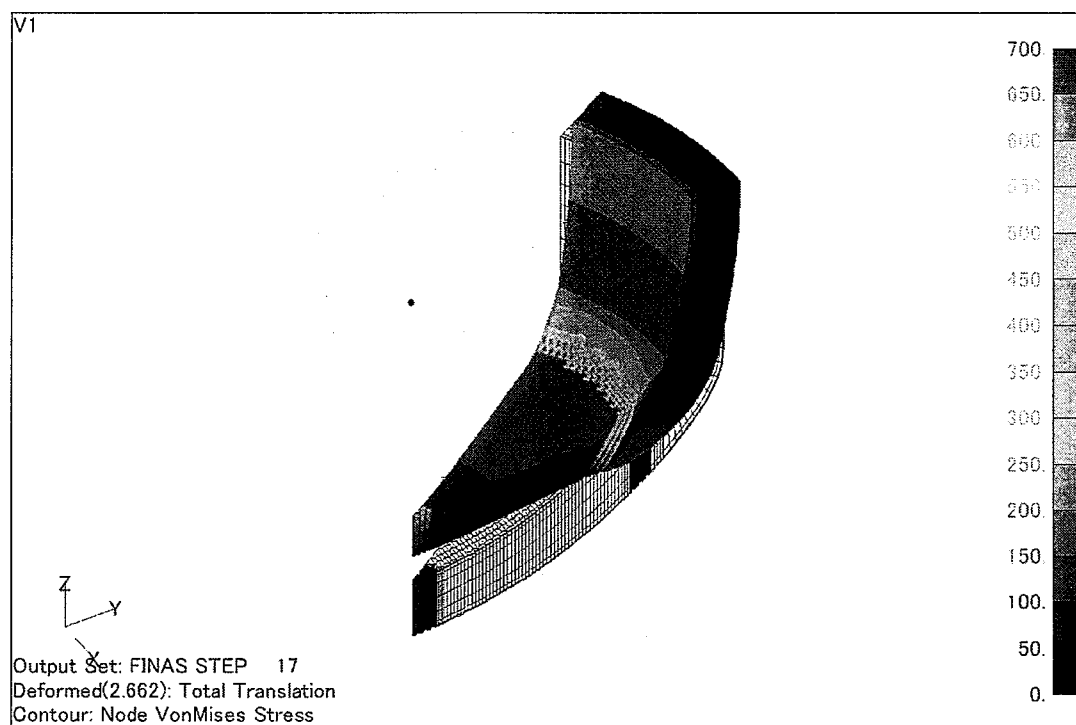


图 4.2 7 相当应力分布 (t=280 秒)

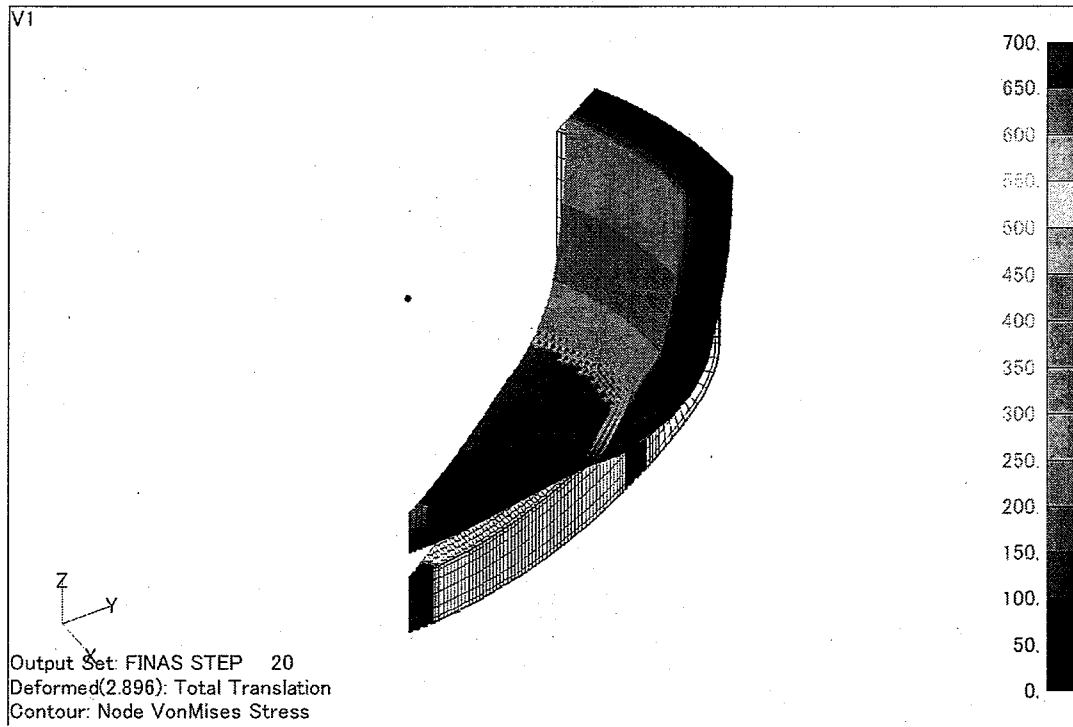


图 4.2 8 相当应力分布 (t=500 秒)

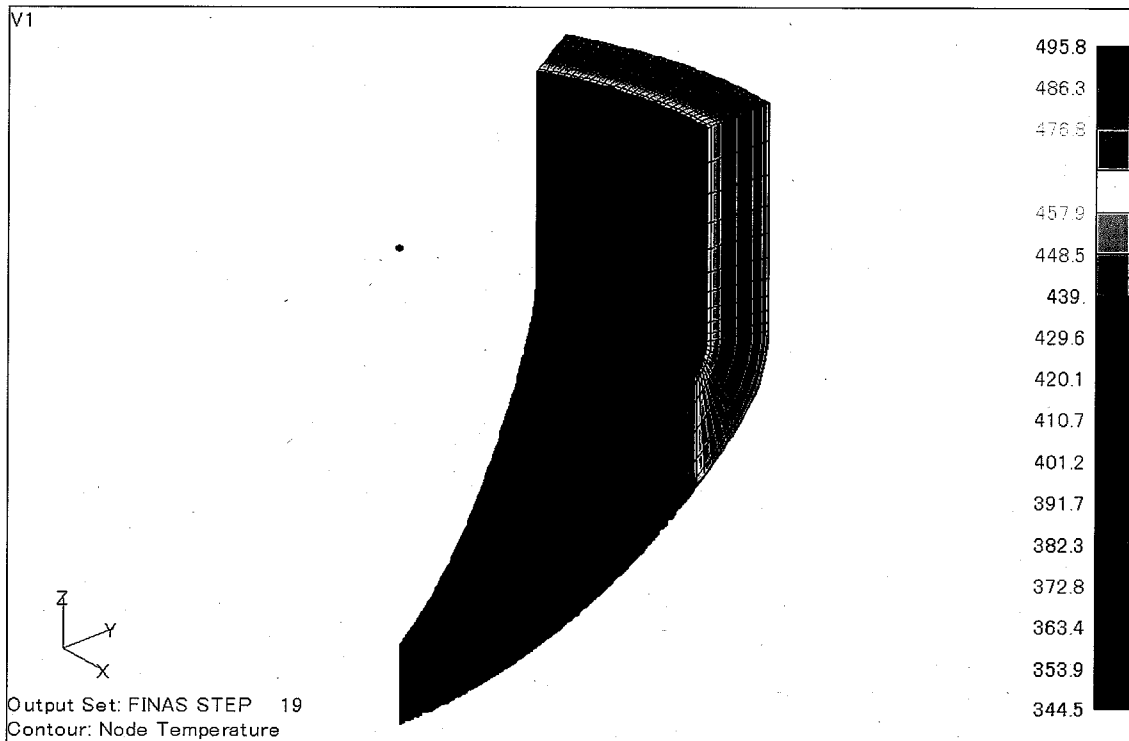


図4. 29 球形管板大規模モデルの熱伝導解析(最終ステップの温度分布、BCS-Metis)

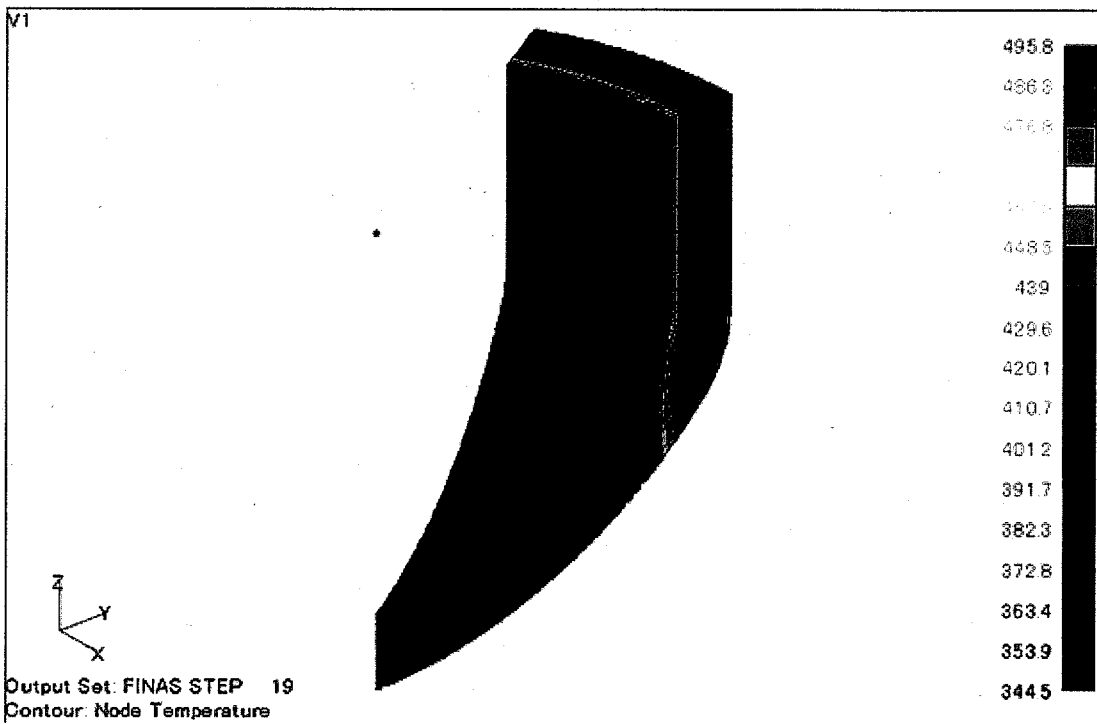


図4. 30 球形管板大規模モデルの熱伝導解析(最終ステップの温度分布、点ヤコビ-CG法)

表4.1 直管型球形管板モデルの概要

項 目		規 模
節点数		342,830
要素数	熱伝導解析	93,237
	応力解析	73,464
自由度数	熱伝導解析	34,2830
	応力解析	1,028,490
解析ステップ数	熱伝導解析	19
	応力解析	20

表4.2 熱伝導解析の物性値

	項 目	値	単 位	備 考
母材部	熱伝導率	7.81E-06	kcal/mm·sec·K	3.27E4W/m·K
	比熱	2.00E-01	1/K	
	密度	7.86E-06	kg/mm ³	
等価領域	熱伝導率	7.81E-06	kcal/mm·sec·K	
	比熱	2.00E-01	1/K	
	密度	5.90E-06	kg/mm ³	体積比で 0.75 倍

表4.3 熱伝導解析に用いた熱伝達係数

場所	値	単 位	備 考
伝熱管内	1.586E-06	kcal/mm ² ·sec·K	6,640W/m ² ·K
管板内部	1.099E-07	K	460W/m ² ·K

表4.4 熱応力解析で用いた材料物性値

	項 目	値	単 位	備 考
母材部	弾性係数	174000	MPa	
	ポアソン比	0.302	mm/mm	
	線膨張係数	1.25E-05	1/K	
	密 度	7.86E-06	kg/mm ³	
等価領域	弾性係数	88740	MPa	ASME Sec III A-8000 より
	ポアソン比	0.3	mm/mm	
	線膨張係数	1.25E-05	1/K	
	密 度	7.86E-06	kg/mm ³	

表4.5 球形管板大規模モデル熱伝導解析の処理時間（自由度数：342,830、19 ステップ）

手 法	合計時間[Sec.]	ソルバ [sec.]	ディスク [Block]	反復法の時間 [sec.]	反復数	メモリ [MB]
BCS (Metis)	10,159.8	9,409.4	68532			
C.G.法(ポイントヤコビ)	2,927.2	2,257.5	68533	80.8~122.5 1,928.0 (合計)	257~390 322.8 (平均)	157.86

表4.6 球形管板大規模モデル熱伝導解析での反復回数

Step	反復数	時間[秒]	Step	反復数	時間[秒]
1	257	80.8	11	331	104.0
2	273	85.9	12	327	102.9
3	290	91.2	13	327	102.9
4	303	95.1	14	330	103.7
5	295	92.7	15	331	104.0
6	289	90.9	16	330	103.7
7	363	114.1	17	334	105.1
8	334	105.0	18	390	122.5
9	330	103.6	19	373	117.2
10	327	102.9	合計	6,134	1,928.0
			平均	322.8	101.5

表4.7 球形管板大規模モデル熱応力解析の処理時間 (自由度数: 1,028,490、20 ステップ)

手 法	合計時間 [Sec.]	ソルバ [sec.]	ディスク [Block]	反復法の時間 [sec.]	反復数	メモリ [MB]
BCS (Metis)	127,548.3	125,416.7	1,814,523			
C.G.法(ポイントヤコビ)*	259,876.6	257,741.2	1,814,523	18,546.7~ 19,033.9/Step	7,367~ 7,567/Step	1,283.2

5. 結言

実用化戦略調査研究で求められている大型球形管板の実用化に必要な、孔廻りの詳細形状を考慮した3次元構造解析法の開発を行った。

3次元解析は、外縁の孔廻りに限定した部分孔開きモデルとしても大規模になるため、通常の直接法ソルバでは計算が難しくなる。このため、大規模解析で計算効率の高くなる反復ソルバを開発し、FINASへ実装した。小規模モデルから段階的にモデル規模を大きくし検証することにより、その適用性を確認した。最終的に100万自由度を超える大型球形管板の部分孔開きモデルを作成し、手動トリップを想定した20ステップの非定常熱伝導解析と熱弾性解析の試計算を実施した。

Pen4 3.06GHzのPCを用いると上記解析を72時間で終了することができ、球形管板の孔廻りの詳細な応力分布を得ることが出来た。以上から、球形管板に対する3次元構造解析法の適用性が確認できた。

6. 参考文献

- [1] 野田 宏, 高速増殖炉サイクルの実用化戦略調査研究の取組み状況, 原学会誌, Vol. 42, No. 7, (2000)
- [2] ASME (1983), Stresses in Perforated Flat Plates, ASME Boiler and Pressure Vessel Code Sec. III Appendix A-8000.
- [3] O'Donnell, W. J. and Porowski, J. S. (1982), Pressure Vessels and Piping Design Technology-1982-A Decade of Progress, ASME, pp. 359/373.
- [4] N. Kasahara et al. (1989), High Temperature Design Methods for Tubesheet Structures with Validation by Thermal Transient Testing, E, 10th SMIRT pp1/12
- [5] 笠原, 他, (1992), '弾性解析による管板構造設計評価法の開発(2)', 機械学会第70期材料力学講演会
- [6] 笠原他, (1993), '等価弾塑性中実板による多孔板の非弾性解析法', 日本機械学会, No. 930-71, 材料力学部門講演会論文集
- [7] 梅田, 他, (1992), '弾性解析による管板構造設計評価法の開発(1)', 機械学会第70期材料力学講演会
- [8] 行列計算ソフトウェア、(1991)、小国力、丸善
- [9] 長谷川里美、長谷川秀彦、藤野清次 訳、(1996)、反復法 Templates、応用数値計算ライブラリ、朝倉書店
- [10] 藤野清次、張紹良、(1996)、反復法の数理、応用数値計算ライブラリ、朝倉書店