



JAEA-Data/Code

2025-018

DOI:10.11484/jaea-data-code-2025-018

高速炉燃料集合体領域の要素分割法の開発

Development of Mesh Generation Method in a Fast Reactor Fuel Assembly

菊地 紀宏 今井 康友 吉川 龍志 田中 正暁
大島 宏之

Norihiro KIKUCHI, Yasutomo IMAI, Ryuji YOSHIKAWA, Masaaki TANAKA
and Hiroyuki OHSHIMA

大洗原子力工学研究所
高速炉研究開発部

Fast Reactor Research and Development Department
Oarai Nuclear Engineering Institute

March 2026

Japan Atomic Energy Agency

日本原子力研究開発機構

JAEA-Data/Code

本レポートは国立研究開発法人日本原子力研究開発機構が不定期に発行する成果報告書です。本レポートはクリエイティブ・コモンズ表示 4.0 国際 ライセンスの下に提供されています。本レポートの成果（データを含む）に著作権が発生しない場合でも、同ライセンスと同様の条件で利用してください。（<https://creativecommons.org/licenses/by/4.0/deed.ja>）
なお、本レポートの全文は日本原子力研究開発機構ウェブサイト（<https://www.jaea.go.jp>）より発信されています。本レポートに関しては下記までお問合せください。

国立研究開発法人日本原子力研究開発機構 研究開発推進部 科学技術情報課
〒 319-1112 茨城県那珂郡東海村大字村松 4 番地 49
E-mail: ird-support@jaea.go.jp

This report is issued irregularly by Japan Atomic Energy Agency.

This work is licensed under a Creative Commons Attribution 4.0 International License (<https://creativecommons.org/licenses/by/4.0/deed.en>).

Even if the results of this report (including data) are not copyrighted, they must be used under the same terms and conditions as CC-BY.

For inquiries regarding this report, please contact Library, Institutional Repository and INIS Section, Research and Development Promotion Department, Japan Atomic Energy Agency.

4-49 Muramatsu, Tokai-mura, Naka-gun, Ibaraki-ken 319-1112, Japan

E-mail: ird-support@jaea.go.jp

高速炉燃料集合体領域の要素分割法の開発

日本原子力研究開発機構 大洗原子力工学研究所 高速炉研究開発部

菊地 紀宏、今井 康友*、吉川 龍志、田中 正暁、大島 宏之†

(2025 年 11 月 27 日受理)

日本原子力研究開発機構では、ナトリウム冷却高速炉（高速炉）の燃料集合体内の詳細な熱流動評価を目的として、有限要素法による詳細熱流動解析コード SPIRAL の整備を進めている。高速炉での特長的なワイヤスペーサ型燃料集合体では、計算に利用する計算格子（要素）の品質が予測精度に大きく影響するため、燃料集合体ピンバンドル領域に高品質の要素を配置することが数値解析を実行する上で重要な課題となる。複雑な燃料集合体領域の要素分割を行う手段としては、燃料集合体形状を CAD データで再現した上で市販されている汎用の計算格子生成プログラム（メッシャー）を利用することが考えられるが、極めて煩雑な作業となる。そこで、高品質の要素分割を効率的に配置するため、燃料集合体の幾何形状（設計情報）と要素分割を設定するパラメーターを入力条件として、燃料集合体領域の要素分割を自動で実行するメッシャーを開発した。本報では、このメッシャーの各種要素分割モデルの詳細とその利用法について詳説する。本メッシャーでは、複雑形状となる燃料集合体領域に対して、計算格子を規則的に配置するためマルチブロック法による領域分割を行った上で、それぞれのブロック領域で曲線座標系による境界適合格子を生成し、最終的に統合して一つの燃料集合体体系とする要素分割法を採用した。また、隣接するブロック領域間での要素の連続性を維持するため、六面体（Hexa）要素とプリズム状の（Prism）要素を併用する要素配置とした。以上の六角形断面のラップ管で囲まれた燃料集合体に対する基本的な要素分割機能に加え、熔融燃料の排出を促進するため燃料集合体内部にダクトを設けた変則的な燃料集合体に対する要素分割も可能である。

本メッシャーの開発によって、様々な条件における複雑な燃料集合体領域の要素分割を正確かつ効率的に実行することが可能となった。

大洗原子力工学研究所：〒311-1393 茨城県東茨城郡大洗町成田町 4002 番地

† エネルギー研究開発領域

* エヌデーデー

Development of Mesh Generation Method in a Fast Reactor Fuel Assembly

Norihiro KIKUCHI, Yasutomo IMAI*, Ryuji YOSHIKAWA, Masaaki TANAKA and Hiroyuki OHSHIMA⁺

Fast Reactor Research and Development Department

Oarai Nuclear Engineering Institute

Japan Atomic Energy Agency

Oarai-machi, Higashiibaraki-gun, Ibaraki-ken

(Received November 27, 2025)

In the Japan Atomic Energy Agency, a detailed thermal-hydraulic analysis code named SPIRAL based on the finite element method (FEM) is being developed to evaluate the detailed thermal-hydraulic properties of fuel assemblies (FAs) in sodium-cooled fast reactors (FBRs). Because the quality of the computational grid (elements) used in the calculations has a significant impact on the prediction accuracy, the allocation of high-quality elements in the wire-spacer-type FA pin bundle region is an important issue for numerical analysis. Although a commercial mesh generation program (mesher) with CAD data of FA's geometric shape can be considered as one measure, it is an extremely complicated task to perform element division of complex FA region. Therefore, to efficiently allocate high-quality elements, we developed a mesher that automatically performs element division in the FA region using the FA's geometric shape (design information) and meshing parameters as input conditions. This report describes the details of the mesher's various meshing models and their usage. To regularly allocate the computational grid for the complex FA region, the mesher first divides the region into multiple blocks using a multi-block method, then generates boundary-fitted curvilinear coordinate grids for each block region, and finally integrates them into a single FA mesh system. In addition, a combination of hexahedral elements and prism-shaped elements is arranged to maintain element continuity between adjacent block regions. Element division for both the normal FAs surrounded by a hexagonal cross-section tube and the irregular FAs, inside which a duct is installed to promote the discharge of molten fuel, is possible. The development of this mesher has made it possible to accurately and efficiently perform element division of complex FA region on various conditions.

Keywords: Fast Reactor, Fuel Assembly, Finite Element Method, Mesh Generation

⁺ Nuclear Energy Research and Development Domain

* NDD Corporation

目 次

1. 緒言.....	1
2. 燃料集合体領域の基本的な要素分割.....	3
2.1 マルチブロック法の適用.....	3
2.2 曲線座標系による境界適合格子の生成.....	3
2.2.1 構造材表面の節点座標値の設定.....	6
2.2.2 構造材表面の節点座標値の補正.....	8
2.2.3 構造材要素の生成.....	9
2.2.4 境界層要素の生成.....	9
2.2.5 双子節点の生成.....	10
2.3 三次元要素の生成.....	10
2.4 領域データ・境界データの設定.....	11
3. 燃料集合体領域の随意的な要素分割モデル.....	30
3.1 構造材変形モデル.....	30
3.2 構造材接触モデル.....	31
3.3 内部ダクト付き燃料集合体モデル.....	31
3.4 1/nセクタモデル.....	32
3.5 局所閉塞モデル.....	32
3.6 要素修正モデル.....	33
4. 結言.....	45
参考文献.....	46
付録 燃料集合体領域計算格子生成プログラム 利用マニュアル.....	47

Contents

1. Introduction	1
2. Basic Mesh Generation Model in Fuel Assembly Region.....	3
2.1 Application of Multi-Block Method.....	3
2.2 Generation of Boundary-Fitted Curvilinear Coordinate Grids	3
2.2.1 Calculation of Nodal Coordinates on Solid Surface.....	6
2.2.2 Correction of Nodal Coordinates on Solid Surface.....	8
2.2.3 Generation of Solid Meshes	9
2.2.4 Generation of Boundary Layer Meshes.....	9
2.2.5 Generation of Twin-Nodes	10
2.3 Generation of Three-Dimensional Meshes	10
2.4 Setting of Region data and Boundary Data.....	11
3. Optional Mesh Generation Models in Fuel Assembly Region	30
3.1 Structure Deformation Model.....	30
3.2 Structure Contact Model.....	31
3.3 Fuel Assembly with Inner Duct Model.....	31
3.4 $1/n$ Sector Model.....	32
3.5 Local Blockage Model.....	32
3.6 Mesh Correction Model.....	33
4. Conclusion.....	45
References	46
Appendix User’s Manual on the Computational Mesh Generation Program for Fuel Assembly Region .	47

図リスト

図 1.1	燃料集合体ピンバンドルの概念	2
図 2.1	燃料集合体領域のブロック分割 (7 本ピン集合体)	12
図 2.2.1	燃料ピン周辺ブロック領域における境界適合格子 (ワイヤなし条件)	13
図 2.2.2	燃料ピン周辺ブロック領域における境界適合格子 (ワイヤあり条件)	14
図 2.2.3	ラップ管周辺ブロック領域における境界適合格子	15
図 2.3.1	サブチャンネル中心点上の節点	16
図 2.3.2	ギャップ線上の節点	16
図 2.3.3	燃料ピン周辺領域上の節点	17
図 2.3.4	ラップ管周辺領域上の節点	17
図 2.4.1	19 本ピン集合体におけるサブチャンネル中心点番号	18
図 2.4.2	19 本ピン集合体におけるギャップ線番号	18
図 2.5.1	燃料ピン表面の節点座標値の設定	19
図 2.5.2	ワイヤ表面の節点座標値の設定	20
図 2.5.3	ラップ管表面の節点座標値の設定	21
図 2.6.1	燃料ピン表面・ワイヤ表面における境界層要素の生成	22
図 2.6.2	ラップ管表面における境界層要素の生成 (ラップ管頂点付近の拡大)	23
図 2.7.1	燃料ピン周辺ブロック領域における三次元要素の生成	24
図 2.7.2	ラップ管周辺ブロック領域における三次元要素の生成	25
図 2.8.1	燃料ピン周辺ブロック領域における領域データ	26
図 2.8.2	ラップ管周辺ブロック領域における領域データ	27
図 2.9.1	燃料ピン周辺ブロック領域における境界データ	28
図 2.9.2	ラップ管周辺ブロック領域における境界データ	29
図 3.1	燃料ピン・ワイヤ変形モデル	34
図 3.2.1	構造材接触モデル (構造材接触点の検出)	35
図 3.2.2	構造材接触モデル (固体節点の追加・流体節点の削除)	36
図 3.2.3	構造材接触モデル (要素種類の変更)	37
図 3.2.4	構造材接触モデル (低品質要素の修正)	38
図 3.2.5	要素種類の変更	39
図 3.3.1	内部ダクト付き 18 本ピン集合体	42
図 3.3.2	変形六角管型ラップ管 3 本ピン集合体	42
図 3.4	ベアバンドル集合体・ $1/n$ セクタモデル	43
図 3.5	燃料集合体局所閉塞モデル	44

記号表

D_d	ラップ管内対面間距離[m]
D_p	燃料ピン径[m]
D_w	ワイヤ径[m]
e_d	ラップ管表面のメッシュ緩和係数[-]
e_p	燃料ピン表面のメッシュ緩和係数[-]
H	ワイヤ巻き付けピッチ[m]
J_{ij}	ヤコビ行列[-]
n_d	ラップ管表面の節点数[-]
n_p	燃料ピン表面の節点数[-]
n_w	ワイヤ表面の節点数[-]
N_{gap}	ギャップ線数[-]
N_{pin}	燃料ピン本数[-]
$N_{subch.}$	サブチャンネル中心点数[-]
N_{SOR}	SOR法の最大反復計算回数[-]
$P(\xi, \eta), Q(\xi, \eta)$	要素分布密度の制御関数[-]
(x, y, z)	直交座標系[m]
(x_i, y_i, z_i)	節点 i の節点座標値[m]
(x_p, y_p)	燃料ピン中心点座標値[m]
(x_w, y_w)	ワイヤ中心点座標値[m]
(X_n, Y_n)	ラップ管頂点の座標値[m]
W_d	ラップ管肉厚[m]
α_{SOR}	SOR法の緩和係数[-]
ϵ_{SOR}	SOR法の収束判定値[-]
ΔD_d	ラップ管内対面間距離の膨張量[m]
ΔD_p	燃料ピン径の膨張量[m]
ΔD_q	燃料ペレット径の膨張量[m]
ΔD_w	ワイヤ径の膨張量[m]
Δl_w	境界層要素幅[m]
$(\Delta x_p, \Delta y_p)$	燃料ピン中心点の偏心量[m]
$(\Delta x_q, \Delta y_q)$	燃料ペレット中心点の偏心量[m]
$(\Delta x_w, \Delta y_w)$	ワイヤ中心点の偏心量[m]
ΔW_d	ラップ管肉厚の膨張量[m]
θ_w	ワイヤ位相[rad]
(ξ, η, ζ)	曲線座標系[-]

1. 緒言

高速炉燃料集合体内の詳細な熱流動評価を目的として、有限要素法による詳細熱流動解析コード SPIRAL の整備を進めている¹⁾²⁾。高速炉での特長的なワイヤスペーサ型燃料集合体では、計算に利用する計算格子（要素）の品質が予測精度に大きく影響するため、高品質の要素を燃料集合体内のピンバンドル領域に配置することが数値解析を実行する上で重要な課題となる。

図 1.1 に燃料集合体ピンバンドルの概念を示す。燃料集合体内の燃料ピンで囲まれた狭い流路となるサブチャンネル、燃料ピンの周りに螺旋状に巻かれたワイヤスペーサ（以下、ワイヤ）、通過流量を制御するために設置された六角形の断面を有するラップ管から構成される複雑な形状を呈する燃料集合体領域の要素分割を行う手段としては、燃料集合体形状を CAD データで再現した上で市販されている汎用の計算格子生成プログラム（メッシャー）を利用することが考えられる。このとき、計算格子としては、座標系に沿って規則正しく配置する構造格子と、境界形状の再現性に優れた非構造格子があるが、燃料集合体領域には複数本の燃料ピンとそれに螺旋状に巻かれたワイヤの存在により、座標系がワイヤの巻き方向に従って変化するため、燃料集合体内に構造格子を配置することは困難となる。他方、非構造格子による要素分割は、複雑形状に対して汎用的に適用できるメリットがあるが、四角形や六面体などの規則的に配置された構造格子による要素分割と比較して、同程度の格子サイズでは計算に必要な要素数が増大する上、一般的に数値解析の予測精度が低下するとの懸念がある。このような理由から、複雑な燃料集合体を対象とする場合、汎用のメッシャーを利用した格子生成作業は極めて煩雑な作業となる。

このように、汎用メッシャーの利用は容易ではないことから、高品質の要素分割を効率的に配置するため、燃料集合体幾何形状（設計情報）と要素分割を設定するパラメーターを入力条件として、燃料集合体領域の要素分割を自動で実行するメッシャーを開発した。複雑形状となる燃料集合体領域に対して、構造格子のような要素分割を行うため、本メッシャーでは、燃料ピン周りを一つの基準としてマルチブロック法による領域分割を実行した上で、それぞれのブロック領域で曲線座標系による境界適合格子を生成する要素分割法を採用した。また、隣接するブロック領域間で要素の連続性を維持するため、六面体（Hexa）要素を基本として、ワイヤ周辺にはプリズム状の（Prism）要素を併用することでワイヤを含む複雑な燃料集合体内の基本的な要素分割を可能とした。さらに、様々なニーズに対応するため、例えば内部ダクト付き燃料集合体³⁾のような変則的な燃料集合体を対象とした随意的な要素分割も可能とした。

本報では、第 2 章で燃料集合体領域の基本的な要素分割モデルを、第 3 章で随意的な要素分割モデルを、それぞれ解説する。また、付録において本メッシャーの利用法を記載する。

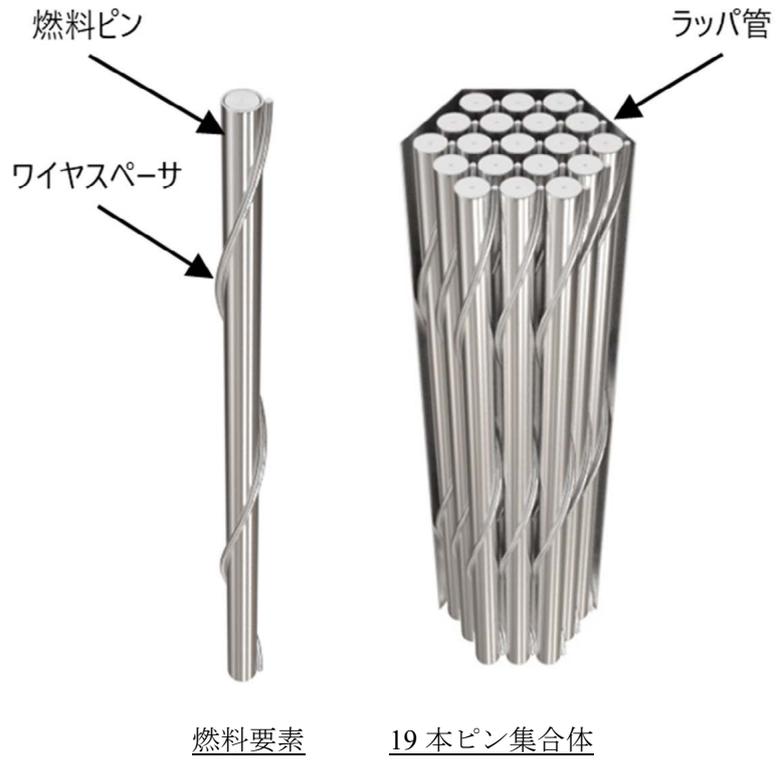


図 1.1 燃料集合体ピンバンドルの概念

2. 燃料集合体領域の基本的な要素分割

燃料集合体内のピンバンドル領域の要素分割に際しては、燃料集合体を構成する燃料ピン、ワイヤ、ラップ管の幾何形状を正確に取り扱うことが数値解析の予測精度を向上させるために重要となる。その課題を解決するため、燃料集合体領域の要素分割法として、曲線座標系による境界適合格子を採用した。境界適合格子による要素分割法では、領域の境界形状に沿って曲線座標系の座標軸が定義されるため、複雑形状領域においても、形状を正確に模擬した構造的な要素を生成できる利点がある。ただし、燃料集合体領域には複数本の燃料ピンとそれに螺旋状に巻かれたワイヤが存在し、ワイヤの巻き付け位置に応じて座標系が変化するため、それらを一つの座標系で一元的に表現することは極めて難しい。そのため、燃料集合体内の燃料ピン周り及びラップ管周辺領域について複数のブロック領域に分割して、それぞれのブロック領域で曲線座標系を定義するマルチブロック法を採用した。

燃料集合体領域の要素分割の手順を示すと、以下の通りとなる。

- (1) 入力データ（設計情報等）ファイルを作成する。
- (2) 燃料集合体領域を各燃料ピン周辺領域とラップ管周辺領域にブロック分割する。
また、集合体鉛直方向に対し、各ブロック領域で均等に分割する。
- (3) 各ブロック領域の各水平断面において、二次元の境界適合格子を生成する。
- (4) 隣接する水平断面間で二次元要素を結合して、三次元要素を生成する。
- (5) 生成された三次元要素に対して、領域データ・境界データを設定する。
- (6) 要素データファイルを出力する。

2.1 マルチブロック法の適用

境界適合格子による要素分割に際して、曲線座標系を容易に設定するため、燃料集合体領域を各燃料ピン周辺領域とラップ管周辺領域にブロック分割する。

図 2.1 に 7 本ピン集合体の場合の燃料集合体領域のブロック分割図を示す。N 本ピン集合体では、(N+1) のブロック領域が生成される。また、集合体鉛直方向に対しては、各ブロック領域で均等に分割する。

2.2 曲線座標系による境界適合格子の生成

各ブロック領域の各水平断面において、曲線座標系による二次元の境界適合格子を生成する。曲線座標系は、構造材の接線方向と法線方向に張られたそれぞれ交差する座標系 (ξ, η) として定

義される。すなわち、燃料ピンまたはラッパ管に付随する各ブロック領域において、図 2.2.1、図 2.2.2、図 2.2.3 に示す通り、それぞれの周方向を ξ 方向、径方向を η 方向と定義する。したがって、各ブロック領域では、リング状の計算格子が生成される（これを O 型格子と呼ぶ⁴⁾）

計算格子生成法としては、Thompson⁵⁾らによる楕円型偏微分方程式を利用した格子生成法を採用する。同法では、直交座標系 (x, y) と曲線座標系 (ξ, η) を次の楕円型偏微分方程式(ポアソン方程式)に代入して、境界適合格子を生成する。

$$\left. \begin{aligned} \xi_{xx} + \xi_{yy} &= P(\xi, \eta) \\ \eta_{xx} + \eta_{yy} &= Q(\xi, \eta) \end{aligned} \right\} \dots (2.1)$$

ここで、 ξ_{xx} は、変数 ξ の x 方向の 2 階偏微分を表す。直交座標系 (x, y) と曲線座標系 (ξ, η) の座標変換に必要なヤコビ行列 J_{ij} と逆行列 J_{ij}^{-1} は、次式で定義される。

$$\left. \begin{aligned} J_{ij} &\equiv \frac{\partial x_i}{\partial \xi_j} \equiv \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix} \\ J_{ij}^{-1} &\equiv \frac{\partial \xi_i}{\partial x_j} \equiv \begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} = \frac{1}{J} \begin{bmatrix} y_\eta & -x_\eta \\ -y_\xi & x_\xi \end{bmatrix} \\ J &\equiv \det |J_{ij}| = x_\xi y_\eta - x_\eta y_\xi \end{aligned} \right\} \dots (2.2)$$

式 (2.1) を式 (2.2) で座標変換すると、

$$\left. \begin{aligned} \alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} &= -\frac{1}{J^2} [P(\xi, \eta)x_\xi + Q(\xi, \eta)x_\eta] \\ \alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} &= -\frac{1}{J^2} [P(\xi, \eta)y_\xi + Q(\xi, \eta)y_\eta] \\ \alpha &= x_\eta^2 + y_\eta^2, \quad \beta = x_\xi x_\eta + y_\xi y_\eta, \quad \gamma = x_\xi^2 + y_\xi^2 \end{aligned} \right\} \dots (2.3)$$

となる。ここで、式 (2.3) の右辺に現れる $P(\xi, \eta)$ 、 $Q(\xi, \eta)$ は、要素の分布密度を制御するための関数であり、Steger-Sorenson⁶⁾ による定式化では、次式で定義される。

$$\begin{aligned}
 P(\xi, \eta) &= p_1(\xi) \exp[-a_1(\eta - \eta_{\min})] + p_2(\xi) \exp[-a_2(\eta_{\max} - \eta)] \\
 Q(\xi, \eta) &= q_1(\xi) \exp[-b_1(\eta - \eta_{\min})] + q_2(\xi) \exp[-b_2(\eta_{\max} - \eta)] \\
 p_1(\xi) &= J(y_\eta R_{11} - x_\eta R_{12}) \Big|_{\eta=\eta_{\min}} \\
 q_1(\xi) &= J(-y_\xi R_{11} + x_\xi R_{12}) \Big|_{\eta=\eta_{\min}} \\
 p_2(\xi) &= J(y_\eta R_{21} - x_\eta R_{22}) \Big|_{\eta=\eta_{\max}} \\
 q_2(\xi) &= J(-y_\xi R_{21} + x_\xi R_{22}) \Big|_{\eta=\eta_{\max}} \\
 R_{11} &= -J^2(\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta}) \Big|_{\eta=\eta_{\min}} \\
 R_{12} &= -J^2(\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta}) \Big|_{\eta=\eta_{\min}} \\
 R_{21} &= -J^2(\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta}) \Big|_{\eta=\eta_{\max}} \\
 R_{22} &= -J^2(\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta}) \Big|_{\eta=\eta_{\max}} \\
 R_{22} &= -J^2(\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta}) \Big|_{\eta=\eta_{\max}}
 \end{aligned}
 \tag{2.4}$$

式 (2.3) のポアソン方程式は、各節点の座標値 (x_i, y_i) に対して導出される。したがって、水平断面上に存在する節点数を n_{xy} とすると、連立方程式の次数は $2n_{xy}$ となる。連立方程式の解法としては、アルゴリズムが明瞭かつ収束性に優れた SOR 法 (Successive Over-Relaxation method) ⁷⁾ を採用する。SOR 法では、連立方程式を逐次反復的に計算するが、反復計算内での計算順序は、以下の通りに行われる。

- (1) サブチャンネル中心点上の節点
- (2) ギャップ線上の節点
- (3) 各燃料ピン周辺領域上の節点
- (4) ラップ管周辺領域上の節点

図 2.3.1、図 2.3.2、図 2.3.3、図 2.3.4 にそれぞれの定義を示す。基本的に 3 個のブロック領域の境界点がサブチャンネル中心点 (図 2.3.1)、2 個のブロック領域の境界線がギャップ線 (図 2.3.2) と定義される。また、サブチャンネル中心点、ギャップ線に対しては、それぞれ昇順の番号が設

定されるが、図 2.4.1 及び図 2.4.2 にその設定規則を示す。基本的な規則として、集合体中心側からラッパ管側へと向けて、 $\theta=0$ を起点として反時計回りに番号が設定される。なお、燃料集合体の燃料ピン層数を n とすると、燃料ピン本数 N_{pin} 、サブチャンネル中心点数 $N_{subch.}$ 、ギャップ線数 N_{gap} は、それぞれ次式で算出される。

$$N_{pin} = 3n^2 - 3n + 1, \quad N_{subch.} = 6n^2, \quad N_{gap} = 9n^2 - 3n \quad \dots (2.5)$$

反復回数 k における節点 i の座標値 (x_i^k, y_i^k) に対して、ポアソン方程式から逐次的に算出される座標値 (x_i^*, y_i^*) とすると、新しい座標値 (x_i^{k+1}, y_i^{k+1}) は、次式で設定される。

$$\left. \begin{aligned} x_i^{k+1} &= x_i^k + \alpha_{SOR} \Delta x_i, & y_i^{k+1} &= y_i^k + \alpha_{SOR} \Delta y_i \\ \Delta x_i &= x_i^* - x_i^k, & \Delta y_i &= y_i^* - y_i^k \end{aligned} \right\} \dots (2.6)$$

ここで、 α_{SOR} は SOR 法の反復過程で用いる加速係数である。SOR 法の最大反復計算回数 N_{SOR} 、収束判定値 ε_{SOR} は、それぞれ入力データとして指定する。デフォルト値は、 N_{SOR} が各水平断面上の節点数、 ε_{SOR} が 10^{-5} である。収束の判定では、反復過程間における節点座標値の差 Δr_i を燃料ピン径 D_p で無次元化した値が、すべての節点で ε_{SOR} 未満となった場合に収束と判定される。すなわち、

$$\frac{\max |\Delta r_i|}{D_p} < \varepsilon_{SOR}, \quad \Delta r_i = \sqrt{\Delta x_i^2 + \Delta y_i^2} \quad \dots (2.7)$$

2.2.1 構造材表面の節点座標値の設定

構造材表面の節点座標値は、燃料集合体形状から幾何学的に算出する。構造材表面では、基本的に等間隔の要素分割を適用する。算出された節点座標値は、式 (2.3) のポアソン方程式において、ディレクレ型境界条件として用いられる。

燃料ピン表面の節点座標値 (x_i, y_i) は、次式で算出される。燃料ピン表面の節点数 n_p に対して、燃料ピン表面の節点番号 $i=1, 2, 3, \dots, n_p$ であり、燃料ピンワイヤ接触点を起点として、反時計回りに番号が設定される。また、図 2.5.1 に示す通り、局所座標系 (u, v) は、燃料ピン中心点を原点、燃料ピンワイヤ接触点の方向 ($\theta=\theta_w$) を $+u$ 方向とする直交座標系である。

$$\left. \begin{aligned} x_i &= x_p + \Delta x_i, & y_i &= y_p + \Delta y_i \\ \Delta x_i &= u_i \cos \theta_w - v_i \sin \theta_w, & \Delta y_i &= u_i \sin \theta_w + v_i \cos \theta_w \\ u_i &= r_p \cos \varphi_i, & v_i &= r_p \sin \varphi_i \\ r_p &= \frac{D_p}{2}, & \varphi_i &= 2\pi \xi_i, & \xi_i &= \frac{i-1}{n_p} \end{aligned} \right\} \dots (2.8)$$

- (x_i, y_i) : 燃料ピン表面の節点座標値[m]
- i : 燃料ピン表面の節点番号[-] ($i=1, 2, 3, \dots, n_p$)
- n_p : 燃料ピン表面の節点数[-]
- (x_p, y_p) : 燃料ピン中心点座標値[m]
- D_p : 燃料ピン径[m]
- θ_w : ワイヤ位相[rad]

ワイヤ表面の節点座標値 (x_i, y_i) は、次式で算出される。ワイヤ表面の節点数 n_w に対して、ワイヤ表面の節点番号 $i=1, 2, 3, \dots, n_w$ であり、燃料ピン-ワイヤ接触点を起点として、反時計回りに番号が設定される。また、図 2.5.2 に示す通り、局所座標系 (u, v) は、ワイヤ中心点を原点、燃料ピン-ワイヤ接触点の方向 ($\theta=\theta_w+\pi$) を $+u$ 方向とする直交座標系である。なお、燃料ピンに螺旋状に巻かれたワイヤは、水平断面上での切り口が楕円となるため、楕円率 c でその影響を考慮している。

$$\begin{aligned}
 x_i &= x_w + \Delta x_i, & y_i &= y_w + \Delta y_i \\
 x_w &= x_p + (r_p + r_w) \cos \theta_w, & y_w &= y_p + (r_p + r_w) \sin \theta_w \\
 -\Delta x_i &= u_i \cos \theta_w - v_i \sin \theta_w, & -\Delta y_i &= u_i \sin \theta_w + v_i \cos \theta_w \\
 u_i &= r_w \cos \varphi_i, & v_i &= \frac{r_w}{c} \sin \varphi_i \\
 r_w &= \frac{D_w}{2}, & \varphi_i &= 2\pi \xi_i, & \xi_i &= \frac{i-1}{n_w} \\
 c &= \frac{H}{\sqrt{H^2 + L^2}}, & L &= \pi(D_p + D_w)
 \end{aligned}
 \tag{2.9}$$

- (x_i, y_i) : ワイヤ表面の節点座標値[m]
- i : ワイヤ表面の節点番号[-] ($i=1, 2, 3, \dots, n_w$)
- n_w : ワイヤ表面の節点数[-]
- (x_w, y_w) : ワイヤ中心点座標値[m]
- D_w : ワイヤ径[m]
- H : ワイヤ巻き付けピッチ[m]

ラップ管表面の節点座標値 (x_i, y_i) は、次式で算出される。ラップ管表面の節点数 n_d に対して、ラップ管表面の節点番号 $i=1, 2, 3, \dots, n_d$ であり、 $+x$ 方向のラップ管頂点を起点として、反時計回りに番号が設定される。また、図 2.5.3 に示す通り、 (X_n, Y_n) は、ラップ管頂点の座標値である。従って、次式では、ラップ管の各辺で等間隔の要素分割が実行される。なお、 $[\xi_i]$ は、 ξ_i の Floor 関数值 (ξ_i 以下の最大の整数値) であり、 $0 \leq \xi_i < 6$ から $n \equiv [\xi_i] + 1 = 1, 2, 3, 4, 5, 6, 0 \leq \alpha_i < 1$ である。

$$\begin{aligned}
 x_i &= (1-\alpha_i)X_n + \alpha_i X_{n+1}, & y_i &= (1-\alpha_i)Y_n + \alpha_i Y_{n+1} \\
 X_n &= \frac{D_d}{\sqrt{3}} \cos \Phi_n, & Y_n &= \frac{D_d}{\sqrt{3}} \sin \Phi_n, & \Phi_n &= \frac{\pi}{3}(n-1) \\
 \alpha_i &= \zeta_i - [\zeta_i], & n &= [\zeta_i] + 1, & \zeta_i &= \frac{i-1}{n_d/6}
 \end{aligned}
 \tag{2.10}$$

- (x_i, y_i) : ラップ管表面の節点座標値[m]
- i : ラップ管表面の節点番号[-] ($i=1, 2, 3, \dots, n_d$)
- n_d : ラップ管表面の節点数[-]
- (X_n, Y_n) : ラップ管頂点の座標値[m]
- D_d : ラップ管内対面間距離[m]
- $[\zeta_i]$: ζ_i の Floor 関数值 (ζ_i 以下の最大の整数値)[-]

2.2.2 構造材表面の節点座標値の補正

燃料ピンとラップ管の間隔が密な燃料集合体では、ラップ管表面を等間隔の要素分割とすると、その周辺領域の要素に歪が生じる場合がある。そのような場合、ラップ管表面の節点座標値に以下の補正を適用すると、要素の歪を緩和できる。

- (1) ラップ管表面を対称軸として、ラップ管表面に隣接する要素を流体側から構造材側に折り返した仮想的な要素を生成する。
- (2) ラップ管表面の節点座標値 (x_i, y_i) に対して、ラップ管表面に隣接する要素と前項で生成した仮想的な要素を利用して、式 (2.3) のポアソン方程式を適用する。
- (3) ポアソン方程式で修正された節点座標値に関して、ラップ管表面に投影した節点座標値 (x'_i, y'_i) を算出する。
- (4) ラップ管表面の要素幅に関して、修正された節点座標値 (x'_i, y'_i) による要素幅 $\Delta l'_i$ が、ラップ管表面を均等に分割した際の要素幅 Δl_d の e_d 倍以下となるように節点座標値 (x'_i, y'_i) を修正する。

$$\begin{aligned}
 \Delta \bar{l}'_i &= \sqrt{\Delta x_i'^2 + \Delta y_i'^2}, & \Delta x_i' &= x'_{i+1} - x'_i, & \Delta y_i' &= y'_{i+1} - y'_i \\
 \Delta l'_i &= \min(\Delta \bar{l}'_i, e_d \Delta l_d), & \Delta l_d &= \frac{2\sqrt{3}D_d}{n_d}
 \end{aligned}
 \tag{2.11}$$

式 (2.11) の補正は、SOR 法の各反復過程で適用される。メッシュ緩和係数 e_d は、入力データで指定され、 $e_d=1$ を指定すると、等間隔の要素分割となる。また、同様の補正は、燃料ピン表面の節点座標値に対しても適用できる。

2.2.3 構造材要素の生成

自然循環時等の伝熱流動解析では、構造材領域の熱伝導が現象評価に際して重要となる場合がある。そのような場合は、構造材領域でも要素を生成して、同領域の熱伝導を再現する必要がある。

構造材領域の要素としては、以下の領域で要素を生成できる。

- (1) 被覆管領域
- (2) 燃料ギャップ領域
- (3) 燃料ペレット領域
- (4) ワイヤ領域
- (5) ラップ管領域
- (6) インターラップ領域（隣接する燃料集合体とのギャップ部）

インターラップ領域は流体領域であるが、要素を生成する過程はラップ管領域と同様であるので、本報では構造材領域に分類する。

燃料ピン領域とワイヤ領域の要素は、構造材の中心点を原点とした同心円状と放射線状の格子線で形成される。特異点となる構造材の中心点に関しては、三次元要素のスキームの関係から中空の領域を設定する。また、同様にして、ラップ管領域とインターラップ領域の要素は、同心多角形状と放射線状の格子線で形成される。

2.2.4 境界層要素の生成

高レイノルズ数型乱流モデルによる乱流解析では、壁に隣接する要素に対して壁関数が適用される。壁関数の適用範囲は、壁無次元座標で $30 \leq y^+ \leq 200$ であり、定格条件の燃料集合体 ($Re=50,000$ 程度) では、 $20 \mu\text{m} \leq y \leq 200 \mu\text{m}$ 程度に相当する。そのため、要素分割の際に同条件を満足する薄い境界層要素を生成すると、数値解析の予測精度を向上させられる。

図 2.6.1 及び図 2.6.2 に構造材表面で生成される境界層要素を示す。境界層要素を生成する場合は、流体要素と壁境界の間に指定された要素幅の薄い要素を配置する。境界層要素の流体側節点に対しては、式 (2.3) のポアソン方程式を適用するが、SOR 法の各反復過程で壁からの距離が指定された要素幅以下になるように節点座標値を補正する。すなわち、ポアソン方程式から算出された流体側節点の座標値を (x_j, y_j) 、対応する固体側節点の座標値を (x_s, y_s) 、指定された要素幅を Δl_w とすると、補正後の座標値 (x'_j, y'_j) は、次式で設定される。

$$\left. \begin{aligned}
 x'_f &= x_s + \gamma \Delta x, & y'_f &= y_s + \gamma \Delta y, & \gamma &= \frac{\min(\Delta l_w, \Delta r)}{\Delta r} \\
 \Delta x &= x_f - x_s, & \Delta y &= y_f - y_s, & \Delta r &= \sqrt{\Delta x^2 + \Delta y^2}
 \end{aligned} \right\} \dots \quad (2.12)$$

2.2.5 双子節点の生成

燃料ピンに螺旋状に巻かれたワイヤは、燃料ピンと線接触の状態であるため、接触位置の両側の流体は、流動的に分割されており、直接的に干渉しない。他方、物理量の定義点を節点とするレギュラー格子では、燃料ピン-ワイヤ接点に対して単一の節点を配置すると、同節点では単一の圧力しか定義されないため、接触位置の両側の流体は、圧力を介して干渉する結果となる。

その対策として、本メッシャーでは、燃料ピン-ワイヤ接点に対して位置的に重複する2節点を配置できる。すなわち、図 2.2.2 で解説すると、接点である A_0 に対して、 A_1 側の節点 A_{0-1} と A_6 側の節点 A_{0-6} の2節点を配置できる。 A_{0-1} と A_{0-6} は、節点座標値は同一であるが、 A_{0-1} では A_1 側の圧力を、 A_{0-6} では A_6 側の圧力をそれぞれ定義できるため、燃料ピン-ワイヤ接点の両側で圧力が干渉することはない。

以降、上述した位置的に重複する2節点の A_{0-1} と A_{0-6} を双子節点と表記する。双子節点は、レギュラー格子を採用する SPIRAL の解析で必要となるが、それ以外の汎用熱流体解析ソフトウェア FLUENT 等で解析を実行する際は必要なく、燃料ピン-ワイヤ接点に対して単一の節点を配置すれば良い。燃料ピン-ワイヤ接点に対して、単一節点を配置するか、双子節点を配置するかは、入力データで指定する。

2.3 三次元要素の生成

各水平断面で生成した二次元の境界適合格子に関して、隣接する水平断面間で対応する二次元要素を結合して、三次元要素を生成する。基本的な要素パターンとしては、四角形 (Quad) 要素と Quad 要素の結合により、六面体 (Hexa) 要素を生成する。

二次元要素の結合に際しては、ワイヤの影響を考慮に入れる必要がある。燃料ピンに螺旋状に巻かれたワイヤは、鉛直方向に対して傾斜を有する。そのため、ワイヤを起点として水平断面間で二次元要素を結合すると、生成される三次元要素は、すべて鉛直方向に対して傾斜したものとなる。その結果、隣接するブロック領域間では、境界での要素の傾斜が逆向きとなるため、三次元的な要素の連続性に不整合が生じる。

その不整合を回避するため、図 2.7.1 及び図 2.7.2 に示す通り、ワイヤに隣接する要素には Hexa 要素に換えてプリズム状の (Prism) 要素を配置する。斜面を有する Prism 要素をワイヤに隣接して配置することで、ワイヤの傾斜の影響をその隣接領域に限定でき、結果的に、それ以外の領域

には傾斜のない Hexa 要素を配置できる。なお、要素の連続性を維持するため、Prism 要素を配置した領域では、ブロック領域の径方向に隣接するすべての要素に Prism 要素を配置する必要がある。また、本スキームの制約から、ワイヤ巻き付けピッチあたりの鉛直方向メッシュ分割数は、燃料ピン周辺領域の周方向メッシュ分割数と同値となる。

2.4 領域データ・境界データの設定

前節で生成された三次元要素に対しては、流体または各種構造材の領域データが付与される。また、各領域の境界に対しては、流入境界・流出境界等の境界データが付与される。図 2.8.1 及び図 2.8.2 に領域データの一覧を、図 2.9.1 及び図 2.9.2 に境界データの一覧をそれぞれ示す。

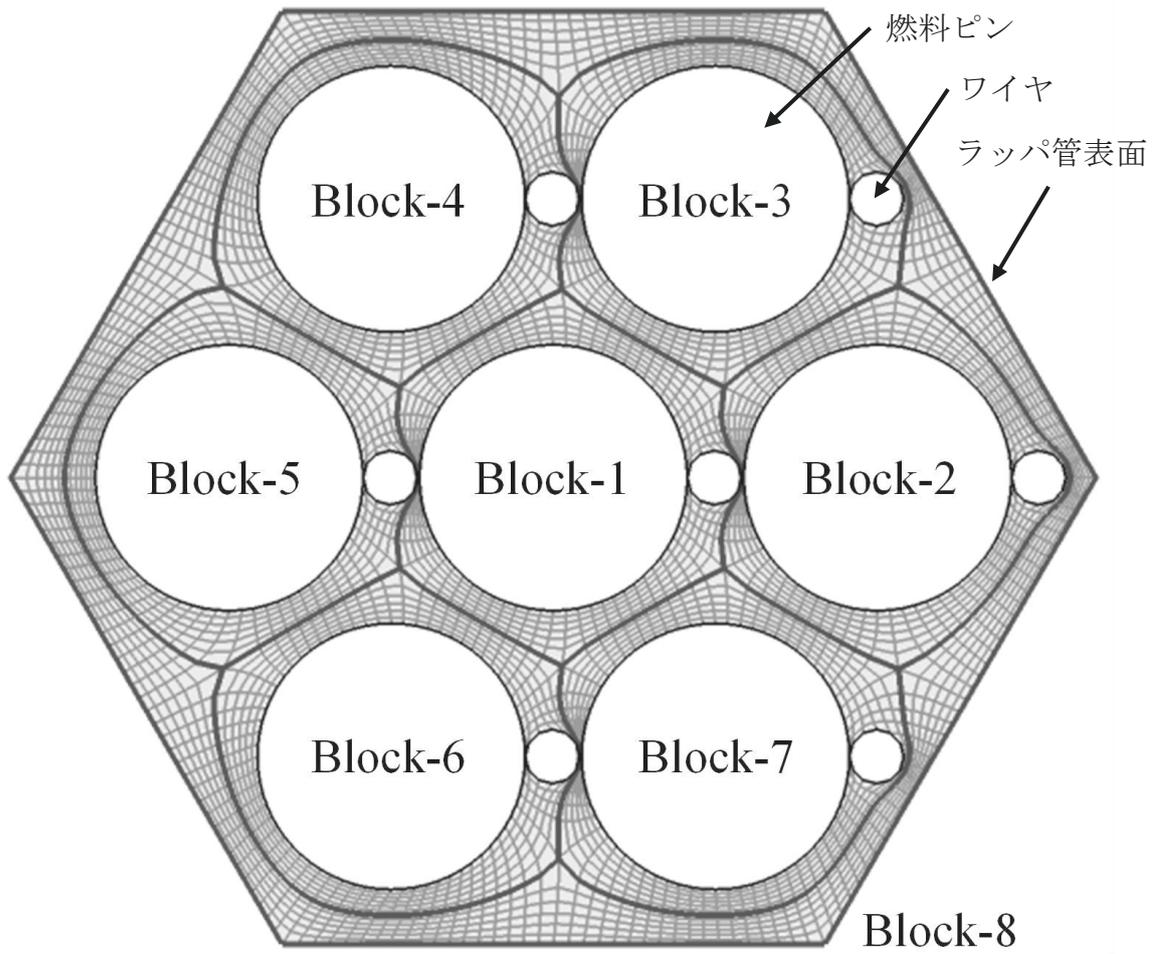
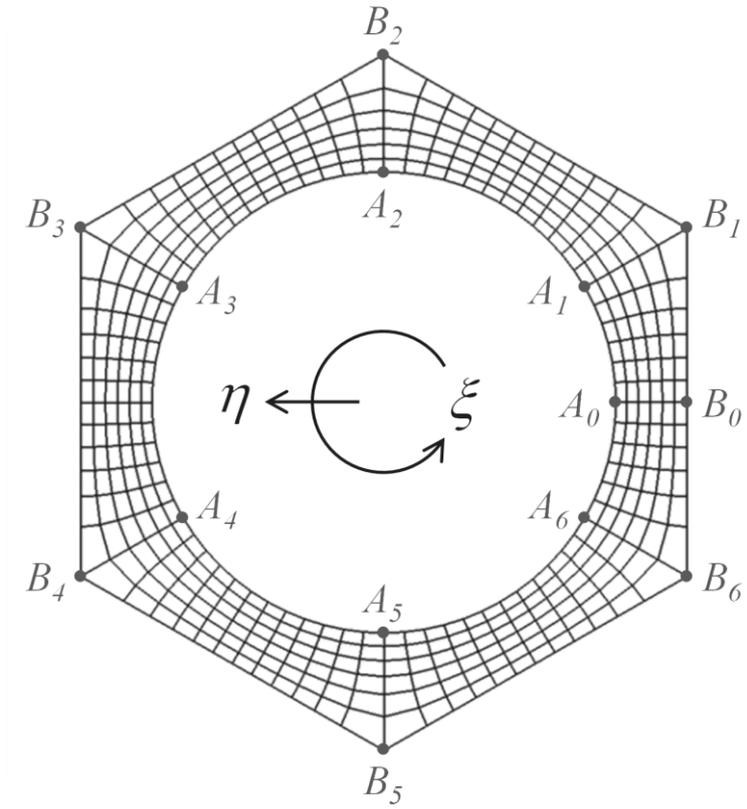
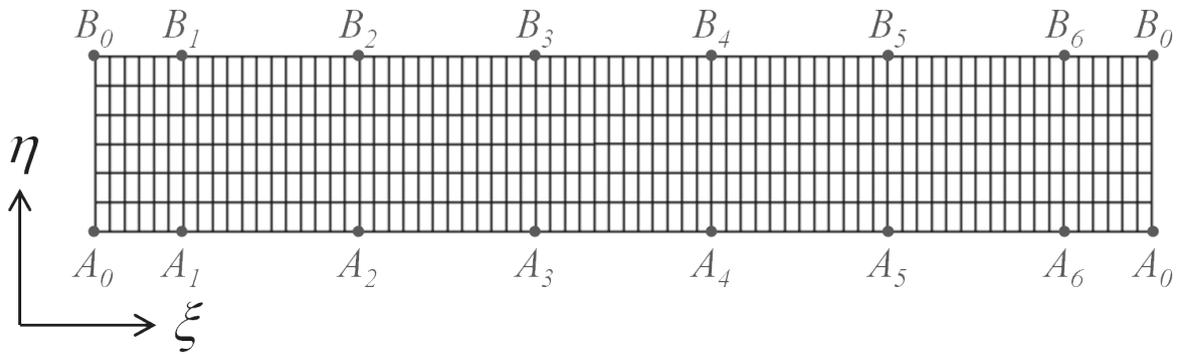


図 2.1 燃料集合体領域のブロック分割 (7 本ピン集合体)



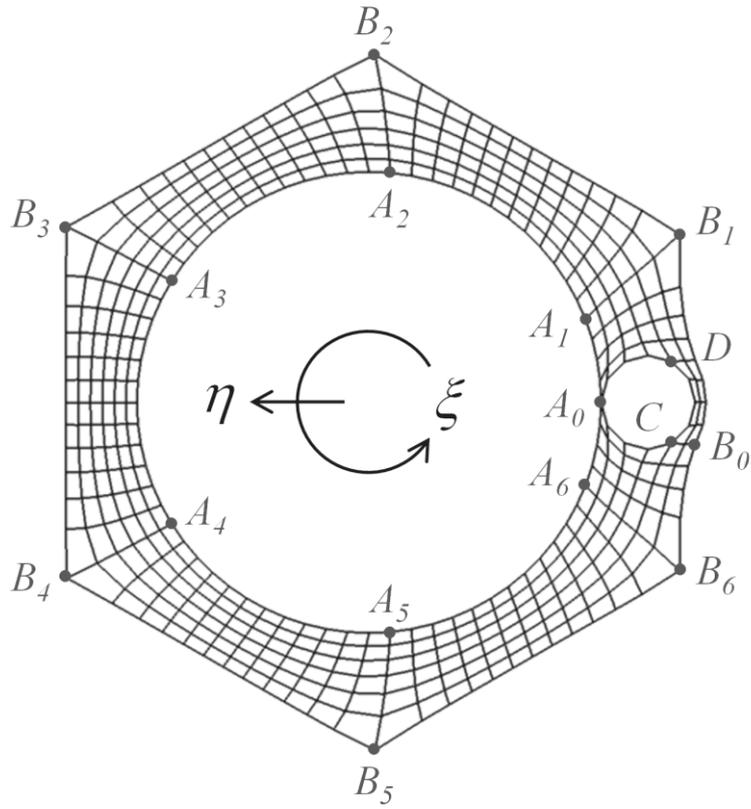
物理空間における要素分割スキーム



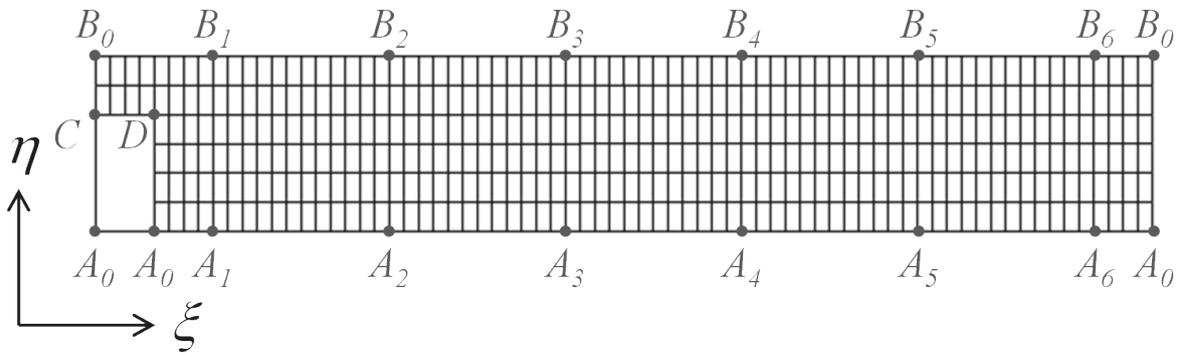
(表示倍率 $\xi:\eta=1:2$)

写像空間における要素分割スキーム

図 2.2.1 燃料ピン周辺ブロック領域における境界適合格子
(ワイヤなし条件)



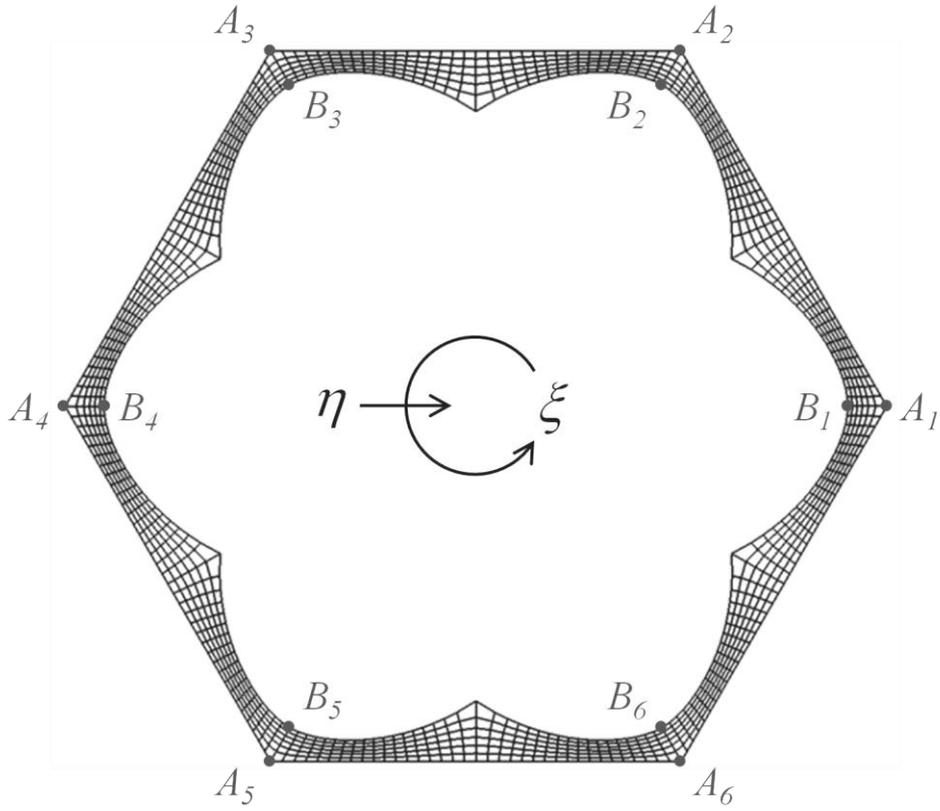
物理空間における要素分割スキーム



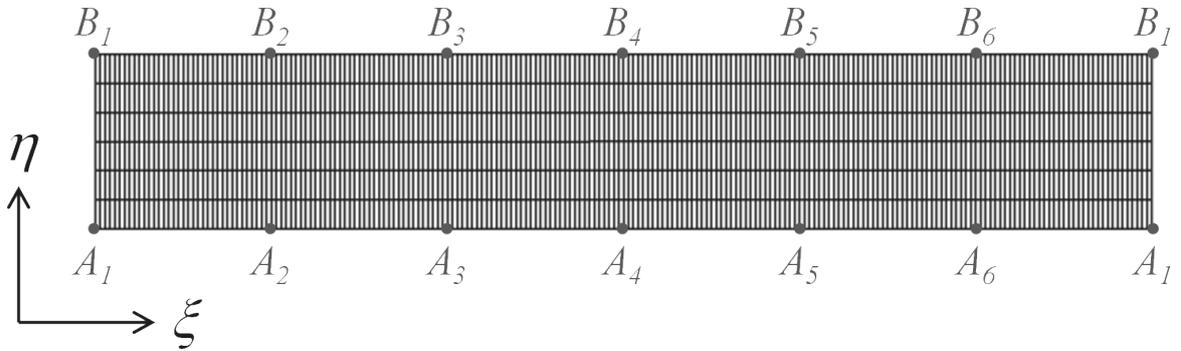
(表示倍率 $\xi:\eta=1:2$)

写像空間における要素分割スキーム

図 2.2.2 燃料ピン周辺ブロック領域における境界適合格子
(ワイヤあり条件)



物理空間における要素分割スキーム



(表示倍率 $\xi:\eta=1:6$)

写像空間における要素分割スキーム

図 2.2.3 ラップ管周辺ブロック領域における境界適合格子

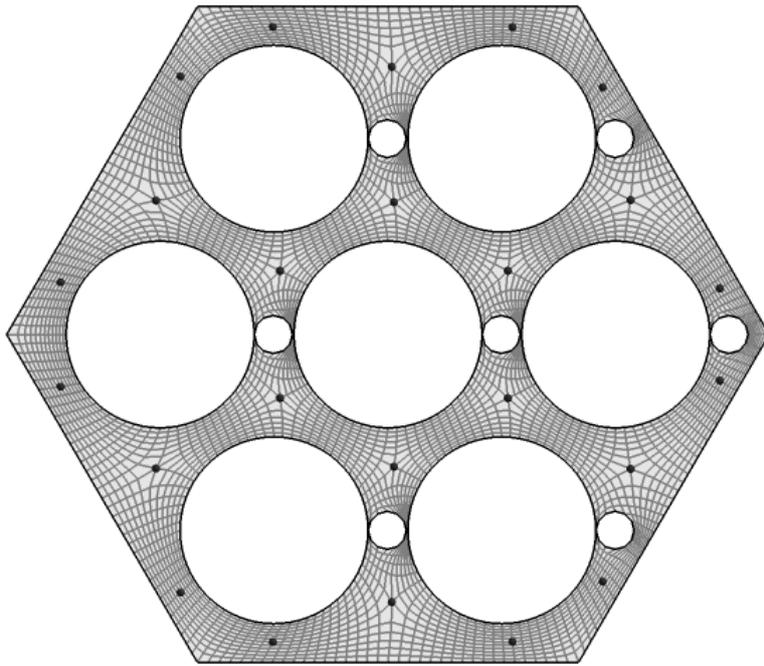


図 2.3.1 サブチャンネル中心点上の節点

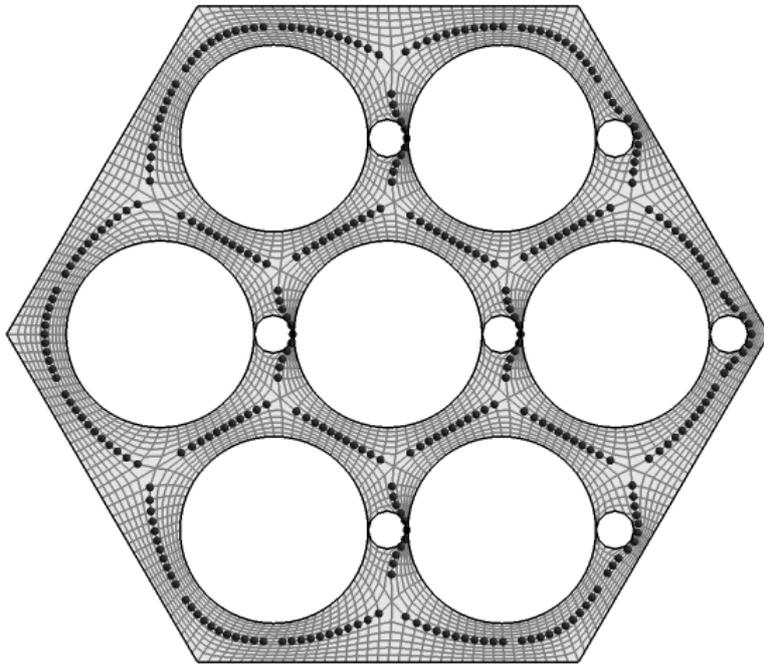


図 2.3.2 ギャップ線上の節点

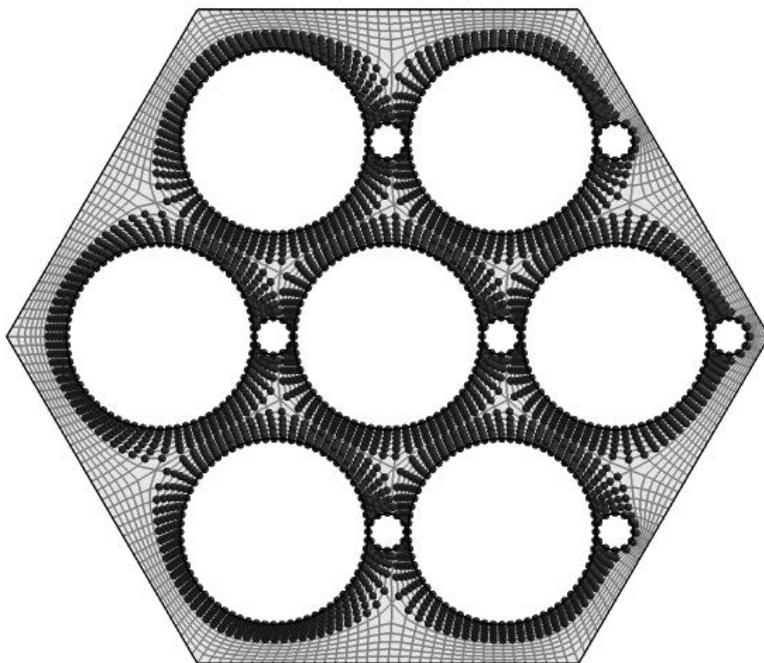


図 2.3.3 燃料ピン周辺領域上の節点

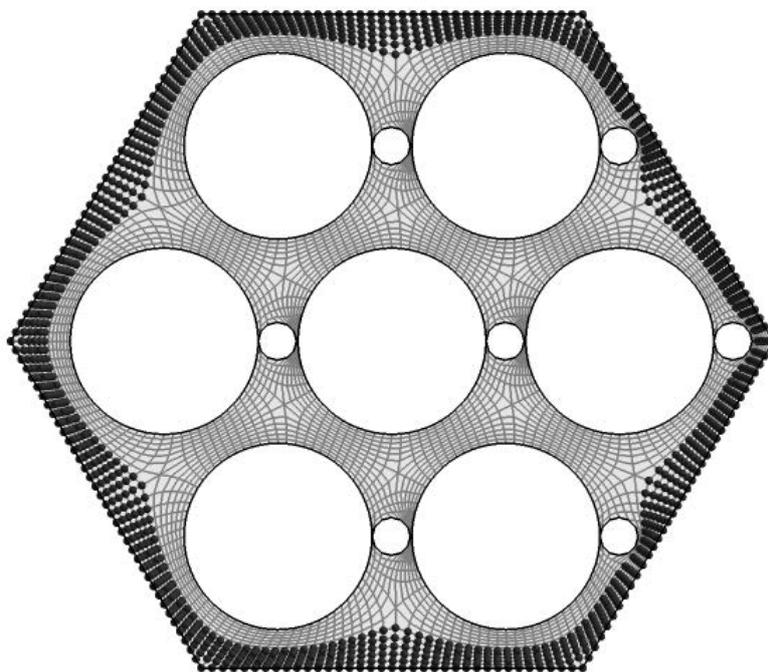


図 2.3.4 ラッパ管周辺領域上の節点

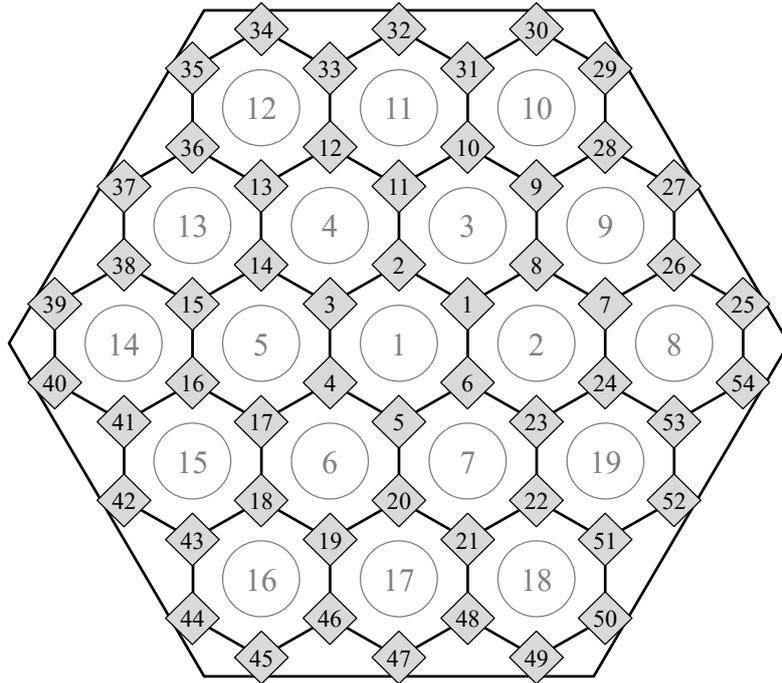


図 2.4.1 19 本ピン集合体におけるサブチャンネル中心点番号

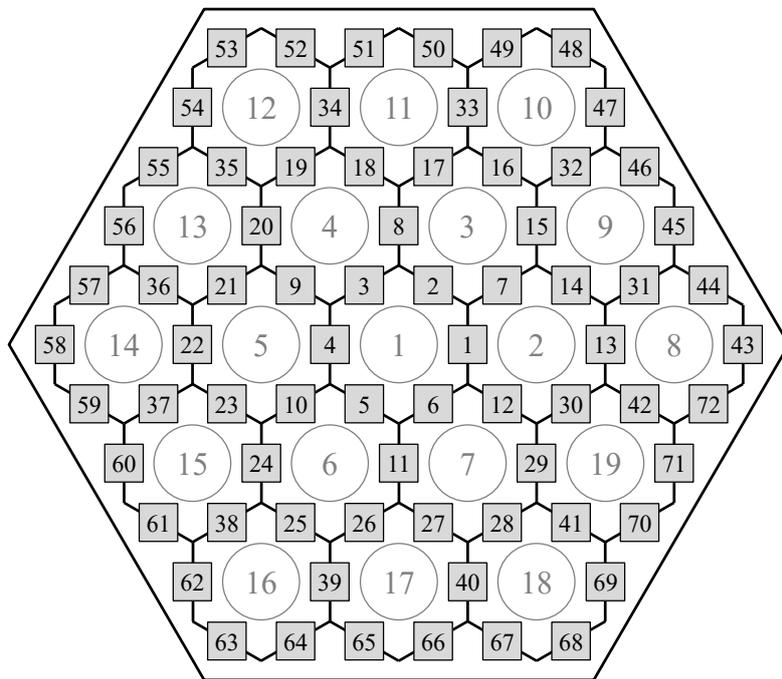


図 2.4.2 19 本ピン集合体におけるギャップ線番号

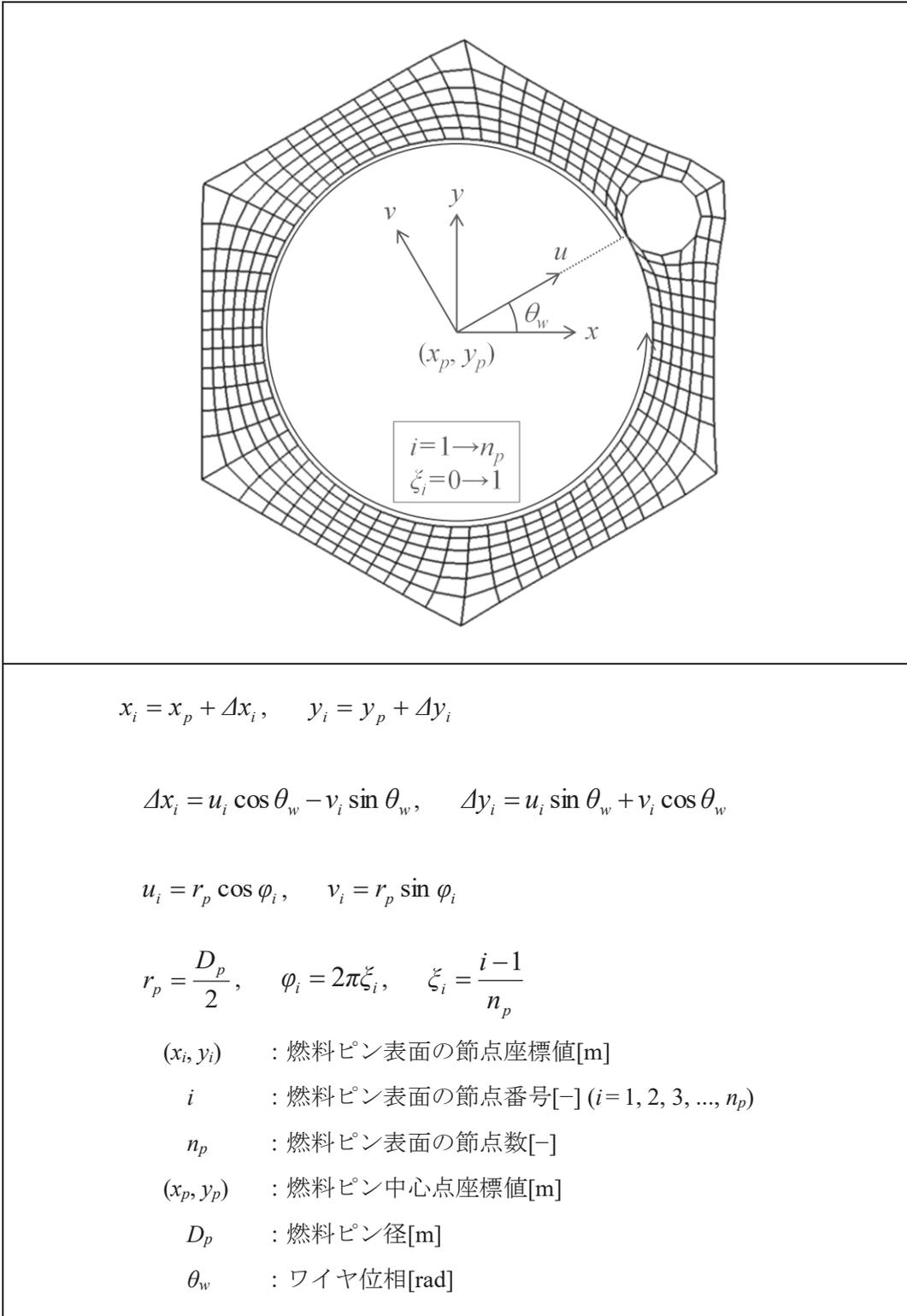
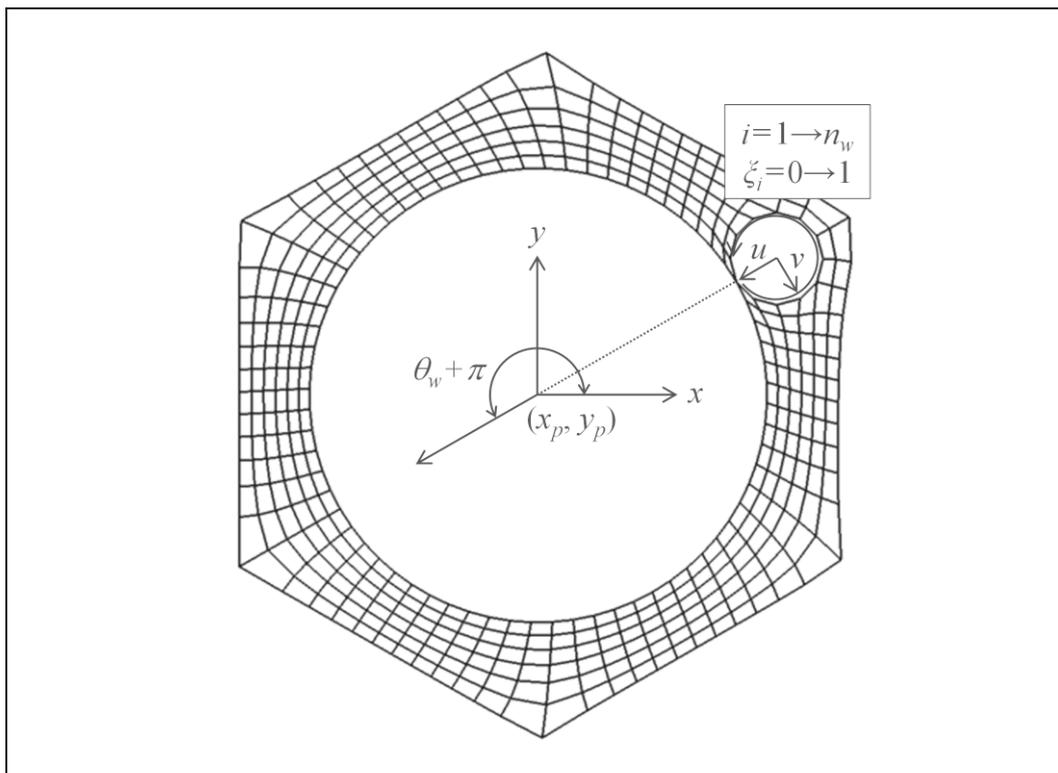


図 2.5.1 燃料ピン表面の節点座標値の設定



$$x_i = x_w + \Delta x_i, \quad y_i = y_w + \Delta y_i$$

$$x_w = x_p + (r_p + r_w) \cos \theta_w, \quad y_w = y_p + (r_p + r_w) \sin \theta_w$$

$$-\Delta x_i = u_i \cos \theta_w - v_i \sin \theta_w, \quad -\Delta y_i = u_i \sin \theta_w + v_i \cos \theta_w$$

$$u_i = r_w \cos \varphi_i, \quad v_i = \frac{r_w}{c} \sin \varphi_i, \quad r_w = \frac{D_w}{2}, \quad \varphi_i = 2\pi \xi_i$$

$$\xi_i = \frac{i-1}{n_w}, \quad c = \frac{H}{\sqrt{H^2 + L^2}}, \quad L = \pi(D_p + D_w)$$

(x_i, y_i) : ワイヤ表面の節点座標値[m]

i : ワイヤ表面の節点番号[-] ($i=1, 2, 3, \dots, n_w$)

n_w : ワイヤ表面の節点数[-]

(x_w, y_w) : ワイヤ中心点座標値[m]

D_w : ワイヤ径[m]

図 2.5.2 ワイヤ表面の節点座標値の設定

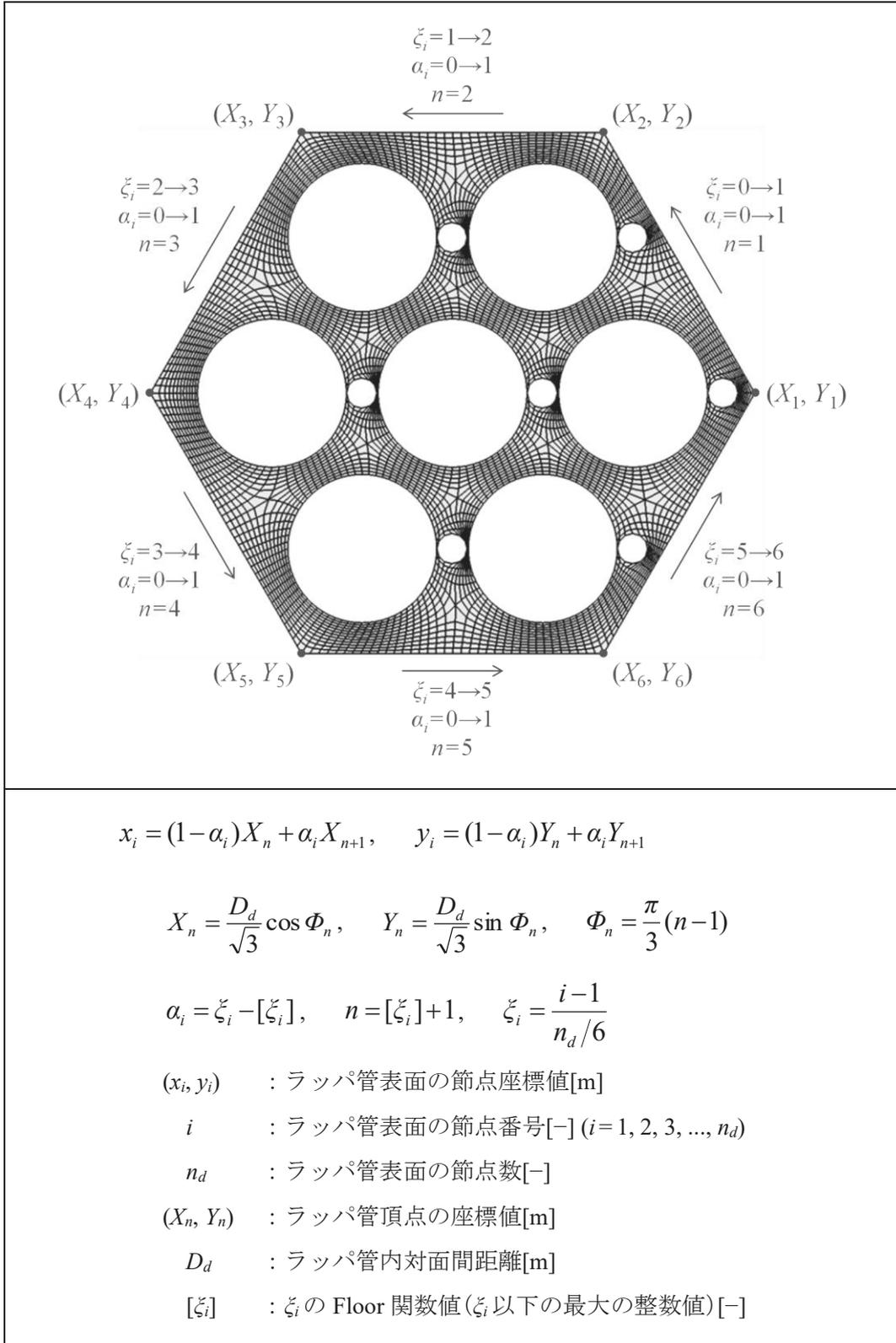


図 2.5.3 ラップ管表面の節点座標値の設定

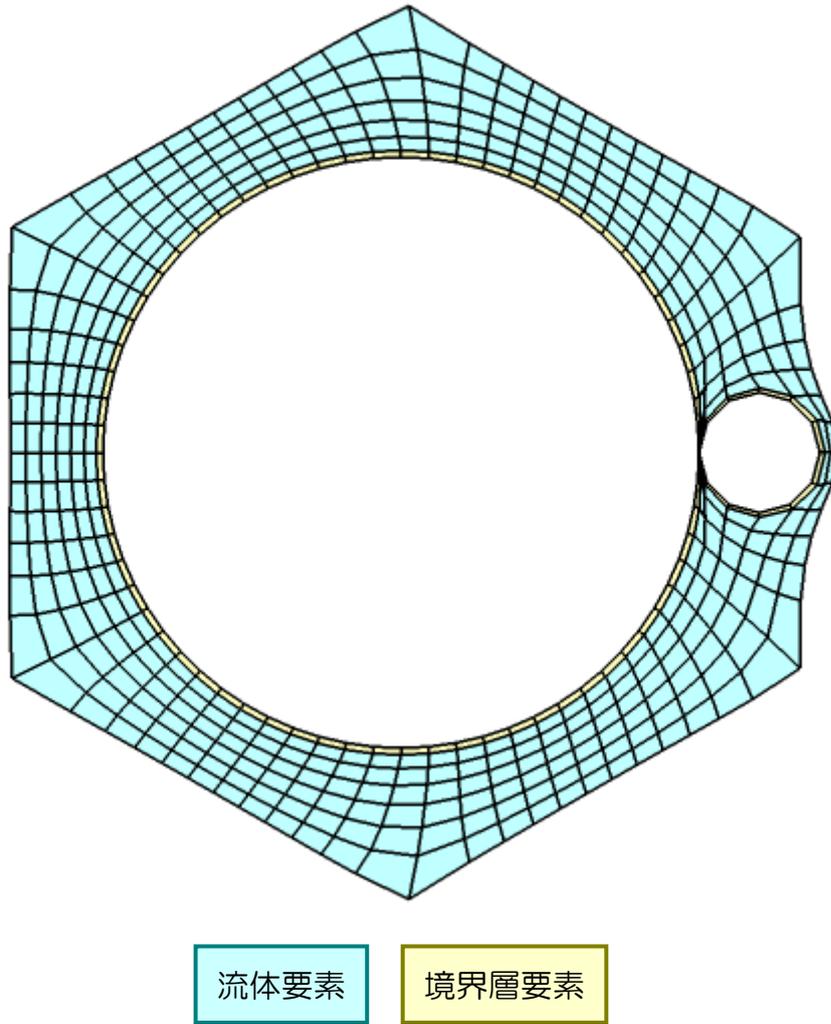


図 2.6.1 燃料ピン表面・ワイヤ表面における境界層要素の生成

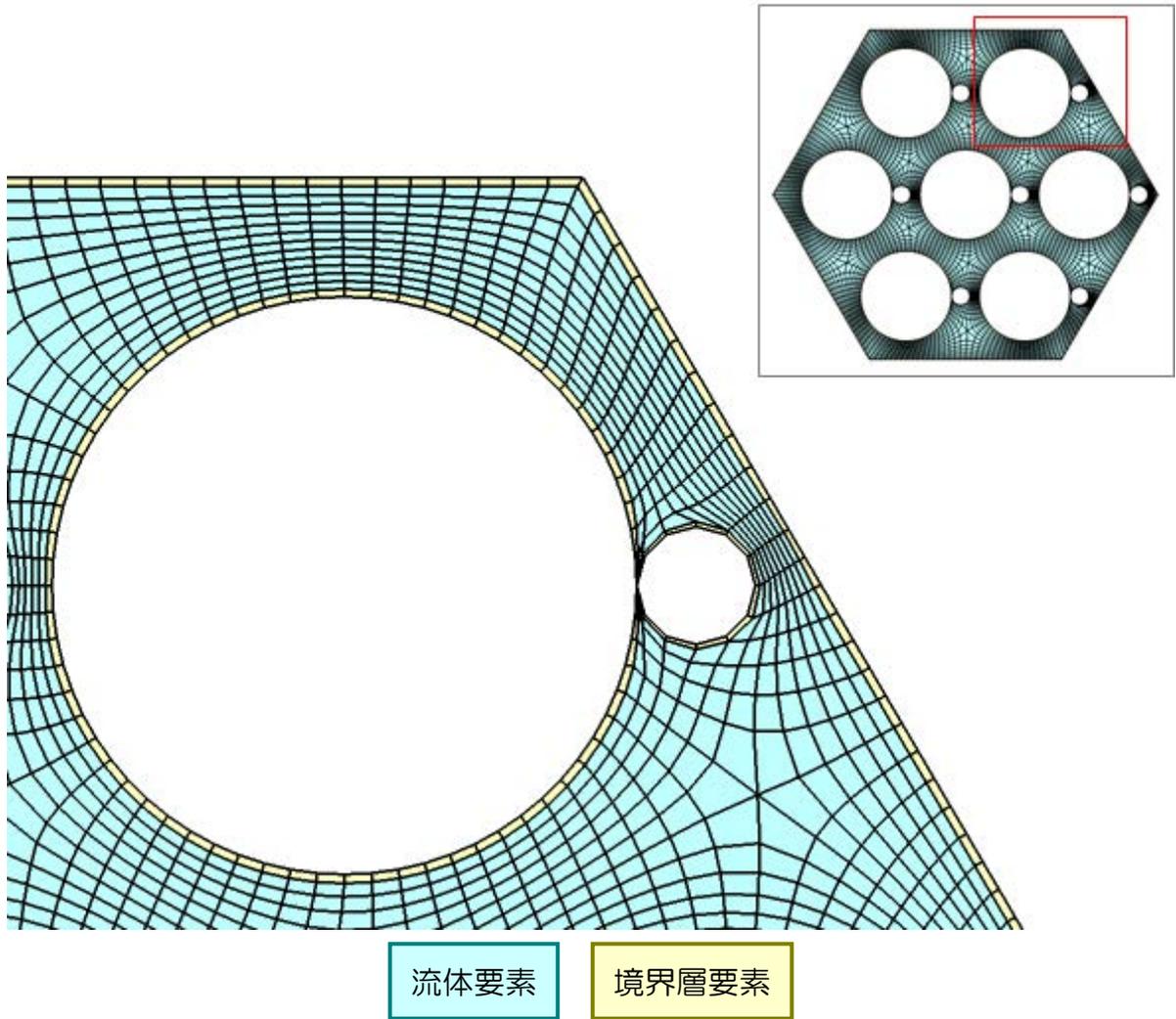


図 2.6.2 ラップ管表面における境界層要素の生成
(ラップ管頂点付近の拡大)

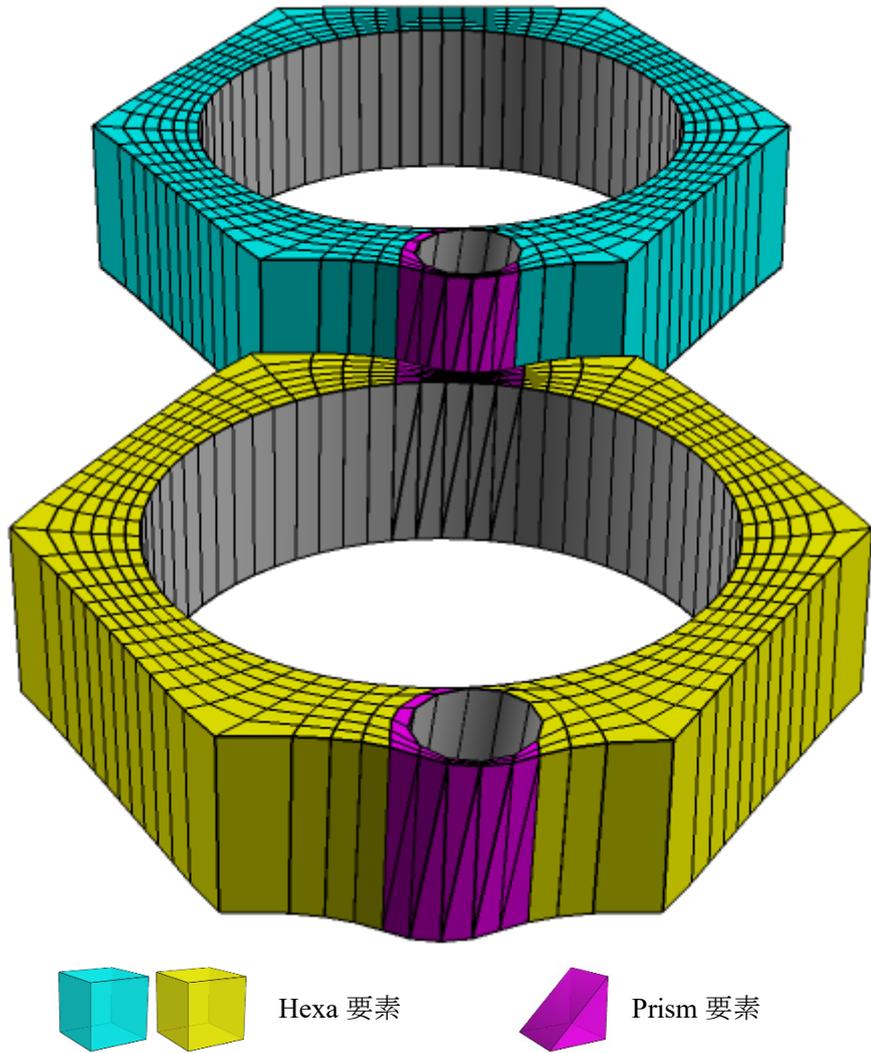
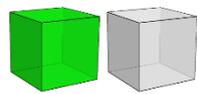
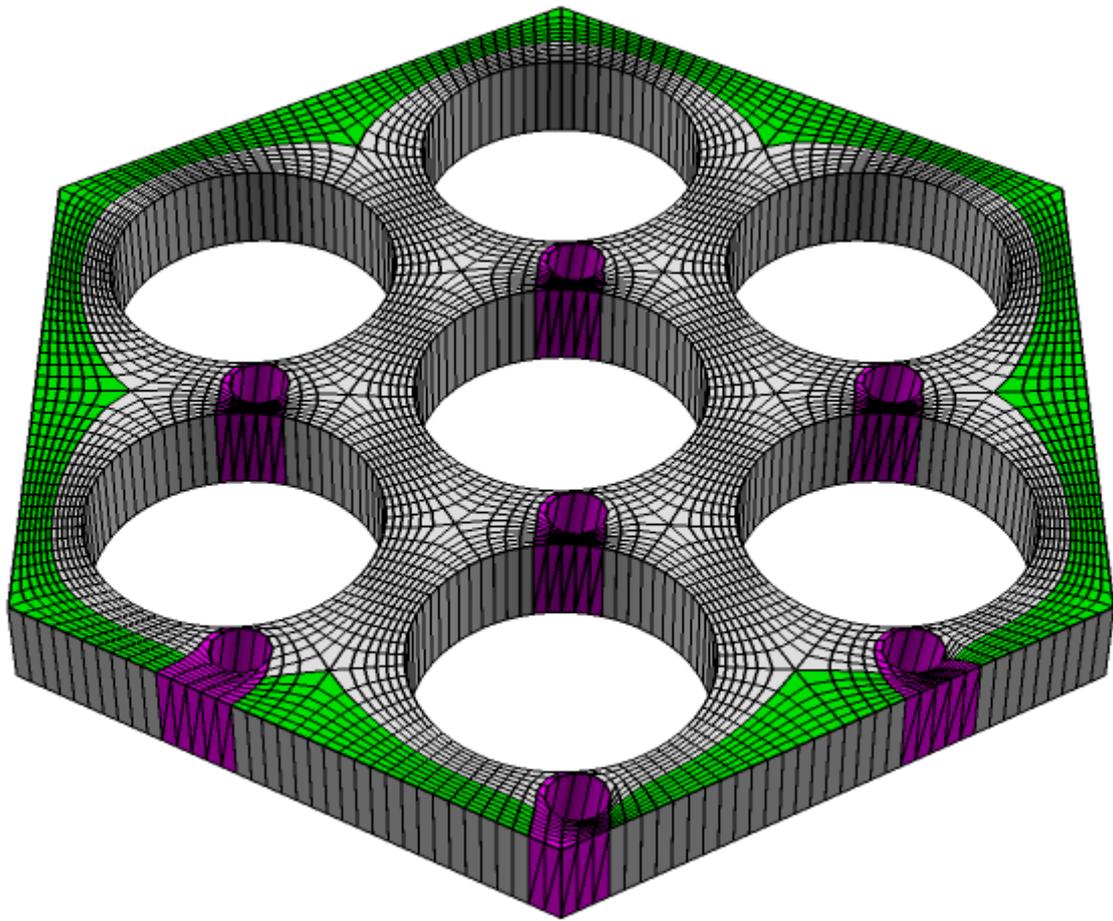


図 2.7.1 燃料ピン周辺ブロック領域における三次元要素の生成

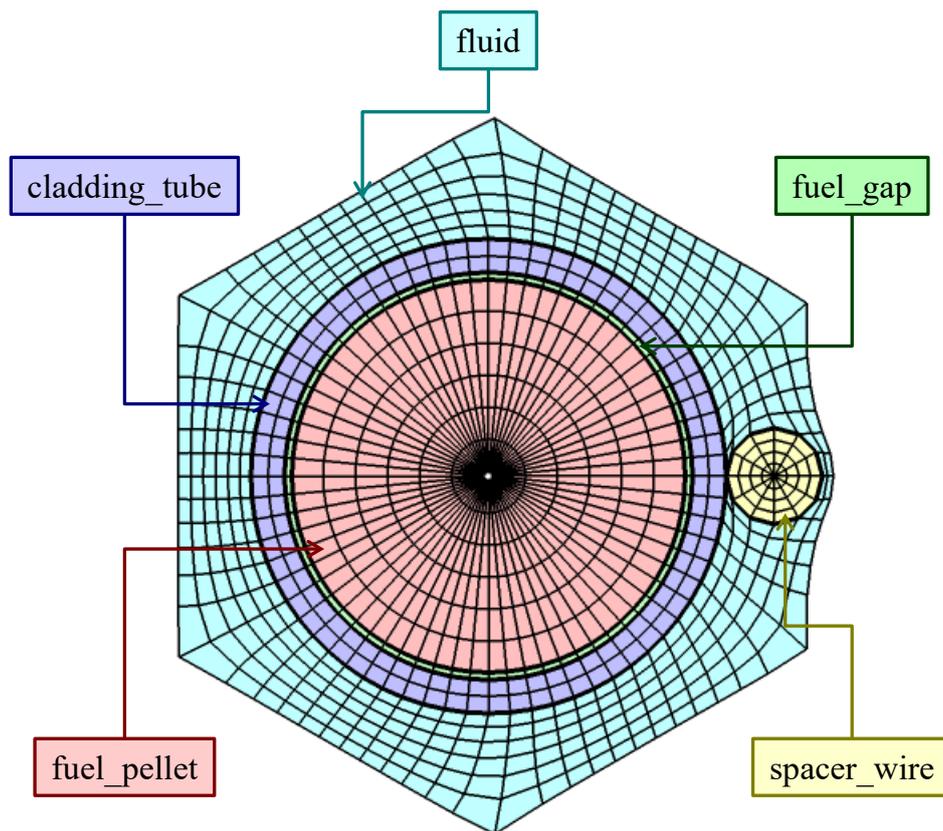


Hexa 要素



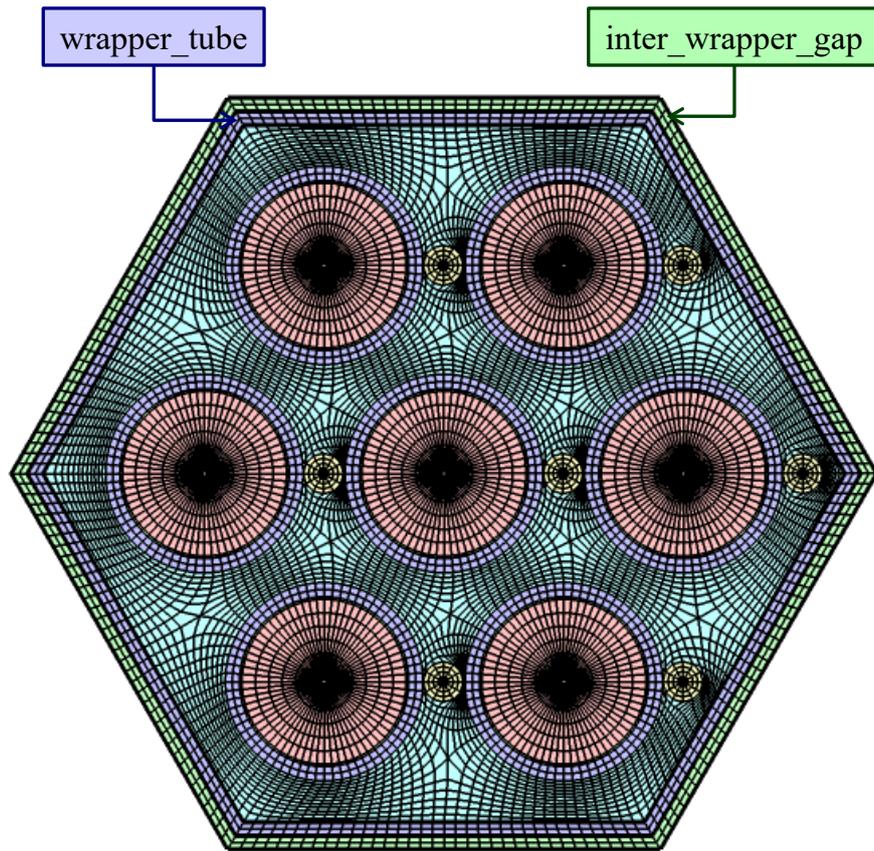
Prism 要素

図 2.7.2 ラッパ管周辺ブロック領域における三次元要素の生成



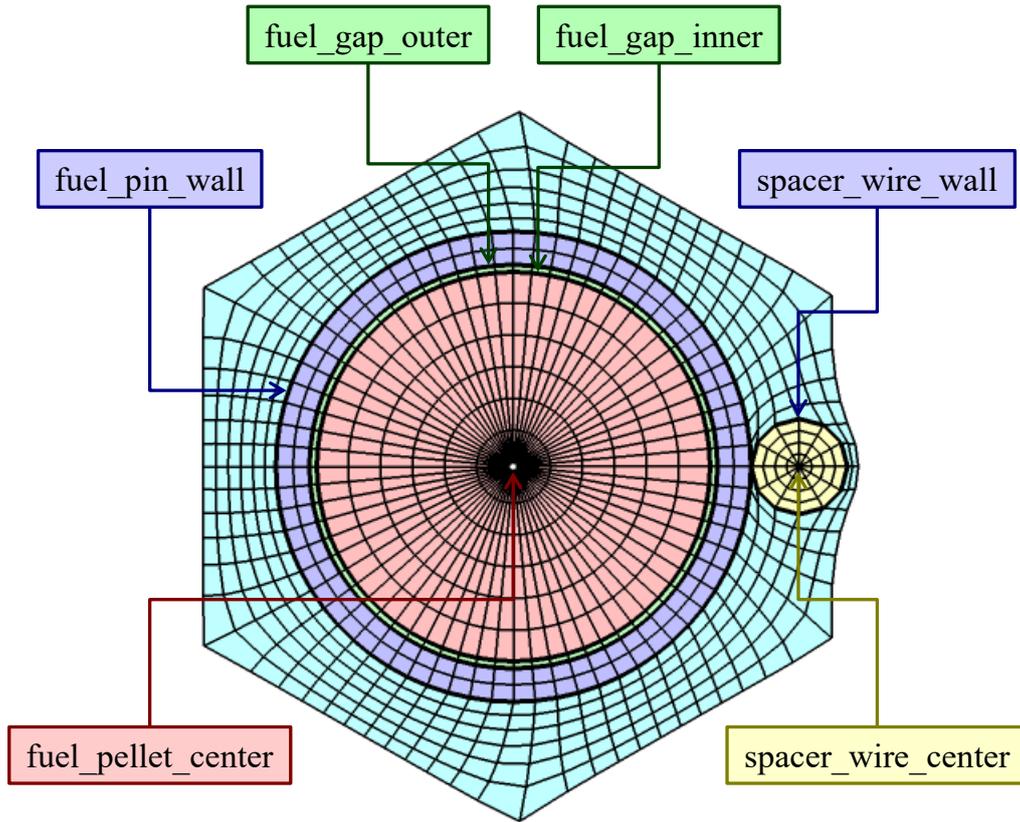
領域データ名称	領域データ解説	領域データ属性
fluid	流体領域	流体
cladding_tube	被覆管領域	構造材
fuel_gap	燃料ギャップ領域	構造材
fuel_pellet	燃料ペレット領域	構造材
spacer_wire	ワイヤ領域	構造材

図 2.8.1 燃料ピン周辺ブロック領域における領域データ



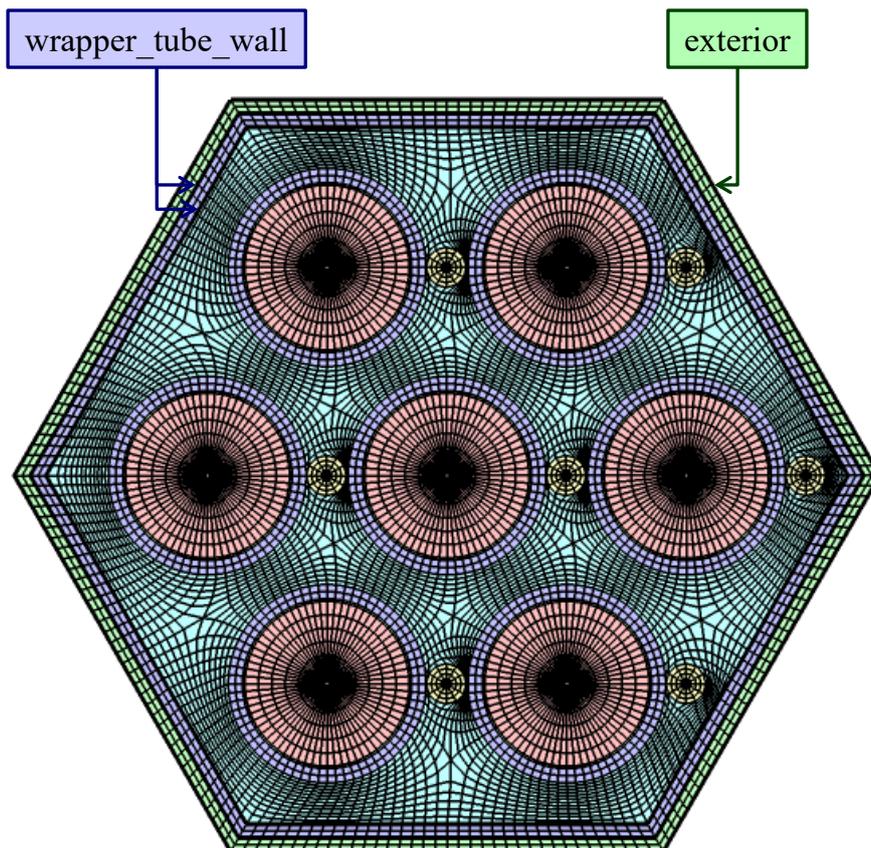
領域データ名称	領域データ解説	領域データ属性
wrapper_tube	ラップ管領域	構造材
inter_wrapper_gap	インターラップ領域	流体

図 2.8.2 ラップ管周辺ブロック領域における領域データ



境界データ名称	境界データ解説	隣接領域データ
inlet	流入境界	全領域
outlet	流出境界	全領域
fuel_pin_wall	被覆管外表面境界	fluid, cladding_tube
spacer_wire_wall	ワイヤ表面境界	fluid, spacer_wire
fuel_gap_outer	被覆管内表面境界	cladding_tube, fuel_gap
fuel_gap_inner	燃料ペレット表面境界	fuel_gap, fuel_pellet
fuel_pellet_center	燃料ペレット中心境界	fuel_pellet
spacer_wire_center	ワイヤ中心境界	spacer_wire

図 2.9.1 燃料ピン周辺ブロック領域における境界データ



境界データ名称	境界データ解説	隣接領域データ
wrapper_tube_wall	ラップ管表面境界	fluid, wrapper_tube
exterior	インターラップ側面境界	inter_wrapper_gap

図 2.9.2 ラップ管周辺ブロック領域における境界データ

3. 燃料集合体領域の随意的な要素分割モデル

前章では、燃料集合体領域の基本的な要素分割モデルを解説した。本章では、様々なニーズに対応するため、随意的なモデルとして開発した、構造材変形モデル、構造材接触モデル、内部ダクト付き燃料集合体モデル、局所閉塞モデル等の要素分割モデルについて解説する。

3.1 構造材変形モデル

燃料ピンの偏心・膨張等による構造材変形時の燃料集合体では、2.2.1 項に記述した構造材表面の節点座標値を設定する際にその情報を反映させる。

一例として、燃料ピンの偏心時は、式 (2.8) に現れる燃料ピン中心点座標値 (x_p, y_p) に対してノミナル値からの偏心量 $(\Delta x_p, \Delta y_p)$ を加算する。また、燃料ピンの膨張時は、式 (2.5) に現れる燃料ピン径 D_p に対して膨張量 ΔD_p を加算する。同様に、ワイヤの偏心時・膨張時は、式 (2.9) に現れるワイヤ中心点座標値 (x_w, y_w) 、ワイヤ径 D_w に対してノミナル値からの偏心量 $(\Delta x_w, \Delta y_w)$ 、膨張量 ΔD_w を加算する。ただし、要素スキームの制約から、燃料ピンとワイヤは接触している必要があるため、図 3.1 に示す通り、最終的なワイヤ中心点座標値 (x''_w, y''_w) は、次式で設定される。

$$\begin{aligned}
 x''_w &= x'_p + (r'_p + r'_w) \cos \theta'_w, & y''_w &= y'_p + (r'_p + r'_w) \sin \theta'_w \\
 r'_p &= \frac{D'_p}{2}, & r'_w &= \frac{D'_w}{2} \\
 \cos \theta'_w &= \frac{x'_w - x'_p}{r'_p + r'_w}, & \sin \theta'_w &= \frac{y'_w - y'_p}{r'_p + r'_w} \\
 x'_p &= x_p + \Delta x_p, & y'_p &= y_p + \Delta y_p, & D'_p &= D_p + \Delta D_p \\
 x'_w &= x_w + \Delta x_w, & y'_w &= y_w + \Delta y_w, & D'_w &= D_w + \Delta D_w
 \end{aligned}
 \quad \left. \vphantom{\begin{aligned} x''_w &= x'_p + (r'_p + r'_w) \cos \theta'_w, \\ r'_p &= \frac{D'_p}{2}, \\ \cos \theta'_w &= \frac{x'_w - x'_p}{r'_p + r'_w}, \\ x'_p &= x_p + \Delta x_p, \end{aligned}} \right\} \dots (3.1)$$

ラップ管に関しては、膨張モードのみの取り扱いが可能であり、式 (2.10) に現れるラップ管内対面間距離 D_d に対して膨張量 ΔD_d を加算する。また、ラップ管肉厚 W_d に対して膨張量 ΔW_d を加算する。ラップ管の複雑な変形を考慮する場合は、3.3 節に記述する内部ダクト付き燃料集合体の要素分割機能を利用すると良い。

なお、燃料ピンの偏心時は、燃料ペレットも連動して偏心する。燃料ピンとは独立に燃料ペレットを偏心させたい場合は、燃料ピンの偏心量 $(\Delta x_p, \Delta y_p)$ に加えて、燃料ペレットの偏心量 $(\Delta x_q, \Delta y_q)$ を追加で指定する必要がある。

3.2 構造材接触モデル

燃料ピン等の変形が顕著となり、隣接する構造材間で接触が発生した場合は、その接触状態に応じて要素スキームを修正する必要がある。本メッシュャーでは、構造材の接触点を検出した上で、接触状態に応じて要素スキームを修正する一連の処理をすべて自動で実行できる。取り扱いできる接触形態としては、点接触・線接触・面接触のすべてに対応している。

構造材の接触点を検出する処理では、図 3.2.1 に示す通り、二次元要素を生成した各水平断面上において、構造材表面を構成する要素の各辺を抽出して、そのすべての辺の組み合わせで交点の有無を判定する。検出された交点は、構造材の接触点として処理される。

要素スキームを修正する処理では、節点の追加と削除、要素の再分割等の以下の処理が実行される。

- (1) 構造材の接触位置における固体節点の追加 (図 3.2.2)
- (2) 構造材の接触位置における流体節点の削除 (図 3.2.2)
- (3) 節点の追加・削除に伴う要素種類の変更 (図 3.2.3)
- (4) 低品質要素の修正 (図 3.2.4)

処理(1)では、構造材の接触点に対して新しい節点が追加され、それに関連する要素は再分割される。また、要素の連続性を維持するため、再分割された要素に対して構造材の径方向に隣接する要素も同様に再分割される。処理(2)では、構造材の接触面は従来の流体領域が塞がれるため、当該位置の流体節点は削除される。処理(3)では、節点の追加・削除に応じて、Hexa 要素→Prism 要素、Prism 要素→Pyramid 要素等の要素種類の変更が実行される。図 3.2.5 に要素種類の変換パターンを示す。また、要素の再分割処理では、アスペクト比の悪い低品質な要素が生成される場合もあるため、それらの要素を処理(4)で修正できる。処理(4)では、入力データで指定された再分割比 χ に関して、それ未満の比で要素の再分割が実行された場合は、短辺側の隣接節点を統合して一つの節点とする。なお、再分割比 χ は、再分割する前の要素の辺長を L_0 、再分割された辺長をそれぞれ L_1, L_2 ($L_1 \leq L_2$) として、 $\chi = L_1/L_0$ で定義される。

3.3 内部ダクト付き燃料集合体モデル

本節では、燃料集合体内部にダクトを設けた内部ダクト付き燃料集合体³⁾に関して、その要素分割処理を解説する。

内部ダクト付き燃料集合体を対象とした要素分割では、前章に記述した要素分割処理の過程において、以下の処理が追加で実行される。

- (1) 内部ダクトに置き換えられる燃料ピンの除去
- (2) ラップ管形状の設定

処理(1)は、内部ダクトに置き換えられる燃料ピンとそれに付随するブロック領域を無効化する処理である。無効化されたブロック領域では、2.2 節、2.3 節に記述した要素の生成処理は実行されない。また、内部的な処理として、隣接ブロック領域の接続情報を変更する処理が実行される。タイミングとしては、2.1 節に記述したブロック領域の分割後に実行される。

処理(2)は、変形等のない正常な燃料集合体では正六角管となるラップ管に関して、一つのコーナー部を凹形状とする処理である。ラップ管の各辺に配分される節点数は、正六角管型ラップ管ではすべて同数となるが、内部ダクト付きラップ管ではそれぞれの辺長に応じて節点が配分される。タイミングとしては、2.2 節に記述した境界適合格子の生成前に実行される。

内部ダクト付き燃料集合体の一例として、図 3.3.1 に内部ダクト付き 18 本ピン集合体の要素分割図を示す。なお、除去する燃料ピンとラップ管の形状は自由に設定できるため、本機能を利用して、図 3.3.2 に示す変形六角管型ラップ管 3 本ピン集合体のようなデータを作成することも可能である。

3.4 1/n セクタモデル

燃料ピンにワイヤが巻かれていないベアバンドル集合体は、30°軸対称の形状となるため、数値解析の実行に際して、1/n セクタモデルが利用できる。

本メッシュャーでは、入力データでセクタ数の n を指定すれば、1/n セクタモデルの燃料集合体メッシュデータを生成できる。プログラム内の処理順序としては、フルセクタモデルのメッシュデータを生成した後、 $0 \leq \theta \leq 2\pi/n$ の範囲外にある不必要なメッシュを削除する処理となる。また、セクタモデルの対称境界となる $\theta = 0, 2\pi/n$ の境界では、それぞれ「symmetry_min」「symmetry_max」の境界データが生成される。適用例として、図 3.4 にベアバンドル集合体のフルセクタモデルと 1/4, 1/6, 1/12 各セクタモデルの要素分割図を示す。なお、正六角管型集合体では、 $n = 2, 3, 4, 6, 12$ が有効な値である。

3.5 局所閉塞モデル

燃料集合体の流路が異物により局所的に閉塞すると、閉塞物領域及びその後流の領域が高温となり、燃料の健全性が損なわれる可能性がある。そこで、事故解析（安全評価）として局所閉塞を考慮した解析を行う必要がある。本メッシュャーの局所閉塞モデルでは、入力データで閉塞物領域として指定された空間内に存在する要素に関して、その所属する領域データを流体領域から閉

塞物領域へと変更する。閉塞物の形状にあわせた要素のスムージングには対応していない。従って、閉塞物のおおよその位置は指定できるものの、閉塞物の形状を厳密に指定することはできない。

閉塞物領域に変更された要素に対しては、領域データ「blockage」が設定される。また、「blockage」領域と他の領域の境界に対しては、境界データ「blockage_face」が設定される。

閉塞物領域の指定は、任意の数の三角柱を指定して実行する。流体領域の三次元要素に関して、その要素中心点が三角柱の内部に存在する場合、当該要素の領域データが閉塞物領域へと変更される。図 3.5 に流体領域の要素と閉塞物領域の要素を、閉塞物領域として指定した三角柱と併せて示す。

3.6 要素修正モデル

本メッシュャーでは、生成された三次元要素に関して、その品質を判定した上で、低品質な要素を自動で修正する機能がある。要素の品質は、エッジ長またはエッジ交差角で判定する。

エッジ長による判定では、指定したエッジ長未満のエッジが検出された場合、そのエッジの両端の節点を統合して、関連する要素の要素種類を変更する。燃料ピン配列ピッチが燃料ピン径とワイヤ径の和に等しい場合、すなわち、燃料ピンに巻かれたワイヤが隣接する燃料ピンと点接触している場合、3.2 節に記述した構造材接触モデルでは、数値誤差の関係で接触が検出されない場合があるため、本モデルで適当なエッジ長を指定した方が良い。

エッジ交差角による判定では、各節点におけるエッジ交差角 θ_i を算出して、その直角からの差 $|\theta_i - 90^\circ|$ が指定した角度以上となる節点が検出された場合、その節点を削除して、関連する要素の要素種類を変更する。ただし、本機能による修正では、要素の品質が必ずしも改善されるとは限らないため、留意する必要がある。

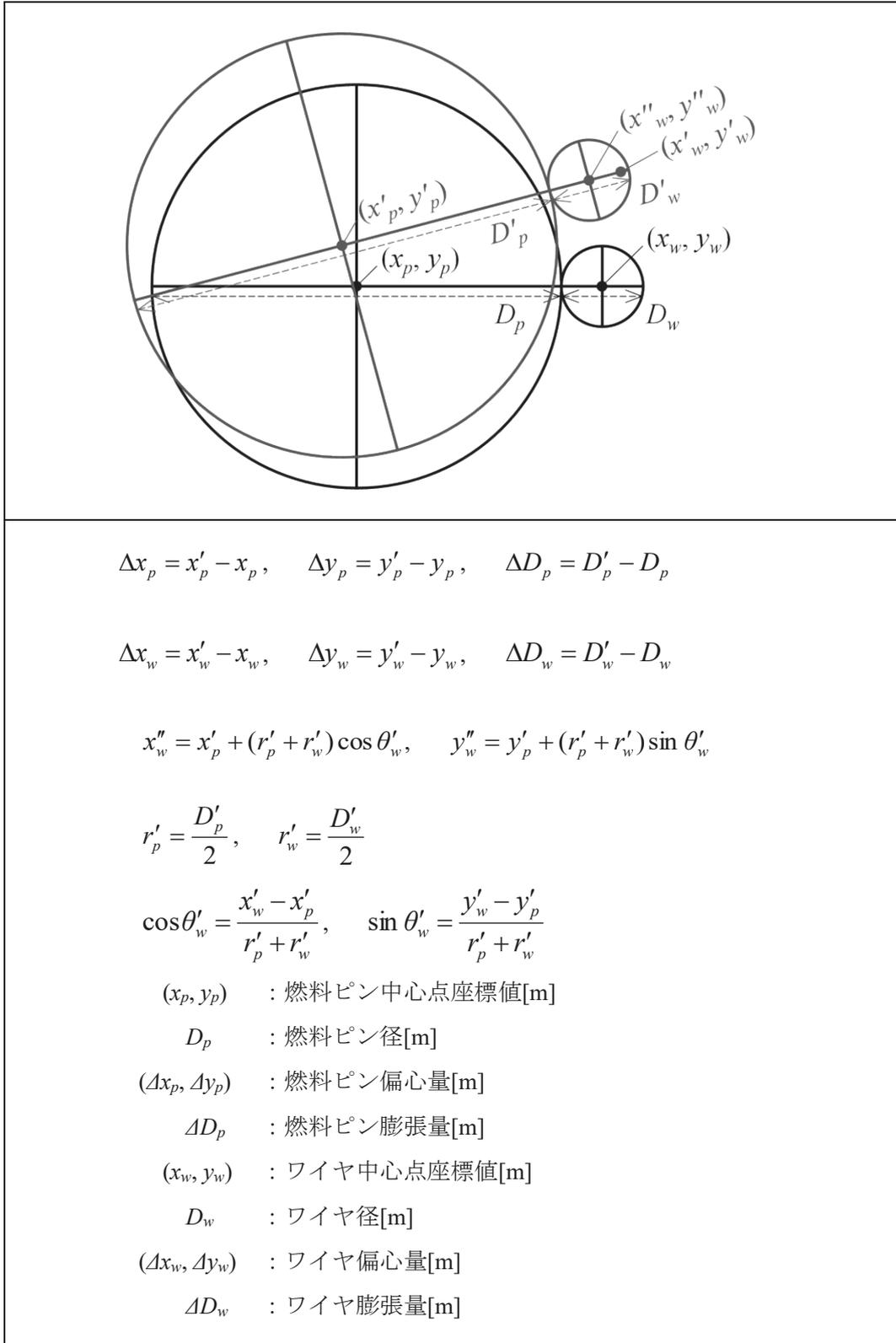
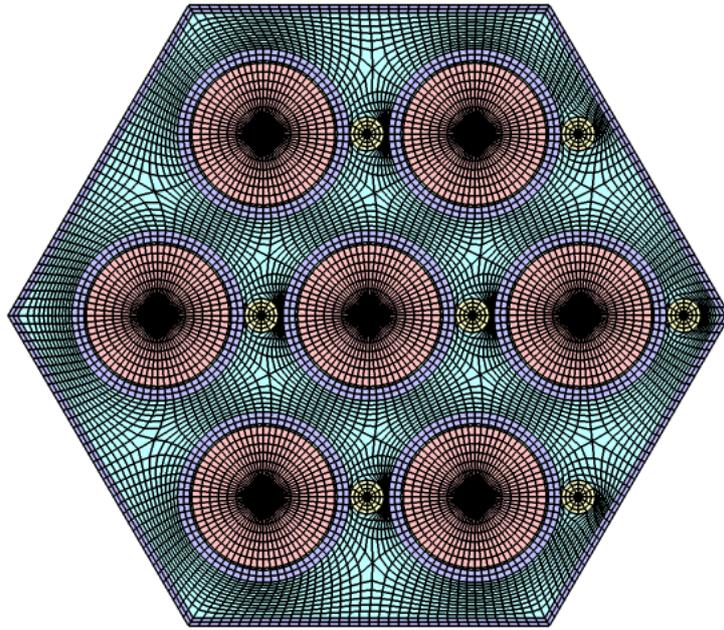
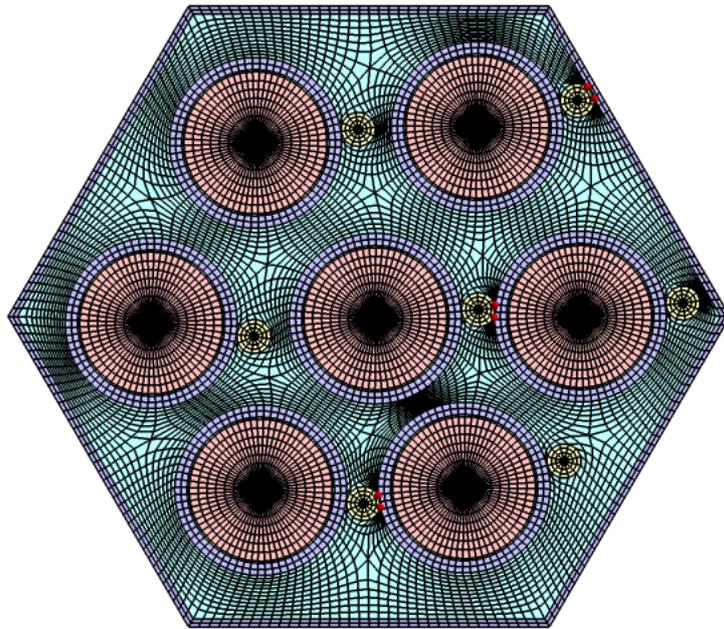


図 3.1 燃料ピン・ワイヤ変形モデル



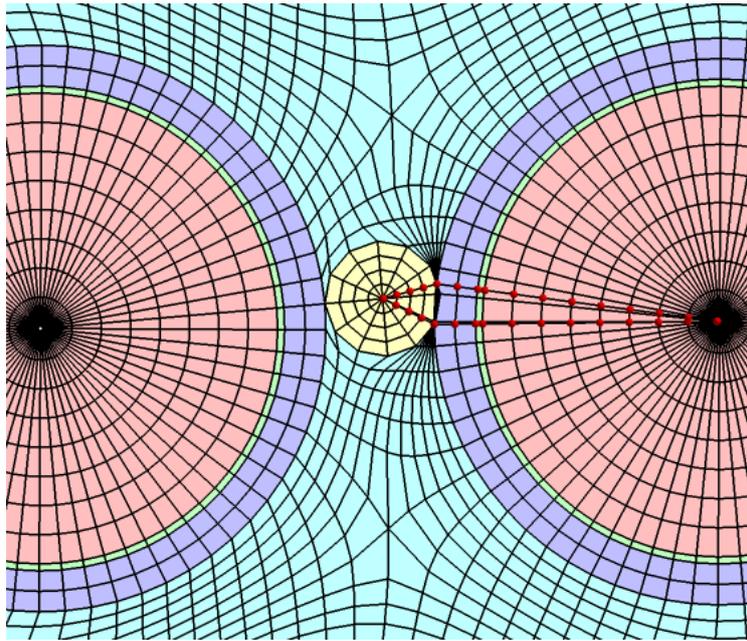
集合体非変形条件



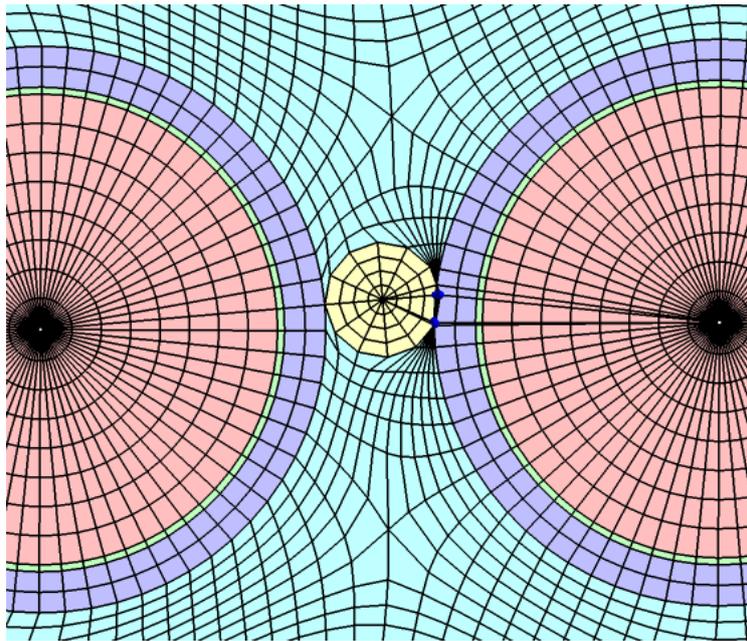
● 構造材接触点

集合体変形条件

図 3.2.1 構造材接触モデル（構造材接触点の検出）



● 追加される固体節点



● 削除される流体節点

図 3.2.2 構造材接触モデル（固体節点の追加・流体節点の削除）

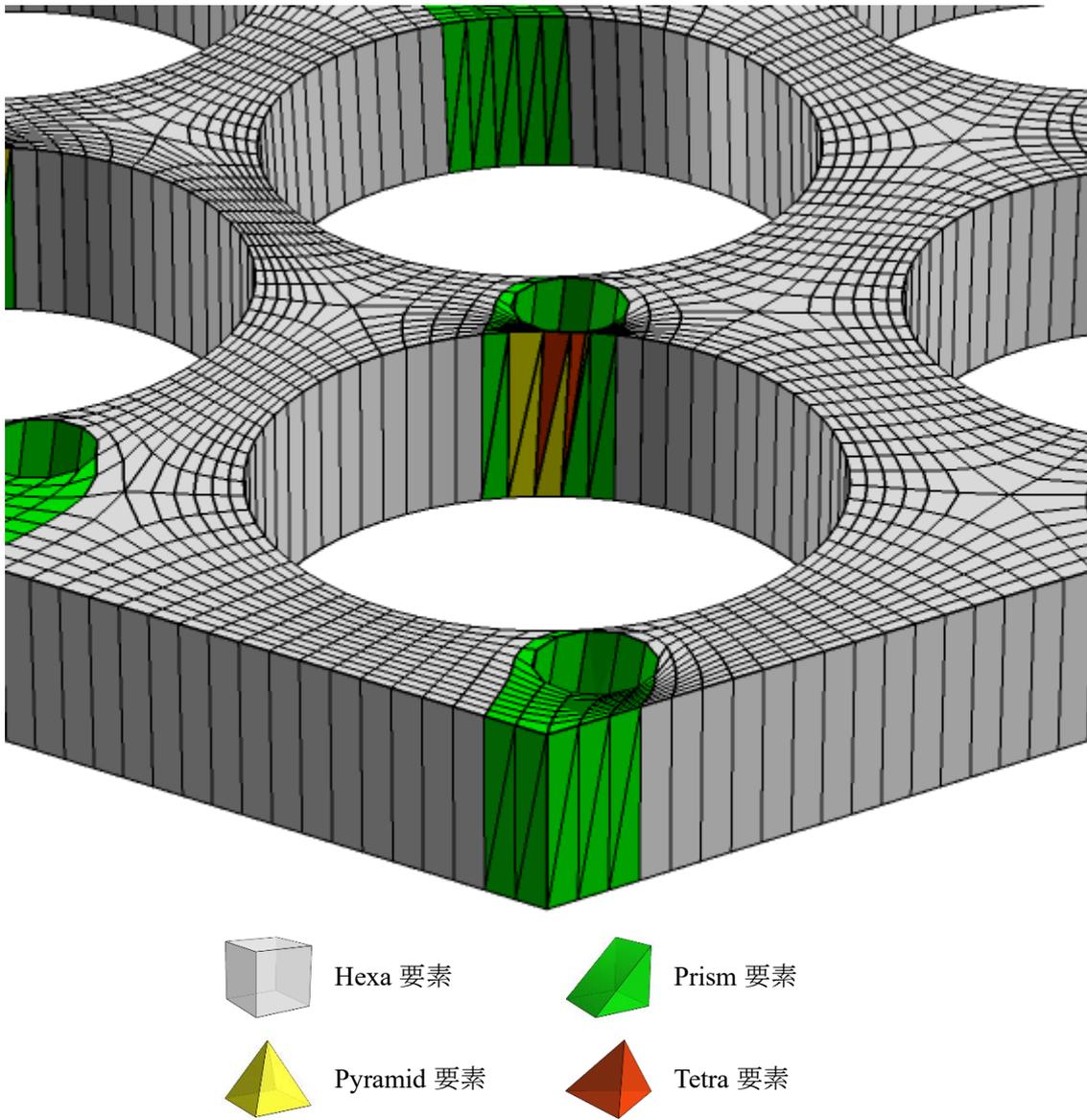
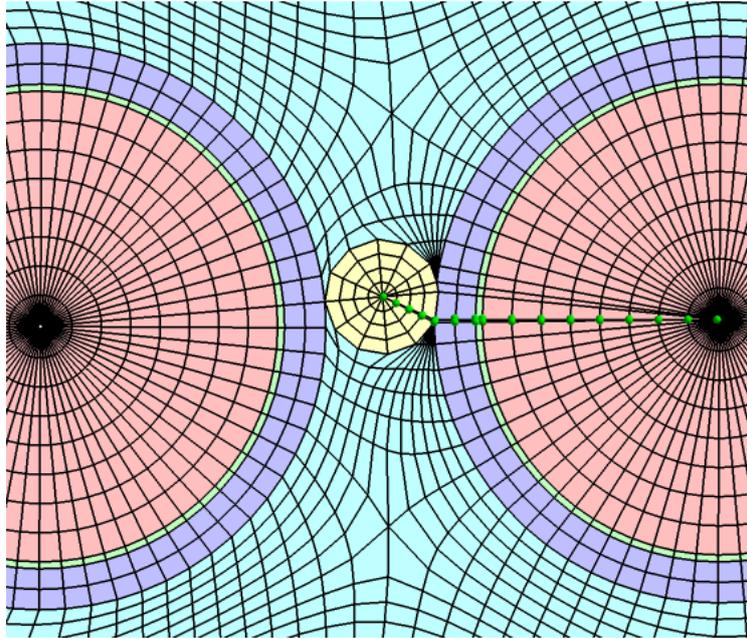
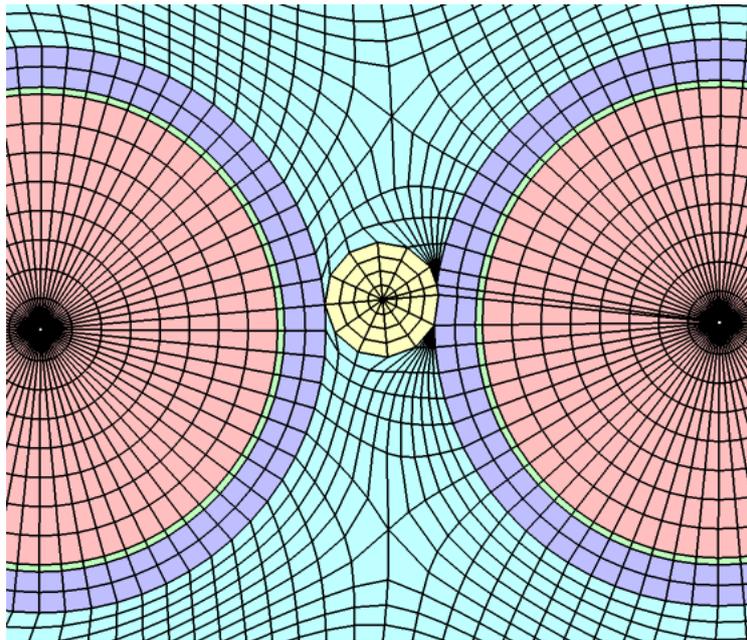


図 3.2.3 構造材接触モデル (要素種類の変更)



● 統合される固体節点



低品質要素の修正後

図 3.2.4 構造材接触モデル（低品質要素の修正）

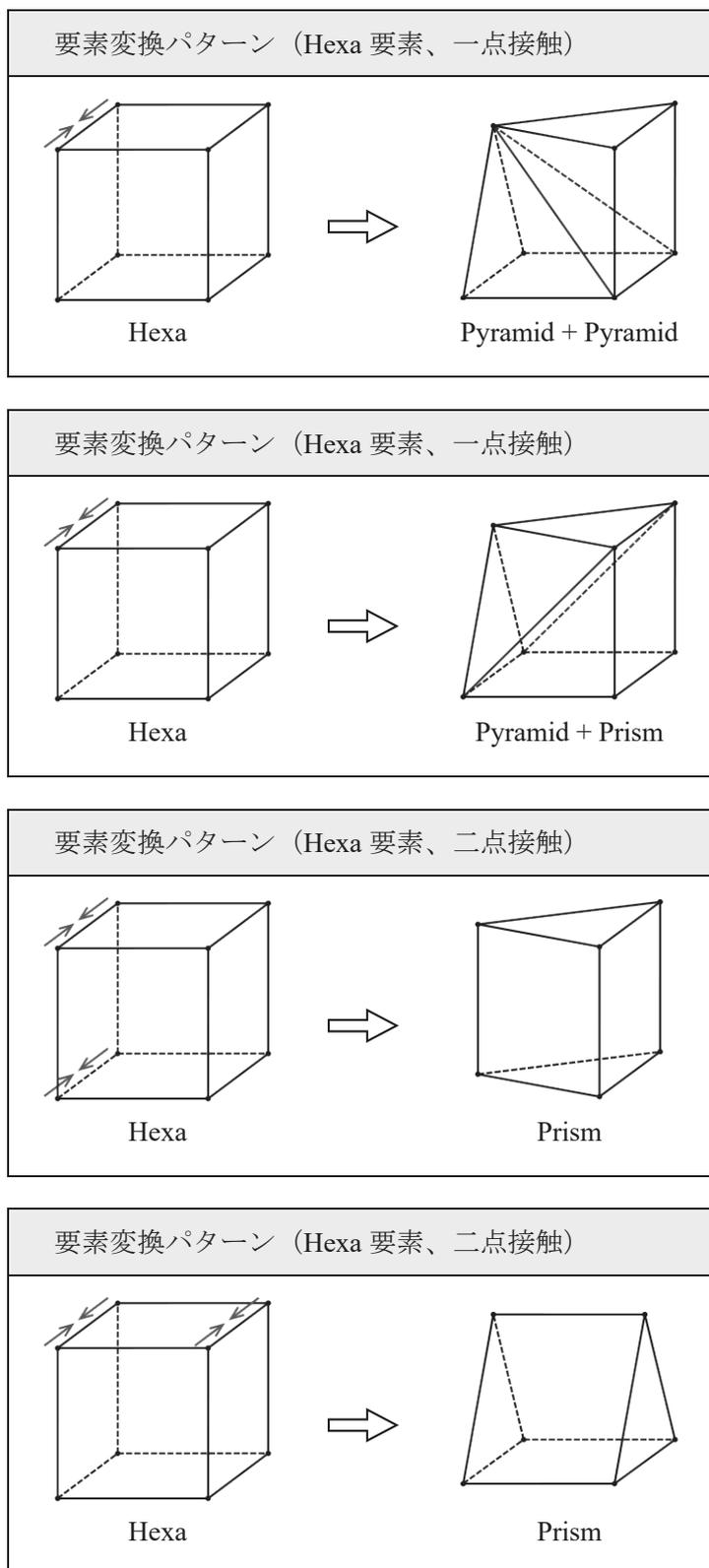


図 3.2.5 要素種類の変更 (1/3)

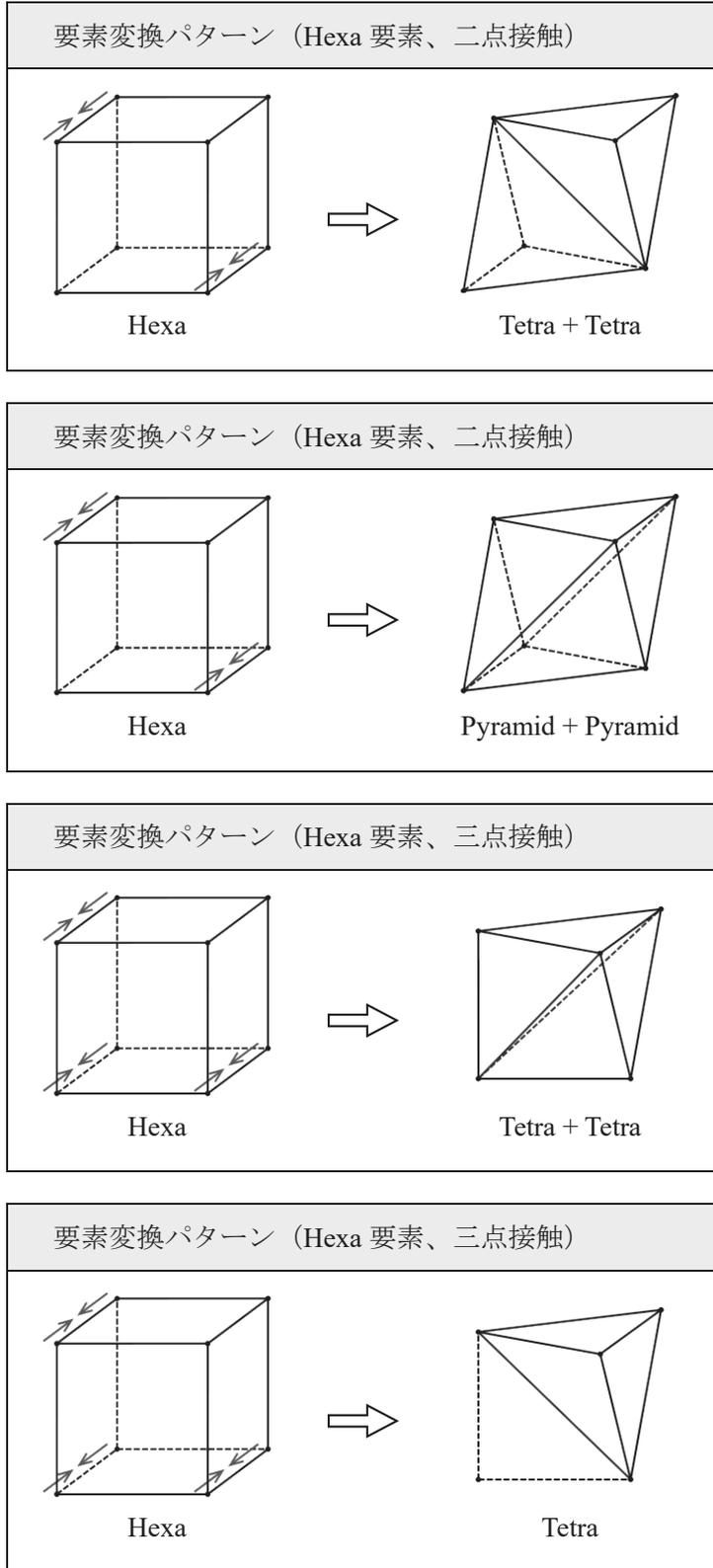


図 3.2.5 要素種類の変更 (2/3)

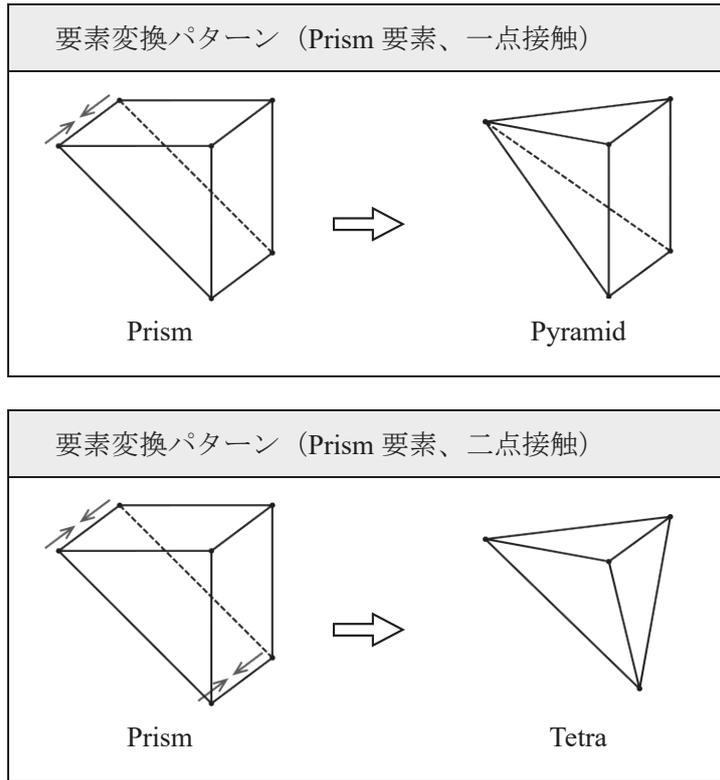


図 3.2.5 要素種類の変更 (3/3)

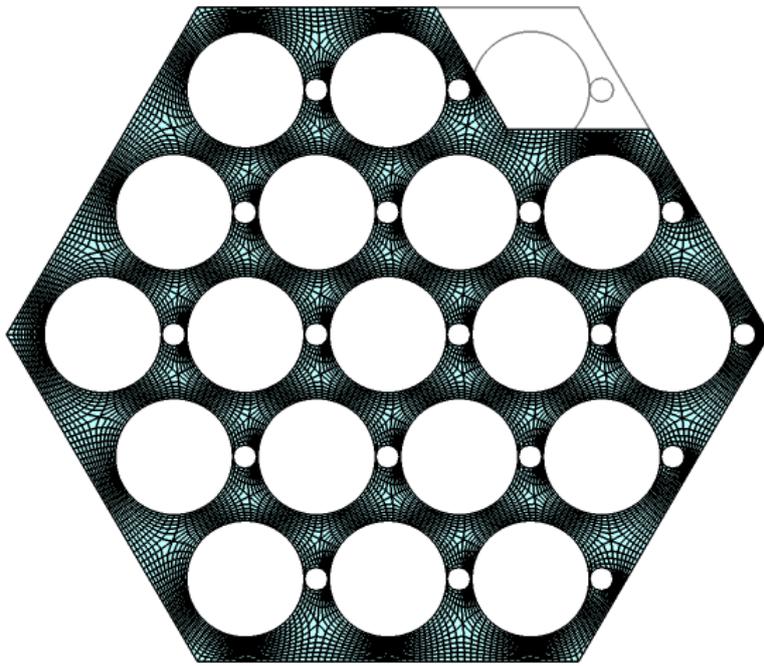


図 3.3.1 内部ダクト付き 18 本ピン集合体

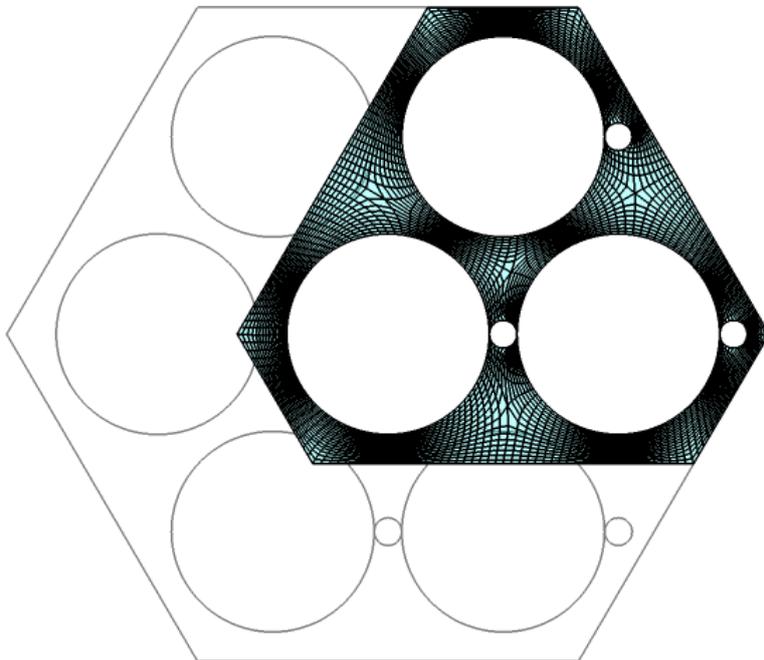


図 3.3.2 変形六角管型ラッパ管 3 本ピン集合体

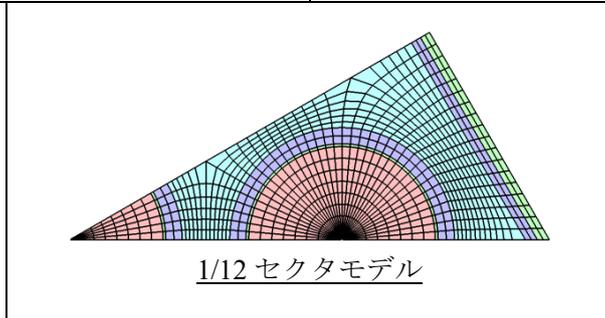
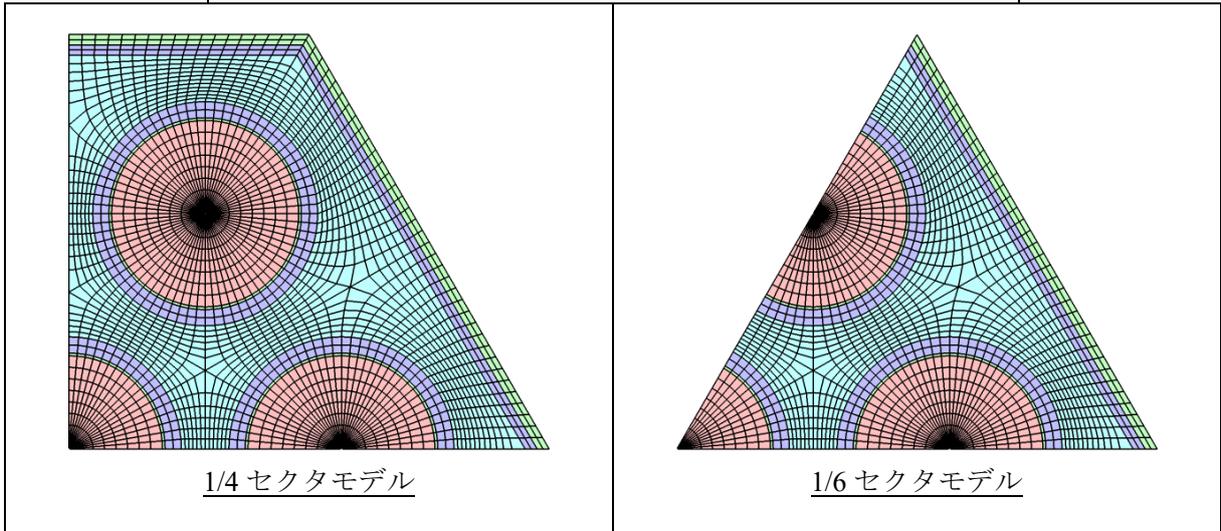
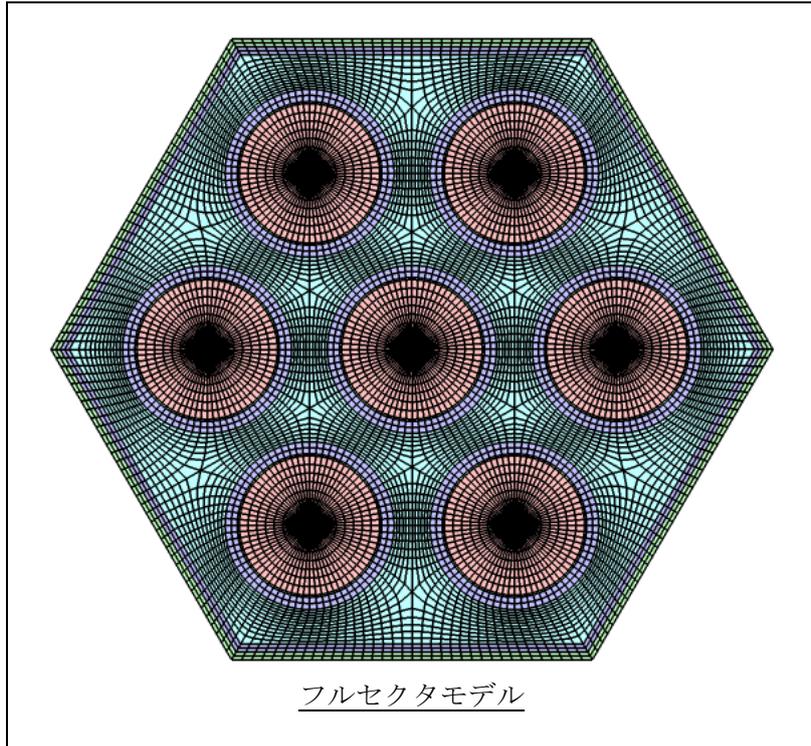
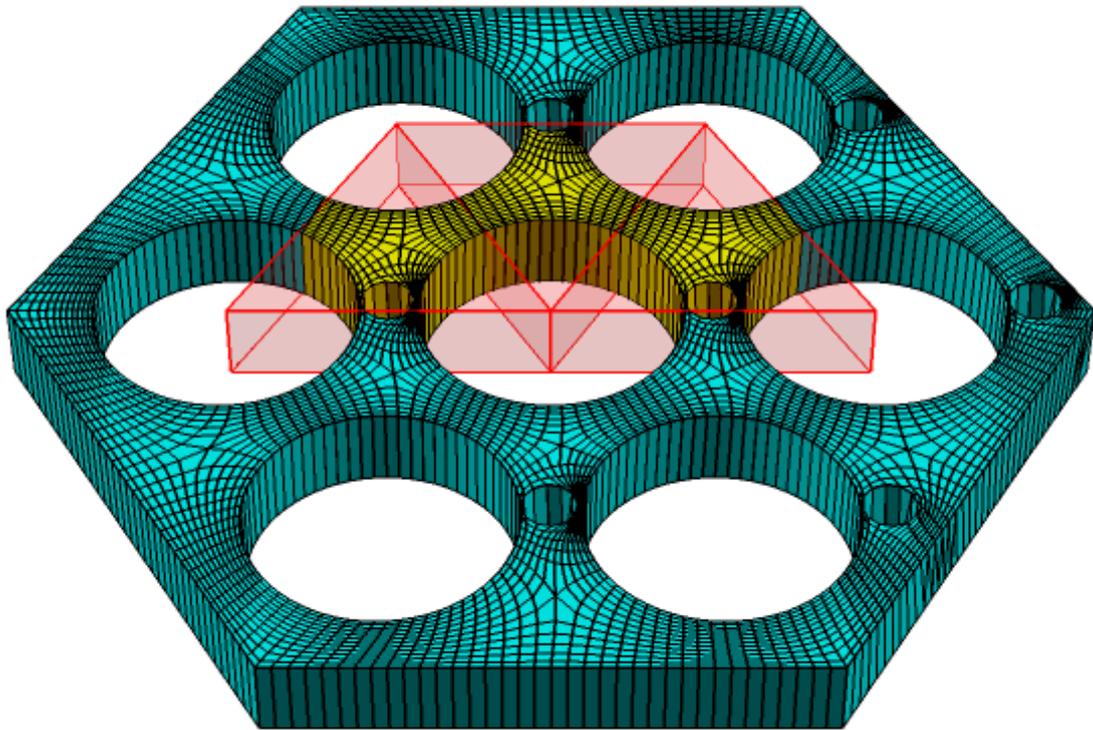


図 3.4 ベアバンドル集合体・ $1/n$ セクタモデル



流体要素



閉塞物要素



閉塞物として指定した領域

図 3.5 燃料集合体局所閉塞モデル

4. 結言

高速炉燃料集合体を対象とした有限要素法による詳細熱流動解析を精度良く実現するため、燃料集合体領域の要素分割を自動で実行する専用のメッシャーを開発した。本メッシャーの特徴として、複雑形状となる燃料集合体領域を構造的な要素で分割するため、マルチブロック法による領域分割を実行した上で、それぞれのブロック領域で曲線座標系による境界適合格子を生成する要素分割法を採用した。また、隣接するブロック領域間で要素の連続性を維持するため、六面体（Hexa）要素とプリズム状の（Prism）要素を併用する要素スキームを採用した。さらに、様々な条件における燃料集合体領域の要素分割にも対応するため、以下の随意的な要素分割モデルを開発した。

- (1) 構造材変形モデル
- (2) 構造材接触モデル
- (3) 内部ダクト付き燃料集合体モデル
- (4) $1/n$ セクタモデル
- (5) 局所閉塞モデル
- (6) 要素修正モデル

本メッシャーの開発によって、様々な条件における複雑な燃料集合体領域の要素分割を正確かつ効率的に実行することが可能となった。

参考文献

- 1) 大島宏之, 今井康友, 高速炉燃料集合体内詳細熱流動解析手法の開発(1), JNC TN9400 2001-064, 2001, 90p.
- 2) 大島宏之, 今井康友, 高速炉燃料集合体内詳細熱流動解析手法の開発(2)―乱流モデルの導入―, JNC TN9400 2003-045, 2003, 74p.
- 3) Aoto, K., Uto, N., Sakamoto, Y., Ito, T., Toda, M., Kotake, S., Design study and R&D progress on Japan sodium-cooled fast reactor, J. Nucl. Sci. Tech., vol.48, 2011, pp.463-471.
- 4) Thompson, J. F., Warsi, Z. U. A., Mastin, C. W., 数値格子生成の基礎と応用, 望月仁, 小国力, 河村哲也訳, 丸善出版, 1994, 352p.
- 5) Thompson, J. F., Thames, F. C., Mastin, C. W., Automatic numerical generation of body-fitted curvilinear coordinate system for field containing any number of arbitrary two-dimensional bodies, J. Comput. Phys., vol.15, 1974, pp.299-319.
- 6) Steger, J. L., Sorenson, R. L., Automatic mesh-point clustering near a boundary in grid generation with elliptic partial differential equations, J. Comput. Phys., vol.24, 1977, pp.274-302.
- 7) 山内二郎, 森口繁一, 一松信, 電子計算機のための数値計算法 2, 培風館, 1967, 239p.

付 録

燃料集合体領域計算格子生成プログラム

利用マニュアル

This is a blank page.

目 次

1. 実行操作法.....	51
2. 入出力ファイル.....	51
3. 各種要素分割モデルの利用法.....	51
3.1 構造材変形モデル.....	51
3.2 構造材接触モデル.....	52
3.3 内部ダクト付き燃料集合体モデル.....	52
3.4 1/n セクタモデル.....	52
3.5 局所閉塞モデル.....	52
3.6 要素修正モデル.....	53
4. プログラム構成.....	53

表リスト

表 A.1	実行メッセージ	54
表 A.2	入力ファイルリスト	58
表 A.3	出力ファイルリスト	59
表 A.4.1	燃料集合体データファイル「fuel_assembly.inp」	60
表 A.4.2	原点座標値データファイル「xyz_org.inp」	65
表 A.4.3	ワイヤ中空比データファイル「rhole.inp」	66
表 A.4.4 (1)	燃料ピン変形データファイル「deform_clad_000.inp」	67
表 A.4.4 (2)	ワイヤ変形データファイル「deform_wire_000.inp」	68
表 A.4.4 (3)	燃料ペレット変形データファイル「deform_plet_000.inp」	69
表 A.4.4 (4)	ラップ管変形データファイル「deform_wrap.inp」	70
表 A.4.5	除去燃料ピン番号データファイル「unused_pin.inp」	71
表 A.4.6	ラップ管頂点座標値データファイル「wrapper_tube.inp」	72
表 A.4.7	閉塞物領域データファイル「blockage.inp」	73
表 A.5.1	FIDAP 型メッシュデータファイル「fidap.FDNEUT」	74
表 A.5.2	SPIRAL 型メッシュデータファイル「restart」	75
表 A.5.3	周期境界データファイル「periodic_faces.inp」	76
表 A.5.4	双子節点データファイル「twin_nodes.inp」	77
表 A.5.5	壁境界データファイル「34_.sin」	78
表 A.5.6	燃料集合体形状データファイル「f.a.facet.stl」	79
表 A.5.7	ブロック領域接続データファイル「f.a.indexes.txt」	80
表 A.5.8	節点データファイル「f.a.nodes.txt」	81
表 A.5.9	壁境界データファイル「f.a.walls.txt」	82
表 A.6	プログラム構成	83
表 A.7	主要変数リスト	88

図リスト

図 A.1	燃料集合体仕様値の定義	94
図 A.2.1	流体領域メッシュ分割数の定義	95
図 A.2.2	構造材領域メッシュ分割数の定義	96

本付録では、本メッシュャーの利用マニュアルとして、その実行操作法と入出力ファイルのデータ形式を解説する。また、本メッシュャーのプログラム解説書として、プログラム構成と主要変数リストを付記する。

1. 実行操作法

表 A.1 に本メッシュャーの実行画面を示す。本メッシュャーの実行操作法としては、燃料集合体データファイル「fuel_assembly.inp」及び必要に応じて他の入力ファイルを用意して、ロードモジュールを実行する。実行後、入力ファイル「fuel_assembly.inp」、出力ファイル「fidap.FDNEUT」の名前を順に指定する必要があるが、Enter キーでスキップすると、デフォルトの名前が設定される。一連の処理が正常に終了した場合は、「Normal termination」のメッセージが出力される。

2. 入出力ファイル

表 A.2、表 A.3 に本メッシュャーの入出力ファイルの一覧を示す。入力ファイルとしては、「fuel_assembly.inp」が必須の入力ファイルであり、その他は必要に応じて用意する任意の入力ファイルである。出力ファイルとしては、「fidap.FDNEUT」がメッシュデータファイルであり、各種のデータ変換プログラムによって、SPIRAL 型データ、FLUENT 型データへの変換が可能である。同じく、「restart」もメッシュデータファイルであり、可視化ソフトを利用して、生成された要素の品質を確認できる。「periodic_faces.inp」「twin_nodes.inp」「34_.sin」は SPIRAL による解析を実行する際に必要となる。また、「f.a.facet.stl」は燃料集合体形状を CAD 上で再現する際に有用である。その他、「f.a.indexes.txt」「f.a.nodes.txt」「f.a.walls.txt」では燃料集合体の要素分割に際しての関連情報を確認できる。表 A.4.1～A.4.7、表 A.5.1～A.5.9 にそれぞれの入出力ファイルのデータ形式を示す。また、図 A.1、図 A.2.1 及び A.2.2 にそれらの関連図を示す。

3. 各種要素分割モデルの利用法

本節では、本文の第 3 章で解説した追加的な要素分割モデルの利用法を解説する。

3.1 構造材変形モデル

本文の 3.1 節に記述した構造材変形モデルを利用する場合は、表 A.4.1 に示す入力データファイル「fuel_assembly.inp」で以下の項目を YES に設定する。

Compute structure deformation ?	:	yes
---------------------------------	---	-----

構造材変形データは、表 A.4.4 に示す入力データファイル「deform_***.inp」で鉛直方向に対す

る離散点型データ ($z_i, \Delta x_i, \Delta y_i, \Delta D_i$) を入力する。任意の水平断面 $z = z_k$ での構造材変形データは、その離散点型データを 0 次補間または 1 次補間した値が設定される。すなわち、0 次補間では、離散点型データから z_i 値が $z = z_k$ に最も近いデータを検出して、検出されたデータをそのまま同断面の構造材変形データとする。対して、1 次補間では、 $z \leq z_k$ 側と $z \geq z_k$ 側でそれぞれ $z = z_k$ に最も近いデータを検出して、それらを $z = z_k$ に線形補間して得られたデータを同断面の構造材変形データとする。構造材変形データの補間次数は、入力データファイル「fuel_assembly.inp」で以下の項目から設定する。(NO で 0 次補間、YES で 1 次補間)

Apply linear interpolation for structure deformation ?	:	yes
--	---	-----

3.2 構造材接触モデル

本文の 3.2 節に記述した構造材接触モデルを利用する場合は、表 A.4.1 に示す入力データファイル「fuel_assembly.inp」で以下の項目を YES に設定する。

Compute structure contact ?	:	yes
-----------------------------	---	-----

また、低品質要素の修正に際して、その判定基準となる再分割比 χ は、同ファイルで以下の項目から設定する。

Threshold rate for repair sub-divided elements	:	1.000E-01
--	---	-----------

3.3 内部ダクト付き燃料集合体モデル

本文の 3.3 節に記述した内部ダクト付き燃料集合体モデルを利用する場合は、表 A.4.5 に示す入力データファイル「unused_pin.inp」で除去する燃料ピンの番号を、表 A.4.6 に示す入力データファイル「wrapper_tube.inp」でラップ管の頂点座標値を、それぞれ指定する。

3.4 1/n セクタモデル

本文の 3.4 節に記述した 1/n セクタモデルを利用する場合は、表 A.4.1 に示す入力データファイル「fuel_assembly.inp」で以下の項目からセクタ数 n を設定する。

Denominator of (1/n) sector model	:	1
-----------------------------------	---	---

3.5 局所閉塞モデル

本文の 3.5 節に記述した局所閉塞モデルを利用する場合は、表 A.4.7 に示す入力データファイル「blockage.inp」で任意の数の三角柱を指定して実行する。三角柱の指定は、任意の 3 点を指定することで実行され、指定した 3 点 $(x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3)$ に対して、 $(x_1, y_1, z_{\min}), (x_2, y_2, z_{\min}), (x_3, y_3, z_{\min})$ を下面、 $(x_1, y_1, z_{\max}), (x_2, y_2, z_{\max}), (x_3, y_3, z_{\max})$ を上面とする三角柱が生成される。なお、 $z_{\min} = \min(z_1, z_2, z_3)$ 、 $z_{\max} = \max(z_1, z_2, z_3)$ である。

3.6 要素修正モデル

本文の 3.6 節に記述した要素修正モデルを利用する場合は、表 A.4.1 に示す入力データファイル「fuel_assembly.inp」でエッジ長とエッジ交差角の片方または双方を設定する。

Minimum size of elements	: 1.000E-06
Threshold angle for repair low-quality elements	: 5.000E+00

4. プログラム構成

本メッシュャーのプログラムは、すべて Fortran90 によって記述されている。表 A.6 に本メッシュャーのプログラム構成を示す。同表において、インデントされたルーチンは、上位ルーチンから呼び出された下位ルーチンを意味する。また、表 A.7 に本メッシュャーの主要変数リストを示す。なお、同表に記載された変数は、すべて「mod_common_parameters.f90」で定義されている。

表 A.1 実行メッセージ (1/4)

メッセージ出力	メッセージ解説
<pre> Enter the input file name (default : fuel_assembly.inp) ... Enter the output file name (default : fidap.FDNEUT) ... Read geometry data ... 4 fuel pins have been removed ... No. 4 5 6 7 Center of all pins coordinates : 3.41858239E-05, 1.18510733E-04 Set nodal points ... 353800 nodal points have been generated . 190240 fluid nodal points have been generated . 163560 solid nodal points have been generated . Calculate nodal coordinates ... Now computing ... Axial No. 1, iteration out ... 63, 7.53E-06 Now computing ... Axial No. 2, iteration out ... 38, 8.01E-06 Now computing ... Axial No. 3, iteration out ... 38, 8.89E-06 ... Now computing ... Axial No. 145, iteration out ... 63, 7.53E-06 </pre>	<p>入力ファイルの名前の指定 ※デフォルトの場合は Enter キーでスキップ可 出力ファイルの名前の指定 ※デフォルトの場合は Enter キーでスキップ可</p> <p>データファイルの入力 ... 除去された燃料ピンの数と番号*1 ... 全燃料ピンの重心座標値</p> <p>節点の設定 ... 生成された節点の数 ... 生成された流体節点の数 ... 生成された構造材節点の数</p> <p>境界適合格子の生成 ... 水平断面番号、SOR 法の反復回数と最終残差</p>

*1 内部データ付き燃料集合体モデル、「unused_pin.inp」で指定した燃料ピンの番号

表 A.1 実行メッセージ (2/4)

メッセージ出力	メッセージ解説
<p>Modify nodal data ...</p> <p>Search structure contacts ...</p> <p>8 structure contacts have been detected .</p> <p>Make sub-divided elements ...</p> <p>192 nodal points have been added .</p> <p>128 nodal points have been deleted .</p> <p>Repair sub-divided elements ...</p> <p>26 nodal points have been unified .</p> <p>Minimum rate of sub-divided elements : 0.068</p> <p>Repair low-quality elements ...</p> <p>Minimum/Maximum size : 0.00E+00, 4.62E-03, Maximum angle : 269.93</p> <p>32 nodal points have been unified .</p> <p>Minimum/Maximum size : 1.00E-09, 4.62E-03, Maximum angle : 269.96</p> <p>16 nodal points have been unified .</p> <p>Minimum/Maximum size : 2.38E-09, 4.62E-03, Maximum angle : 266.81</p> <p>8 nodal points have been unified .</p> <p>Minimum/Maximum size : 1.35E-06, 4.62E-03, Maximum angle : 89.96</p> <p>No nodal points have been unified .</p> <p>Minimum/Maximum size : 1.35E-06, 5.66E-03, Maximum angle : 89.96</p>	<p>構造材接触モデルの処理 構造材接触点の検出</p> <p>... 検出された構造材接触点の数</p> <p>要素スキームの修正</p> <p>... 追加された構造材節点の数</p> <p>... 削除された流体節点の数</p> <p>再分割要素の修正</p> <p>... 統合された節点の数</p> <p>... 要素の再分割比の最小値</p> <p>要素修正モデルの処理</p> <p>... 最小・最大エッジ長、最大エッジ交差角</p> <p>... 修正された節点の数</p>

表 A.1 実行メッセージ (3/4)

メッセージ出力	メッセージ解説
<p>Set brick blocks ...</p> <p>Make brick blocks ... 3 blockages have been detected .</p> <p>Set split types ...</p> <p>Make general elements ... 372568 elements have been generated . 104 4-node elements have been generated . 488 5-node elements have been generated . 77352 6-node elements have been generated . 294624 8-node elements have been generated . (including 55 blockage elements)</p> <p>Make boundary surfaces ... 86048 surfaces have been generated . 17328 3-node surfaces have been generated . 68720 4-node surfaces have been generated . (including 152 blockage surfaces)</p>	<p>三次元要素の設定</p> <p>三次元要素の生成 ... 認識された閉塞物領域の数*1</p> <p>ワイヤ隣接要素の処理 (Hexa 要素→Prism 要素)</p> <p>領域データの設定 ... 生成された領域要素の数 ... 4 節点要素(Tetra 要素)の数 ... 5 節点要素(Pyramid 要素)の数 ... 6 節点要素(Prism 要素)の数 ... 8 節点要素(Hexa 要素)の数 ... (閉塞物領域要素の数)</p> <p>境界データの設定 ... 生成された境界要素の数 ... 3 節点要素(Triangle 要素)の数 ... 4 節点要素(Quad 要素)の数 ... (閉塞物境界要素の数)</p>

*1 局所閉塞モデル、「blockage.inp」で指定した閉塞物領域の数

表 A.1 実行メッセージ (4/4)

メッセージ出力	メッセージ解説
<pre> Search unused nodal points ... No nodal points have been deleted . Write FIDAP data ... 653975 wall boundaries data have been written . 154741 facets have been generated . 4632 inlet facets have been generated . 4632 outlet facets have been generated . 38016 fuel_pin_wall facets have been generated . 6924 spacer_wire_wall facets have been generated . 27653 wrapper_tube_wall facets have been generated . 38016 fuel_gap_outer facets have been generated . 6924 spacer_wire_center facets have been generated . 27660 exterior facets have been generated . 284 blockage_face facets have been generated . ***** Normal termination ***** </pre>	<p>未使用節点の削除 … 削除された節点の数</p> <p>データファイルの出力 … 生成された壁境界データの数*1 … 生成された STL 形式 Facet データの数*2 「inlet」境界 Facet データの数 「outlet」境界 Facet データの数 「fuel_pin_wall」境界 Facet データの数 「spacer_wire_wall」境界 Facet データの数 「wrapper_tube_wall」境界 Facet データの数 「fuel_gap_outer」境界 Facet データの数 「spacer_wire_center」境界 Facet データの数 「exterior」境界 Facet データの数 「blockage_face」境界 Facet データの数</p> <p>正常終了</p>
	<p>*1 壁境界データファイル「34_.sin」に出力される壁境界データの数</p>
	<p>*2 燃料集合体形状データファイル「f.a.facet.stl」に出力される Facet データの数</p>

表 A.2 入力ファイルリスト

ファイル名	ファイル解説	関連項目
fuel_assembly.inp	燃料集合体データファイル	表 A.4.1
xyz_org.inp	原点座標値データファイル	表 A.4.2
rhole.inp	ワイヤ中空比データファイル	表 A.4.3
deform_clad_000.inp deform_wire_000.inp deform_plet_000.inp deform_wrap.inp	構造材変形データファイル (構造材変形モデル)	表 A.4.4、3.1 節
unused_pin.inp	除去燃料ピン番号データファイル (内部ダクト付き燃料集合体モデル)	表 A.4.5、3.3 節
wrapper_tube.inp	ラッパ管頂点座標値データファイル (内部ダクト付き燃料集合体モデル)	表 A.4.6、3.3 節
blockage.inp	閉塞物領域データファイル (局所閉塞モデル)	表 A.4.7、3.5 節

表 A.3 出力ファイルリスト

ファイル名	ファイル解説	関連項目
fidap.FDNEUT	メッシュデータファイル (FIDAP 型)	表 A.5.1
restart	メッシュデータファイル (SPIRAL 型)	表 A.5.2
periodic_faces.inp	周期境界データファイル (SPIRAL の入力ファイル)	表 A.5.3
twin_nodes.inp	双子節点データファイル (SPIRAL の入力ファイル)	表 A.5.4、本文 2.2.5 項
34_.sin	壁境界データファイル (SPIRAL の入力ファイル)	表 A.5.5
f.a.facet.stl	燃料集合体形状データファイル (STL 形式)	表 A.5.6
f.a.indexes.txt	ブロック領域接続データファイル (確認用ファイル)	表 A.5.7
f.a.nodes.txt	節点データファイル (確認用ファイル)	表 A.5.8
f.a.walls.txt	壁境界データファイル (確認用ファイル)	表 A.5.9

表 A.4.1 燃料集合体データファイル「fuel_assembly.inp」(1/5)

データ項目	データ解説
# ----- Array/Order/Number of fuel pins ----- Array of fuel pins : 6 Order of fuel pins : 8 # Number of fuel pins : 169	(未使用) 燃料ピン次数*1 燃料ピン本数*1*2

*1 「燃料ピン次数」と「燃料ピン本数」は、どちらか片方を指定する。本例では、「燃料ピン本数」がコメント化されているため、「燃料ピン次数」が有効となる。双方が指定された場合、後から指定されたデータが有効となる。

*2 「燃料ピン本数」を指定する場合、有効な値は「1, 7, 19, 37, 61, 91, 127, 169, 217, 271, ...」であり、それ以外の値が指定された場合、指定値未満で最大となる有効な値が設定される。

データ項目	データ解説
# ----- Reel/Angle of spacer wire ----- Number of reels for spacer wire : 1 Angle of spacer wire at the bottom of fuel pins : 9.000E+01	ワイヤ巻き付け方向*1 集合体最下面のワイヤ位相*2 [degree]

*1 「ワイヤ巻き付け方向」は、+1 で時計回り、-1 で時計回り、0 で巻き付けなしの直立状態となる。

*2 「集合体最下面のワイヤ位相」は、+X 方向を 0°、+Y 方向を 90°とする。

表 A.4.1 燃料集合体データファイル「fuel_assembly.inp」(2/5)

データ項目	データ解説
# ----- Geometry sizes -----	
Pitch of spacer wire reel	ワイヤ巻き付けピッチ*1 [m]
Pitch of fuel pins	燃料ピン配列ピッチ*1 [m]
Outer diameter of cladding tube	被覆管外径*1 [m]
Inner diameter of cladding tube	被覆管内径*1 [m]
Diameter of fuel pellet	燃料ペレット外径*1 [m]
Diameter of hollow cone	燃料ペレット内径*1 [m]
Diameter of spacer wire	ワイヤ外径*1 [m]
# Width of gap between fuel pins and wrapper tube	最外層燃料ピンラップ管間ギャップ幅*1*2 [m]
Distance between two facing sides of wrapper tube	ラップ管対面間距離*1*2 [m]
Thickness of wrapper tube	ラップ管肉厚*1 [m]
Width of inter-wrapper gap	インターラップ領域ギャップ幅*1 [m]

*1 燃料集合体仕様値の定義は図 A.1 を参照。

*2 「最外層燃料ピンラップ管間ギャップ幅」と「ラップ管対面間距離」は、どちらか片方を指定する。本例では、「最外層燃料ピンラップ管間ギャップ幅」がコメント化されているため、「ラップ管対面間距離」が有効となる。双方が指定された場合、後から指定されたデータが有効となる。

表 A.4.1 燃料集合体データファイル「fuel_assembly.inp」(3/5)

データ項目	データ解説
# ----- Numbers of divisions -----	
Denominator of (1/n) sector model	1/n セクタモデル・セクタ数 (本文 3.4 節参照)
Order of elements	(未使用)
Number of axial meshes	集合体鉛直方向メッシュ分割数 ^{*1*} 2
Number of meshes for gaps between fuel pins	ギャップ線メッシュ分割数 ^{*1*} 3
# Number of circumferential meshes for fluid around fuel pins	燃料ピン周辺領域・周方向メッシュ分割数 ^{*1*} 3*4
Number of radial meshes for fluid around fuel pins	燃料ピン周辺領域・径方向メッシュ分割数 ^{*1*}
Number of radial meshes for fluid around wrapper tube	ラップ管周辺領域・径方向メッシュ分割数 ^{*1*}
Number of circumferential meshes for fluid around spacer wire	ワイヤ周辺領域・周方向メッシュ分割数 ^{*1*}
Number of radial meshes for fluid around spacer wire	ワイヤ周辺領域・径方向メッシュ分割数 ^{*1*} 5
Number of radial meshes for boundary layer	境界層径方向メッシュ分割数 ^{*1*}
Number of radial meshes for cladding tube	被覆管領域・径方向メッシュ分割数 ^{*1*}
Number of radial meshes for fuel gap	燃料ギャップ領域・径方向メッシュ分割数 ^{*1*}
Number of radial meshes for fuel pellet	燃料ペレット領域・径方向メッシュ分割数 ^{*1*}
Number of radial meshes for spacer wire	ワイヤ領域・径方向メッシュ分割数 ^{*1*}
Number of radial meshes for wrapper tube	ラップ管領域・径方向メッシュ分割数 ^{*1*}
Number of radial meshes for inter-wrapper gap	インターラップ領域・径方向メッシュ分割数 ^{*1*}
Number of nodal points on pin-wire contact point	燃料ピン-ワイヤ接触点の節点数 ^{*6*} (本文 2.2.5 項参照)

*1 各領域のメッシュ分割数の定義は図 A.2.1 を参照。

*2 集合体鉛直方向長は、次式で設定される。

*3 「集合体鉛直方向長」=「ワイヤ巻き付けピッチ」×「集合体鉛直方向メッシュ分割数」÷「燃料ピン周辺領域・周方向メッシュ分割数」
 「集合体鉛直方向長」は、「ワイヤ巻き付けピッチ」と「燃料ピン周辺領域・周方向メッシュ分割数」は、どちらか片方を指定する。本例では、「燃料ピン周辺領域・周方向メッシュ分割数」がコメント化されているため、「ギャップ線メッシュ分割数」が有効となる。双方が指定された場合、後から指定されたデータが有効となる。

*4 「燃料ピン周辺領域・周方向メッシュ分割数」を指定する場合、有効な値は $6n$ ($n=1, 2, 3, \dots$) であり、それ以外の値が指定された場合、指定値以下で最大となる有効な値が設定される。

*5 「ワイヤ周辺領域・径方向メッシュ分割数」は、「燃料ピン周辺領域・径方向メッシュ分割数」未満の値を設定する。

*6 「燃料ピン-ワイヤ接触点の節点数」は、SPIRAL による解析では 2 を、それ以外では 1 を設定する。

表 A.4.1 燃料集合体データファイル「fuel_assembly.inp」(4/5)

データ項目	データ解説
<pre> # ----- Computation parameters of nodal coordinates ----- Maximum counts of iteration for coordinates computation : 0 Convergence criteria of iteration for coordinates computation : 0.000E+00 Relaxation parameter of iteration for coordinates computation : 7.000E-01 Modification parameter for cladding tube : 1.000E+00 Modification parameter for wrapper tube : 1.100E+00 Size of near-wall elements : 5.000E-05 Minimum size of elements : 1.000E-06 Maximum size of elements : 0.000E+00 Threshold angle for repair low-quality elements : 5.000E+00 </pre>	<p>SOR 法最大反復計算回数*1 (本文 2.2 節参照) SOR 法収束判定値*2 (本文 2.2 節参照) SOR 法緩和係数 (本文 2.2 節参照) 燃料ピン表面のメッシュ緩和係数(本文 2.2.2 項参照) ラップ管表面のメッシュ緩和係数(本文 2.2.2 項参照) 境界層要素幅[m] (本文 2.2.4 項参照) エッジ長許容値[m] (本文 3.6 節参照) (未使用) エッジ交差角許容値[degree] (本文 3.6 節参照)</p>

*1 「SOR 法最大反復計算回数」は、0 を指定するとデフォルト値(水平断面上の節点数)が設定される。
*2 「SOR 法収束判定値」は、0 を指定するとデフォルト値(10⁻⁵)が設定される。

表 A.4.1 燃料集合体データファイル「fuel_assembly.inp」(5/5)

データ項目	データ解説
# ----- Structure deformation/contact data -----	
Create solid model ?	no
Compute structure deformation ?	yes
Apply linear interpolation for structure deformation ?	no
Compute structure contact ?	yes
Re-compute nodal coordinates for structure contact ?	no
Threshold rate for repair sub-divided elements	: 1.000E-01

*1 構造材モデルを ON に指定した場合、流体要素は生成されず、構造材要素のみ生成される。

データ項目	データ解説
# ----- I/O control parameters -----	
Write text data for FIDAP ?	yes
Write binary data for SPIRAL ?	yes
Write wall boundaries data for ASM/RSM ?	yes
Write facet data by STL-format ?	yes
Write debug informations ?	yes

FIDAP 型メッシュデータファイルの出力 ON/OFF
 SPIRAL 型メッシュデータファイルの出力 ON/OFF
 壁境界データファイルの出力 ON/OFF
 STL 形式データファイルの出力 ON/OFF
 確認用データファイルの出力 ON/OFF

表 A.4.2 原点座標値データファイル「xyz_org.inp」

データ形式
0 0 0.1535
データ解説
<p>入力データは「$(x_e) (y_e) (z_e)$」の形式で記述される一行のみである。データはすべて数値であり、座標系の原点とする座標値(x_e, y_e, z_e)を単位[m]で指定する。また、データは Fortran の標準形式(書式指定子を指定しない形式)で読み込まれる。旧座標系を(x, y, z)、新座標系を(x', y', z')とすると、$(x', y', z') = (x - x_e, y - y_e, z - z_e)$と変換される。なお、旧座標系は、集合体中心に位置する燃料ピン中心点座標値を(x_0, y_0)、集合体最下面の鉛直方向座標値をz_0として、(x_0, y_0, z_0)を原点とする座標系である。</p>

表 A.4.3 ワイヤ中空比データファイル「rhole.inp」

データ形式
0.005
データ解説
入力データは「 (γ_w) 」の形式で記述される一行のみである。データはすべて数値であり、本文 2.2.3 項に記述したワイヤ中心点に生成される中空領域の外径比 γ_w を指定する。ワイヤ外径を D_w とすると、中空領域の外径は $\gamma_w D_w$ に設定される。なお、デフォルト値は $\gamma_w=0.01$ である。

表 A.4.4 (1) 燃料ピン変形データファイル「deform_clad_000.inp」*1

データ形式			
zaxia	d_xclad	dyclad	d_dclad
-----	-----	-----	-----
0.000000E+00	0.000000E+00	0.000000E+00	0.000000E+00
4.166667E-03	-1.074065E-05	-1.414034E-06	1.083333E-05
8.333333E-03	-2.092839E-05	-5.607746E-06	2.166667E-05
1.250000E-02	-3.002608E-05	-1.243721E-05	3.250000E-05
1.666667E-02	-3.752777E-05	-2.166667E-05	4.333333E-05
2.083333E-02	-4.297331E-05	-3.297458E-05	5.416667E-05
2.500000E-02	-4.596194E-05	-4.596194E-05	6.500000E-05
...

データ解説
<p>入力データは「$(z_p)(\Delta x_p)(\Delta y_p)(\Delta D_p)$」の形式で記述され、その形式に一致しないデータ行は無効となる。データはすべて数値であり、燃料ピンの偏心量$(\Delta x_p, \Delta y_p)$、膨張量ΔD_pの鉛直方向分布を単位[m]で指定する。また、各データ行は Fortran の標準形式(書式指定子を指定しない形式)で読み込まれる。データの補間処理に関しては 3.1 節を参照。</p>

*1 ファイル名の「000」で変形を定義する燃料ピンの番号を指定する。

表 A.4.4 (2) ワイヤ変形データファイル「deform_wire_000.inp」*1

データ形式			
zaxia	d_xwire	d_ywire	d_dwire
-----	-----	-----	-----
0. 000000E+00	0. 000000E+00	0. 000000E+00	0. 000000E+00
4. 166667E-03	-1. 074065E-05	-1. 414034E-06	1. 083333E-05
8. 333333E-03	-2. 092839E-05	-5. 607746E-06	2. 166667E-05
1. 250000E-02	-3. 002608E-05	-1. 243721E-05	3. 250000E-05
1. 666667E-02	-3. 752777E-05	-2. 166667E-05	4. 333333E-05
2. 083333E-02	-4. 297331E-05	-3. 297458E-05	5. 416667E-05
2. 500000E-02	-4. 596194E-05	-4. 596194E-05	6. 500000E-05
...

データ解説
<p>入力データは「$(z_w) (\Delta x_w) (\Delta y_w) (\Delta D_w)$」の形式で記述され、その形式に一致しないデータ行は無効となる。データはすべて数値であり、ワイヤの偏心量$(\Delta x_w, \Delta y_w)$、膨張量ΔD_wの鉛直方向分布を単位[m]で指定する。また、各データ行は Fortran の標準形式(書式指定子を指定しない形式)で読み込まれる。データの補間処理に関しては 3.1 節を参照。</p>

*1 ファイル名の「000」で変形を定義する燃料ピンの番号を指定する。

表 A.4.4 (3) 燃料ペレット変形データファイル「deform_plet_000.inp」*1

データ形式			
zaxia	d_xplet	d_yplet	d_dplet
-----	-----	-----	-----
0. 000000E+00	0. 000000E+00	0. 000000E+00	0. 000000E+00
4. 166667E-03	-1. 074065E-05	-1. 414034E-06	1. 083333E-05
8. 333333E-03	-2. 092839E-05	-5. 607746E-06	2. 166667E-05
1. 250000E-02	-3. 002608E-05	-1. 243721E-05	3. 250000E-05
1. 666667E-02	-3. 752777E-05	-2. 166667E-05	4. 333333E-05
2. 083333E-02	-4. 297331E-05	-3. 297458E-05	5. 416667E-05
2. 500000E-02	-4. 596194E-05	-4. 596194E-05	6. 500000E-05
...

データ解説
<p>入力データは「$(z_q) (\Delta x_q) (\Delta y_q) (\Delta D_q)$」の形式で記述され、その形式に一致しないデータ行は無効となる。データはすべて数値であり、燃料ペレットの偏心量$(\Delta x_q, \Delta y_q)$、膨張量ΔD_qの鉛直方向分布を単位[m]で指定する。また、各データ行は Fortran の標準形式(書式指定子を指定しない形式)で読み込まれる。データの補間処理に関しては 3.1 節を参照。</p>

*1 ファイル名の「000」で変形を定義する燃料ピンの番号を指定する。

表 A.4.4 (4) ラップ管変形データファイル「deform_wrap.inp」

データ形式		
zaxia	d_dduct	d_wduct
-----	-----	-----
0. 000000E+00	0. 000000E+00	0. 000000E+00
4. 166667E-03	1. 743333E-04	5. 000000E-06
8. 333333E-03	3. 486667E-04	1. 000000E-05
1. 250000E-02	5. 230000E-04	1. 500000E-05
1. 666667E-02	6. 973333E-04	2. 000000E-05
2. 083333E-02	8. 716667E-04	2. 500000E-05
2. 500000E-02	1. 046000E-03	3. 000000E-05
...
データ解説		
<p>入力データは「(za) (ΔD_d) (ΔW_d)」の形式で記述され、その形式に一致しないデータ行は無効となる。データはすべて数値であり、ラップ管内対面間距離の変位量 ΔD_d、ラップ管肉厚の変位量 ΔW_d の鉛直方向分布を単位[m]で指定する。また、各データ行は Fortran の標準形式(書式指定子を指定しない形式)で読み込まれる。データの補間処理に関しては 3.1 節を参照。</p>		

表 A.4.5 除去燃料ピン番号データファイル「unused_pin.inp」

データ形式
4 5 6 7
データ解説
入力データは「(i_p)」の形式で記述され、その形式に一致しないデータ行は無効となる。データはすべて整数であり、内部ダクト付き燃料集合体モデルにおいて、除去する燃料ピンの番号 i_p を指定する。

表 A.4.6 ラップ管頂点座標値データファイル「wrapper_tube.inp」

データ形式	
(xd)	(yd)
0.021040838	0.0
0.014455419	0.01140628
0.006585419	0.01140628
0.002650419	0.0182219
-0.010520419	0.0182219
-0.021040838	0.0
-0.010520419	-0.0182219
0.010520419	-0.0182219

データ解説
<p>入力データは「(x_d) (y_d)」の形式で記述され、その形式に一致しないデータ行は無効となる。データはすべて数値であり、ラップ管の頂点座標値(x_d, y_d)を単位[m]で指定する。また、各データ行は Fortran の標準形式(書式指定子を指定しない形式)で読み込まれる。指定された頂点は、原点からの角度 θ ($\cos\theta = x/r$, $\sin\theta = y/r$, $r^2 = x^2 + y^2$)でソートされた上で、ラップ管の頂点として設定される。なお、「xyz_org.inp」で座標系の原点を指定した場合は、新座標系でラップ管の頂点座標値を指定する。</p>

表 A.4.7 閉塞物領域データファイル「blockage.inp」

データ形式								
(x1)	(y1)	(z1)	(x2)	(y2)	(z2)	(x3)	(y3)	(z3)
0.0	0.0	0.0	0.00787	0.0	0.0	0.003935	0.006816	0.004264
0.0	0.0	0.0	-0.00787	0.0	0.0	-0.003935	0.006816	0.004264
0.0	0.0	0.0	0.003935	0.006816	0.004264	-0.003935	0.006816	0.004264
データ解説								
<p>入力データは「(x1) (y1) (z1) (x2) (y2) (z2) (x3) (y3) (z3)」の形式で記述され、その形式に一致しないデータ行は無効となる。データはすべて数値であり、閉塞物領域とする三角柱の頂点を単位[m]で指定する。また、各データ行は Fortran の標準形式(書式指定子を指定しない形式)で読み込まれる。指定した3点(x1, y1, z1)(x2, y2, z2)(x3, y3, z3)に対して、(x1, y1, z_{min})(x2, y2, z_{min})(x3, y3, z_{min})を下面、(x1, y1, z_{max})(x2, y2, z_{max})(x3, y3, z_{max})を上面とする三角柱が生成される。なお、z_{min} = min(z1, z2, z3)、z_{max} = max(z1, z2, z3)である。流体領域の三次元要素に関して、その要素中心点が三角柱の内部に存在する場合、当該要素の領域データが閉塞物領域へと変更される。</p>								

表 A.5.1 FIDAP 型メッシュデータファイル「fidap.FDNEUT」

データ形式							
** FIDAP NEUTRAL FILE							
ICEM CFD Model Configuration = fidap							
VERSION 7.00							
<15.05.25> <09:34:14>							
	864	1248	7	3 3			
	0	0	0	0			
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0							
	0	0	0				
NODAL COORDINATES							
1	3.3656322641E-03	1.9431475134E-03	0.0000000000E+00				
2	-1.6802280316E-10	3.8862920914E-03	0.0000000000E+00				
3	-3.3656277609E-03	1.9431458945E-03	0.0000000000E+00				
...				
データ解説							
<p>燃料集合体のメッシュデータとして、節点データ、要素データ等が記述される。各種のデータ変換プログラムによって、SPIRAL 型データ、FLUENT 型データへの変換が可能である。なお、本ファイルを出力させるには、入力ファイル「fuel_assembly.inp」で以下の項目を YES に設定する。</p>							
<table border="1" style="width: 100%;"> <tr> <td style="width: 70%;">Write text data for FIDAP ?</td> <td style="width: 5%;">:</td> <td style="width: 25%;">yes</td> </tr> </table>					Write text data for FIDAP ?	:	yes
Write text data for FIDAP ?	:	yes					

表 A.5.2 SPIRAL 型メッシュデータファイル「restart」

データ形式		
(バイナリ形式)		
データ解説		
<p>燃料集合体のメッシュデータとして、節点データ、要素データ等が記述される。SPIRAL のポスト処理プログラムによって、FIELDVIEW、PARAVIEW 等の可視化ソフト型データへの変換が可能である。各節点において以下のデータが出力されるため、生成された要素の品質を確認する際に有用である。</p> <p>computation-residual : SOR 法の残差、本文 2.2 節の Δr_i</p> <p>min-length : 水平断面上における最小エッジ長</p> <p>max-length : 水平断面上における最大エッジ長</p> <p>aspect-rate : 水平断面上におけるアスペクト比 (max-length/min-length)</p> <p>edge-angle : 水平断面上におけるエッジ交差角、本文 3.6 節の $\theta_i - 90^\circ$</p> <p>なお、本ファイルを出力させるには、入力ファイル「fuel_assembly.inp」で以下の項目を YES に設定する。</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;">Write binary data for SPIRAL ?</td> <td style="text-align: right; padding: 5px;">: yes</td> </tr> </table>	Write binary data for SPIRAL ?	: yes
Write binary data for SPIRAL ?	: yes	

表 A.5.3 周期境界データファイル「periodic_faces.inp」

データ形式						
49	50	98	97			
50	51	99	98			
51	52	100	99			
52	53	101	100			
53	54	102	101			
...			
データ解説						
<p>SPIRAL の入力データであり、集合体最下面の境界要素データが記述される。データ形式は「$(i_{n1}) (i_{n2}) (i_{n3}) (i_{n4})$」であり、境界要素を構成する節点の節点番号が記述される。SPIRAL による解析において、周期境界条件を利用する際に必要となる。なお、本ファイルを出力させるには、入力ファイル「fuel_assembly.inp」で以下の項目を YES に設定する。</p>						
<table border="1" style="width: 100%;"> <tr> <td style="width: 70%;">Write text data for FIDAP ?</td> <td style="width: 5%; text-align: center;">:</td> <td style="width: 25%; text-align: center;">yes</td> </tr> </table>				Write text data for FIDAP ?	:	yes
Write text data for FIDAP ?	:	yes				

表 A.5.4 双子節点データファイル「twin_nodes.inp」

データ形式				
355	423			
855	923			
1355	1423			
1855	1923			
2355	2423			
...	...			
データ解説				
<p>SPIRAL の入力データであり、燃料ピン-ワイヤ接触点に配置される双子節点データが記述される。データ形式は「$(i_{n1}) (i_{n2})$」であり、双子節点の節点番号が記述される。SPIRAL による解析において、温度分布解析を実行する際に必要となる。なお、本ファイルを出力させるには、入力ファイル「fuel_assembly.inp」で以下の項目を 2 に設定する。</p>				
<table border="1" style="width: 100%;"> <tr> <td style="width: 80%;">Number of nodal points on pin-wire contact point</td> <td style="width: 5%; text-align: center;">:</td> <td style="width: 15%; text-align: center;">2</td> </tr> </table>		Number of nodal points on pin-wire contact point	:	2
Number of nodal points on pin-wire contact point	:	2		

表 A.5.5 壁境界データファイル「34_.sin」

データ形式						
1	0	0				
5.47188538E-07	9.98206673E-01	5.98618302E-02	0.00000000E+00			
1	0	0				
3.11707191E-05	-9.92675706E-01	1.20809528E-01	0.00000000E+00			
1	0	0				
3.99095060E-03	-1.39377702E-01	-9.90239292E-01	0.00000000E+00			
2	0	0				
-9.42101728E-09	9.90762986E-01	1.35604958E-01	0.00000000E+00			
2	0	0				
9.96819654E-05	-9.30592195E-01	3.66057599E-01	0.00000000E+00			
...			
...			
データ解説						
<p>SPIRAL の入力データであり、燃料集合体の各流体要素から垂直に直射する壁境界データが記述される。データ形式は「$(i_e)(n_s)(n_f)$」「$(r_n)(n_x)(n_y)(n_z)$」であり、要素番号 i_e、壁法線距離 r_n、壁法線ベクトル (n_x, n_y, n_z) が記述される。また、n_s と n_f にはゼロが設定される。SPIRAL による解析において、高次の乱流モデルを利用する際に必要となる。なお、本ファイルを出力させるには、入力ファイル「fuel_assembly.inp」で以下の項目を YES に設定する。</p>						
<table border="1" style="width: 100%;"> <tr> <td style="width: 70%;">Write wall boundaries data for ASM/RSM ?</td> <td style="width: 5%;">:</td> <td style="width: 25%;">yes</td> </tr> </table>				Write wall boundaries data for ASM/RSM ?	:	yes
Write wall boundaries data for ASM/RSM ?	:	yes				

表 A.5.6 燃料集合体形状データファイル「f.a.facet.stl」

データ形式			
<pre> solid inlet facet normal 0.00000000E+00 0.00000000E+00 1.00000000E+00 outer loop vertex 3.34135003E-03 5.08733976E-04 0.00000000E+00 vertex 3.33796807E-03 3.30262789E-04 0.00000000E+00 vertex 3.25000000E-03 8.08266887E-20 0.00000000E+00 end loop end facet ... </pre>			
データ解説			
<p>燃料集合体の形状データとして、図 2.9.1、2.9.2 に示す燃料集合体の境界データが STL 形式で記述される。燃料集合体の形状を CAD 上で再現する際に有用である。なお、本ファイルを出力させるには、入力ファイル「fuel_assembly.inp」で以下の項目を YES に設定する。</p>			
<table border="1" style="width: 100%;"> <tr> <td style="width: 70%;">Write facet data by STL-format ?</td> <td style="width: 5%; text-align: center;">:</td> <td style="width: 25%; text-align: center;">yes</td> </tr> </table>	Write facet data by STL-format ?	:	yes
Write facet data by STL-format ?	:	yes	

表 A.5.7 ブロック領域接続データファイル「f.a.indexes.txt」

データ形式			
(省略)			
データ解説			
<p>本メッシュの確認用データとして、燃料ピン・サブチャンネル中心点・ギャップ線の接続データが記述される。本ファイルを出力させるには、入力ファイル「fuel_assembly.inp」で以下の項目を YES に設定する。</p>			
<table border="1"><tr><td>Write debug informations ?</td><td>:</td><td>yes</td></tr></table>	Write debug informations ?	:	yes
Write debug informations ?	:	yes	

表 A.5.8 節点データファイル「f.a.nodes.txt」

データ形式			
(省略)			
データ解説			
<p>本メッシュの確認用データとして、集合体最下面の各ブロック領域における節点データが記述される。本ファイルを出力させるには、入力ファイル「fuel_assembly.inp」で以下の項目をYESに設定する。</p> <table border="1" data-bbox="236 613 1337 658"> <tr> <td>Write debug informations ?</td> <td>:</td> <td>yes</td> </tr> </table>	Write debug informations ?	:	yes
Write debug informations ?	:	yes	

表 A.5.9 壁境界データファイル「f.a.walls.txt」

データ形式						
(省略)						
データ解説						
<p>本メッシュの確認用データとして、集合体最下層の各流体要素から垂直に直射する壁境界データが記述される。本ファイルを出力させるには、入力ファイル「fuel_assembly.inp」で以下の項目を YES に設定する。</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">Write wall boundaries data for ASM/RSM ?</td> <td style="text-align: right; padding: 2px;">:</td> <td style="text-align: right; padding: 2px;">yes</td> </tr> <tr> <td style="padding: 2px;">Write debug informations ?</td> <td style="text-align: right; padding: 2px;">:</td> <td style="text-align: right; padding: 2px;">yes</td> </tr> </table>	Write wall boundaries data for ASM/RSM ?	:	yes	Write debug informations ?	:	yes
Write wall boundaries data for ASM/RSM ?	:	yes				
Write debug informations ?	:	yes				

表 A.6 プログラム構成 (1/5)

プログラム構成	プログラム解説
<p>FuelAssemblyPreProcessor</p> <p> OpenFiles</p> <p> ReadGeometryData</p> <p> ReadFuelAssembly</p> <p> SetDefaultControlParameters</p> <p> ReadControlParameters</p> <p> AllocateGeometryArray</p> <p> SetDefaultGeometrySizes</p> <p> ReadGeometrySizes</p> <p> SetDependentControlParameters</p> <p> SeparateInputString</p> <p> MakeIndexes</p> <p> CalculateGeometryData</p> <p> DeleteUnusedPins</p> <p> CalculatePinWireCoordinates</p> <p> CalculateWrapperCoordinates</p> <p> WriteFuelAssembly</p> <p> ReadDeformationData</p> <p> WriteDeformationData</p> <p> SetNodalPoints</p> <p> SetFluidNodalPoints</p> <p> AssignFluidNodalPoints</p>	<p>メインプログラム</p> <p> 入力ファイルの初期化</p> <p> データファイルの入力</p> <p> 燃料集合体データファイル「fuel_assembly.inp」の入力</p> <p> 要素分割数等のデフォルト値の設定</p> <p> 要素分割数等の入力</p> <p> 形状データの配列の初期化</p> <p> 燃料集合体仕様値のデフォルト値の設定</p> <p> 燃料集合体仕様値の入力</p> <p> 燃料集合体データの設定</p> <p> データ行の処理</p> <p> ブロック領域接続データの設定</p> <p> 形状データの設定</p> <p> 除去燃料ピンの設定</p> <p> 燃料ピン中心点座標値の計算</p> <p> ラップ管頂点座標値の計算</p> <p> 燃料集合体データファイル「fuel_assembly.inp」の出力</p> <p> 構造材変形データファイル「deform_***.inp」の入力</p> <p> 構造材変形データファイル「deform_***.inp」の出力</p> <p> 節点データの設定</p> <p> 流体領域における節点データの設定</p> <p> 流体領域における節点データの生成</p>

表 A.6 プログラム構成 (2/5)

プログラム構成	プログラム解説
<p>LinkFluidNodalPoints SetSolidNodalPoints AssignSolidNodalPoints LinkSolidNodalPoints SetBoundaryFlags CalculateNodalCoordinates CalculateWallNodalCoordinates CalculateFluidNodalCoordinates OptimizeFluidNodalCoordinates OptimizeLocalNodalCoordinates OptimizeLayerNodalCoordinates OptimizeWireNodalCoordinates OptimizeSolidNodalCoordinates OptimizePinNodalCoordinates OptimizeWrapperNodalCoordinates CalculateSolidNodalCoordinates ModifyNodalData ModifyStructureContact SearchContactPoints SearchStructureContactPoints SearchLocalContactPoints AddNodalPoints</p>	<p>流体領域における節点データのブロック領域間での接続 構造材領域における節点データの設定 構造材領域における節点データの生成 構造材領域における節点データのブロック領域間での接続 節点データに対する境界情報の設定 境界適合格子の生成 構造材表面における節点座標値の計算 流体領域における節点座標値の計算 流体領域における節点座標値の最適化計算 局所領域における節点座標値の最適化計算 境界層領域における節点座標値の最適化計算 ワイヤ表面における節点座標値の最適化計算 構造材領域における節点座標値の最適化計算 燃料ピン表面における節点座標値の最適化計算 ラップ管表面における節点座標値の最適化計算 構造材領域における節点座標値の計算 各種の要素分割モデルの処理 構造材接触モデルの処理 構造材接触点の検出 構造材接触点の検出 局所領域における構造材接触点の検出 要素スキームの修正</p>

表 A.6 プログラム構成 (3/5)

プログラム構成	プログラム解説
<p>AddStructureNodalPoints AddLocalNodalPoints DeleteStructureNodalPoints ReallocateNodalCoordinates RenumberNodalData UnifyStructureNodalPoints RenumberNodalData CalculateSolidNodalCoordinates ModifySolidModel RenumberNodalData ModifySectorModel RenumberNodalData RepairLowQualityElements EvaluateElementQualities EvaluateFluidElementQualities EvaluateSolidElementQualities UnifyLowQualityElements RenumberNodalData SetBrickBlocks SetFluidBrickBlocks AssignFluidBrickBlocks SetSolidBrickBlocks</p>	<p>構造材接触位置での固体節点の追加 局所領域における構造材接触位置での固体節点の追加 構造材接触位置での流体節点の削除 節点データの配列の再初期化 節点データの再設定 再分割要素の修正 節点データの再設定 構造材領域における節点座標値の計算 構造材モデルの処理 (全流体節点の削除) 節点データの再設定 1/n セクタモデルの処理 節点データの再設定 要素修正モデルの処理 要素品質の評価 流体領域における要素品質の評価 構造材領域における要素品質の評価 低品質要素の修正 節点データの再設定 格子データの設定 流体領域における格子データの設定 流体領域における格子データの生成 構造材領域における格子データの設定</p>

表 A.6 プログラム構成 (4/5)

プログラム構成	プログラム解説
<p>AssignSolidBrickBlocks MakeBrickBlocks MakeFluidBrickBlocks MakeLocalBrickBlocks MakeSolidBrickBlocks MakeLocalBrickBlocks MakeBlockageBlocks MakeLocalBrickBlocks SetSplitTypes AssignSplitTypes ArrangeSplitTypes LinkFluidSplitTypes LinkSolidSplitTypes MakeGeneralElements CountGeneralElements GenerateGeneralElements GenerateSubElements SearchAddedNodalPoints GenerateHexaElement GeneratePrismElement GeneratePrismElement GeneratePyramidElement</p>	<p>格子データにおける格子データの生成 格子データの生成 流体領域における格子データの計算 局所領域における格子データの計算 構造材領域における格子データの計算 局所領域における格子データの計算 閉塞物領域における格子データの計算 局所領域における格子データの計算 格子分割タイプの設定 格子分割タイプ (燃料ピン周辺領域) の設定 格子分割タイプ (隣接領域との接続) の設定 流体領域における格子分割タイプのブロック領域間での接続 構造材領域における格子分割タイプのブロック領域間での接続 要素データの生成 要素データの配列の初期化 要素データの生成 再分割要素の生成 追加節点の検出 Hexa 要素の生成 Prism 要素の生成 Prism 要素の生成 Pyramid 要素の生成</p>

表 A.6 プログラム構成 (5/5)

プログラム構成	プログラム解説
<p>GenerateTetraElement</p> <p>MakeBoundarySurfaces</p> <p>CountBoundarySurfaces</p> <p>GenerateBoundarySurfaces</p> <p> MergeBoundarySurfaces</p> <p>MakeBlockageSurfaces</p> <p>SearchUnusedNodalPoints</p> <p>RenumberNodalData</p> <p>WriteData</p> <p>WriteFIDAP</p> <p> WriteControlParameters</p> <p> WriteNodalCoordinates</p> <p> WriteBoundaryConditions</p> <p> WriteElementSurfaceDefinitions</p> <p>WriteSPIRAL</p> <p>WriteWallBoundaries</p> <p>WriteFacetData</p> <p>WriteDebugData</p> <p> WriteDebugIndexData</p> <p> WriteDebugNodalData</p> <p>End</p>	<p>Tetra 要素の生成</p> <p>境界データの生成</p> <p>境界データの配列の初期化</p> <p>境界データの生成</p> <p> 重複する境界データの削除</p> <p>閉塞物領域における境界データの生成</p> <p>未使用節点の削除</p> <p>節点データの再設定</p> <p>データファイルの出力</p> <p>メッシュデータファイル「fidap.FDNEUT」、双子節点データファイル「twin_nodes.inp」の出力</p> <p>制御データの出力</p> <p>節点データの出力</p> <p>境界データの出力</p> <p>要素データの出力、周期境界データファイル「periodic_faces.inp」の出力</p> <p>メッシュデータファイル「restart」の出力</p> <p>壁境界データファイル「34_sin」「f.a.walls.txt」の出力</p> <p>燃料集合体形状データファイル「f.a.facet.stl」の出力</p> <p>確認用データファイルの出力</p> <p> ブロック領域接続データファイル「f.a.indexes.txt」の出力</p> <p> 節点データファイル「f.a.nodes.txt」の出力</p> <p>処理終了</p>

表 A.7 主要変数リスト (1/6)

変数	型	解説
maxarray	integer	燃料ピン配列型 (=6)
maxorder	integer	燃料ピン層数
maxfpin	integer	燃料ピン本数
maxduct	integer	ラップ管頂点数
maxsubc	integer	サブチャンネル中心点の数
maxsubc_outer	integer	外側サブチャンネル中心点の数
maxsubc_inner	integer	内側サブチャンネル中心点の数
maxsgap	integer	ギャップ線の数
maxsgap_outer	integer	外側ギャップ線の数
maxsgap_inner	integer	内側ギャップ線の数
msubc_outer(:)	integer	外側サブチャンネル中心点の番号
msgap_outer(:)	integer	外側ギャップ線の番号
msubc_fpin(:,:)	integer	燃料ピン-サブチャンネル中心点接続データ
msgap_fpin(:,:)	integer	燃料ピン-ギャップ線接続データ
mfpin_subc(:,:)	integer	サブチャンネル中心点-燃料ピン接続データ
msgap_subc(:,:)	integer	サブチャンネル中心点-ギャップ線接続データ
mfpin_sgap(:,:)	integer	ギャップ線-燃料ピン接続データ
msubc_sgap(:,:)	integer	ギャップ線-サブチャンネル中心点接続データ
isalive_fpin(:)	logical	燃料ピンの存在判定フラグ
isalive_subc(:)	logical	サブチャンネル中心点の存在判定フラグ
isalive_sgap(:)	logical	ギャップ線の存在判定フラグ
pwire	real	ワイヤ巻き付けピッチ[m]
pfpin	real	燃料ピン配列ピッチ[m]
xclad(:,:)	real	燃料ピン中心点の X 座標値[m]
yclad(:,:)	real	燃料ピン中心点の Y 座標値[m]
dclad(:,:)	real	被覆管外径[m]
dfgap(:,:)	real	被覆管内径[m]
xplet(:,:)	real	燃料ペレット中心点の X 座標値[m]
yplet(:,:)	real	燃料ペレット中心点の Y 座標値[m]
dplet(:,:)	real	燃料ペレット外径[m]
dhole(:,:)	real	燃料ペレット内径[m]

表 A.7 主要変数リスト (2/6)

変数	型	解説
xwire(:,:)	real	ワイヤ中心点の X 座標値[m]
ywire(:,:)	real	ワイヤ中心点の Y 座標値[m]
dwire(:,:)	real	ワイヤ外径[m]
xduct(:,:)	real	ラップ管中心点の X 座標値[m]
yduct(:,:)	real	ラップ管中心点の Y 座標値[m]
dduct(:)	real	ラップ管内対面間距離[m]
wduct(:)	real	ラップ管肉厚[m]
wwgap(:)	real	インターラップ領域ギャップ幅[m]
dwgap(:)	real	最外層燃料ピン-ラップ管間ギャップ幅[m]
zaxia(:)	real	水平断面の Z 座標値[m]
angle_wire(:,:)	real	ワイヤ位相[degree]
nwrap_wire(:)	integer	ワイヤ巻き付け方向
ifpin_start(:,:)	integer	ワイヤ接触点における燃料ピン周方向メッシュ番号
iduct_start(:,:)	integer	ラップ管頂点におけるラップ管周方向メッシュ番号
x_org	real	原点の X 座標値[m]
y_org	real	原点の Y 座標値[m]
z_org	real	原点の Z 座標値[m]
maxnode	integer	節点総数
maxbloc	integer	格子総数
maxelem	integer	要素総数
maxface	integer	境界総数
mnode_subc(:,:,:,,:)	integer	サブチャンネル中心点の節点番号リスト
mnode_sgap(:,:,:,,:)	integer	ギャップ線の節点番号リスト
mnode_fpin(:,:,:,,:)	integer	燃料ピン周辺領域の節点番号リスト
mnode_wrap(:,:,:,,:)	integer	ラップ管周辺領域の節点番号リスト
mnode_clad_surf(:,:,:,,:)	integer	燃料ピン境界層の節点番号リスト
mnode_wire_surf(:,:,:,,:)	integer	ワイヤ境界層の節点番号リスト
mnode_duct_surf(:,:,:,,:)	integer	ラップ管境界層の節点番号リスト
mnode_clad(:,:,:,,:)	integer	被覆管領域の節点番号リスト
mnode_fgap(:,:,:,,:)	integer	燃料ギャップ領域の節点番号リスト
mnode_plet(:,:,:,,:)	integer	燃料ペレット領域の節点番号リスト

表 A.7 主要変数リスト (3/6)

変数	型	解説
mnode_wire(:, :, :)	integer	ワイヤ領域の節点番号リスト
mnode_duct(:, :, :)	integer	ラップ管領域の節点番号リスト
mnode_wgap(:, :, :)	integer	インターラップ領域の節点番号リスト
minnode_fluid(:)	integer	流体領域における節点番号の最小値
maxnode_fluid(:)	integer	流体領域における節点番号の最大値
minnode_solid(:)	integer	構造材領域における節点番号の最小値
maxnode_solid(:)	integer	構造材領域における節点番号の最大値
mbloc_fpin(:, :, :)	integer	燃料ピン周辺領域の格子番号リスト
mbloc_wrap(:, :, :)	integer	ラップ管周辺領域の格子番号リスト
mbloc_clad_surf(:, :, :)	integer	燃料ピン境界層の格子番号リスト
mbloc_wire_surf(:, :, :)	integer	ワイヤ境界層の格子番号リスト
mbloc_duct_surf(:, :, :)	integer	ラップ管境界層の格子番号リスト
mbloc_clad(:, :, :)	integer	被覆管領域の格子番号リスト
mbloc_fgap(:, :, :)	integer	燃料ギャップ領域の格子番号リスト
mbloc_plet(:, :, :)	integer	燃料ペレット領域の格子番号リスト
mbloc_wire(:, :, :)	integer	ワイヤ領域の格子番号リスト
mbloc_duct(:, :, :)	integer	ラップ管領域の格子番号リスト
mbloc_wgap(:, :, :)	integer	インターラップ領域の格子番号リスト
minbloc_fluid(:)	integer	流体領域における格子番号の最小値
maxbloc_fluid(:)	integer	流体領域における格子番号の最大値
minbloc_solid(:)	integer	構造材領域における格子番号の最小値
maxbloc_solid(:)	integer	構造材領域における格子番号の最大値
maxpola	integer	要素次数 (=1)
maxaxia	integer	鉛直方向メッシュ分割数
mid_ismbc	integer	(=0)
mid_jsmbc	integer	(=0)
max_ismgap	integer	ギャップ線メッシュ分割数
mid_jsmgap	integer	(=0)
max_ifpin(:)	integer	燃料ピン周辺領域・周方向メッシュ分割数
max_jfpin(:)	integer	燃料ピン周辺領域・径方向メッシュ分割数
max_iwire(:)	integer	ワイヤ周辺領域・周方向メッシュ分割数

表 A.7 主要変数リスト (4/6)

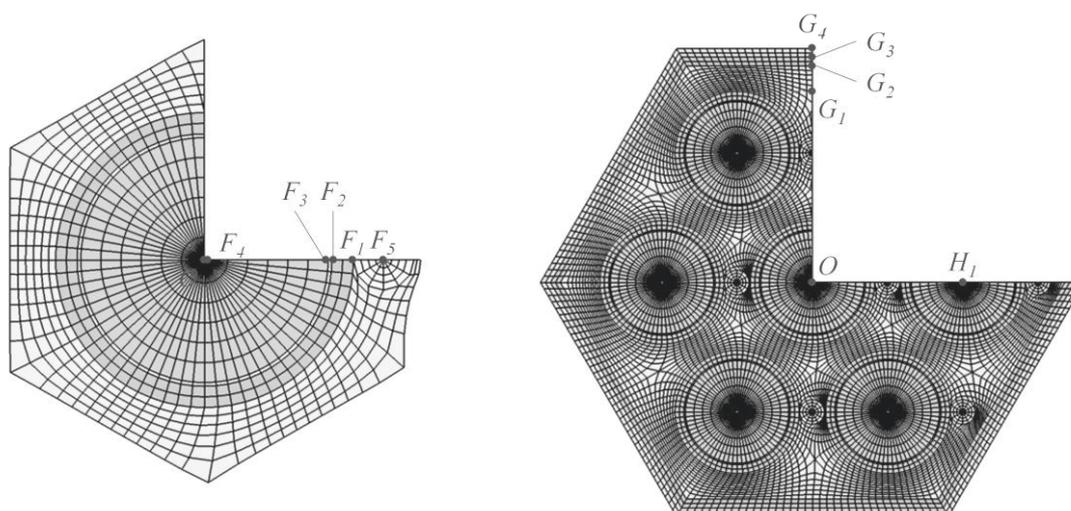
変数	型	解説
max_jwire(:)	integer	ワイヤ周辺領域・径方向メッシュ分割数
max_iwrap	integer	ラップ管周辺領域・周方向メッシュ分割数
max_jwrap	integer	ラップ管周辺領域・径方向メッシュ分割数
max_nsurf	integer	境界層径方向メッシュ分割数
max_nclad(:)	integer	燃料ピン境界層径方向メッシュ分割数
max_nwire(:)	integer	ワイヤ境界層径方向メッシュ分割数
max_nduct	integer	ラップ管境界層径方向メッシュ分割数
max_iclad(:)	integer	被覆管領域・周方向メッシュ分割数
max_jclad(:)	integer	被覆管領域・径方向メッシュ分割数
max_ifgap(:)	integer	燃料ギャップ領域・周方向メッシュ分割数
max_jfgap(:)	integer	燃料ギャップ領域・径方向メッシュ分割数
max_iplet(:)	integer	燃料ペレット領域・周方向メッシュ分割数
max_jplet(:)	integer	燃料ペレット領域・径方向メッシュ分割数
max_lwire(:)	integer	ワイヤ領域・周方向メッシュ分割数
max_mwire(:)	integer	ワイヤ領域・径方向メッシュ分割数
max_iduct	integer	ラップ管領域・周方向メッシュ分割数
max_jduct	integer	ラップ管領域・径方向メッシュ分割数
max_iwgap	integer	インターラップ領域・周方向メッシュ分割数
max_jwgap	integer	インターラップ領域・径方向メッシュ分割数
nnode_contact	integer	燃料ピン-ワイヤ接触点に配置する節点数
xnode(:)	real	節点の X 座標値[m]
ynode(:)	real	節点の Y 座標値[m]
znode(:)	real	節点の Z 座標値[m]
dnode(:)	real	節点座標値の残差[m]
qnode(:,:)	real	節点が属する要素の品質 (エッジ長、エッジ交差角)
delem_clad_surf(:,:,:)	real	燃料ピン境界層要素幅[m]
delem_wire_surf(:,:,:)	real	ワイヤ境界層要素幅[m]
delem_duct_surf(:,:)	real	ラップ管境界層要素幅[m]
maxiter	integer	SOR 法の最大反復計算回数
criteri	real	SOR 法の収束判定値
aiter	real	SOR 法の反復計算進捗率

表 A.7 主要変数リスト (5/6)

変数	型	解説
omega	real	Steger-Sorenson 法の緩和係数 (=1.0)
relax	real	SOR 法の緩和係数
efpin	real	燃料ピン表面のメッシュ緩和係数
ewrap	real	ラップ管表面のメッシュ緩和係数
rhole	real	燃料ピン中空率
delem_wall	real	境界層要素幅[m]
delem_min	real	エッジ長許容値[m]
delem_max	real	(未使用)
angle_thresh	real	エッジ交差角許容値[degree]
nnode_hexa	integer	Hexa 要素の節点数 (=8)
nnode_prism	integer	Prism 要素の節点数 (=6)
nnode_pyramid	integer	Pyramid 要素の節点数 (=5)
nnode_tetra	integer	Tetra 要素の節点数 (=4)
nnode_quad	integer	Quad 要素の節点数 (=4)
nnode_tri	integer	Triangle 要素の節点数 (=3)
mmate_bloc(:)	integer	格子の領域番号
mnode_bloc(:, :, :)	integer	格子の節点番号リスト
mtype_bloc(:)	integer	格子の格子分割タイプ
mmate_elem(:)	integer	要素の領域番号
nnode_elem(:)	integer	要素の節点数
mnode_elem(:, :, :)	integer	要素の節点番号リスト
mtype_face(:)	integer	境界の境界番号
nnode_face(:)	integer	境界の節点数
mnode_face(:, :, :)	integer	境界の節点番号リスト
islower(:)	logical	集合体下端境界の判定フラグ
isupper(:)	logical	集合体上端境界の判定フラグ
iswall(:)	logical	構造材表面境界の判定フラグ
iswall_clad(:)	logical	被覆管外表面境界の判定フラグ
iswall_wire(:)	logical	ワイヤ表面境界の判定フラグ
iswall_duct(:)	logical	ラップ管表面境界の判定フラグ
isgap_outer(:)	logical	被覆管内表面境界の判定フラグ

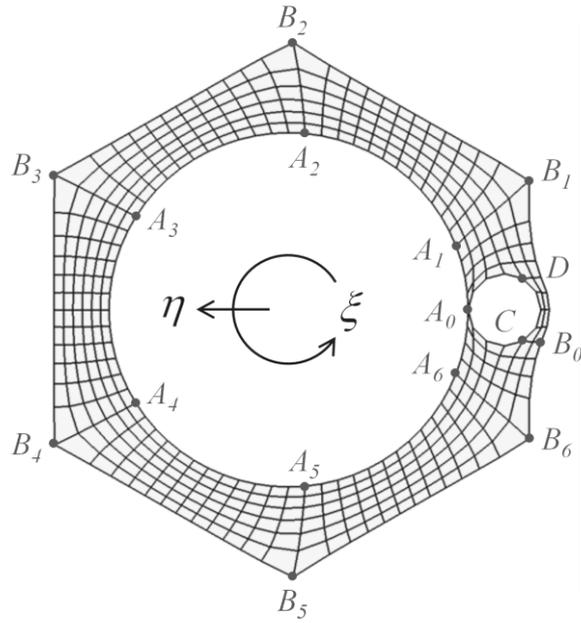
表 A.7 主要変数リスト (6/6)

変数	型	解説
isgap_inner(:)	logical	燃料ペレット表面境界の判定フラグ
iscenter_plet(:)	logical	燃料ペレット中心境界の判定フラグ
iscenter_wire(:)	logical	ワイヤ中心境界の判定フラグ
isexterior(:)	logical	インターラップ側面境界の判定フラグ
isblockage_face(:)	logical	閉塞物表面境界の判定フラグ
issymmetry_min(:)	logical	1/n セクタモデル・対称境界の判定フラグ (θ_{\min} 側)
issymmetry_max(:)	logical	1/n セクタモデル・対称境界の判定フラグ (θ_{\max} 側)
isfixed(:)	logical	節点固定境界の判定フラグ
n_sector	integer	1/n セクタモデル・セクタ数
asector_min	real	1/n セクタモデル・対称境界位相 (θ_{\min} 側) [degree]
asector_max	real	1/n セクタモデル・対称境界位相 (θ_{\max} 側) [degree]
asector_eps	real	1/n セクタモデル・対称境界位相の許容値[degree]
dsector_eps	real	1/n セクタモデル・節点間距離の許容値[m]
maxlock	integer	閉塞物領域の数
solid_or_not	logical	構造材モデルの ON/OFF
deform_or_not	logical	構造材変形モデルの ON/OFF
linear_or_not	logical	線形補間オプションの ON/OFF
contact_or_not	logical	構造材接触モデルの ON/OFF
recompute_or_not	logical	(未使用)
arate_thresh	real	構造材変形モデルにおける再分割比
txt_data_or_not	logical	FIDAP 型メッシュデータファイルの出力 ON/OFF
bin_data_or_not	logical	SPIRAL 型メッシュデータファイルの出力 ON/OFF
asm_data_or_not	logical	壁境界データファイルの出力 ON/OFF
stl_data_or_not	logical	STL 形式データファイルの出力 ON/OFF
debug_or_not	logical	確認用データファイルの出力 ON/OFF



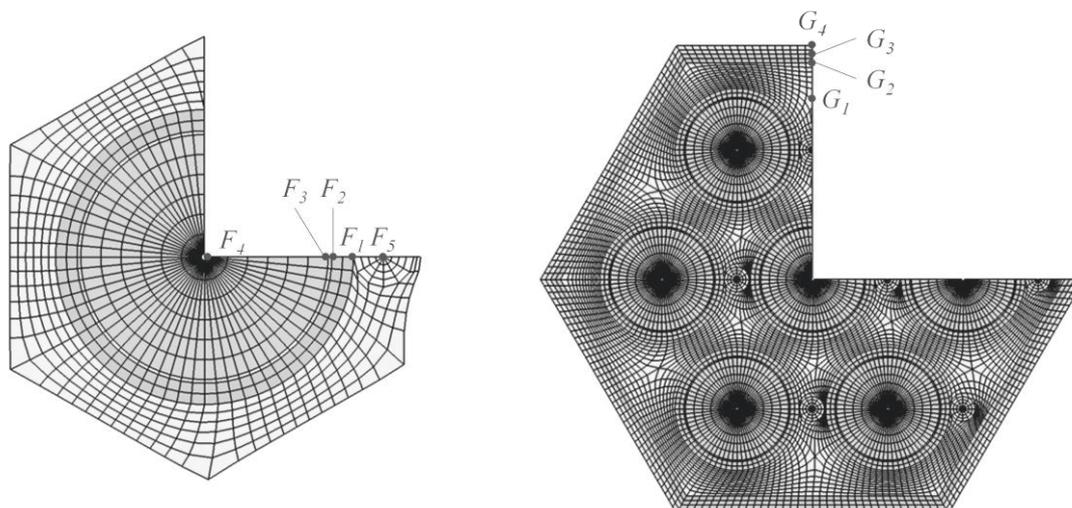
燃料ピン配列ピッチ[m]	$O-H_1$ 間距離
被覆管外径[m]	$O-F_1$ 間距離 $\times 2$
被覆管内径[m]	$O-F_2$ 間距離 $\times 2$
燃料ペレット外径[m]	$O-F_3$ 間距離 $\times 2$
燃料ペレット内径[m]	$O-F_4$ 間距離 $\times 2$
ワイヤ外径[m]	F_1-F_5 間距離 $\times 2$
最外層燃料ピン-ラップ管間ギャップ幅[m]	G_1-G_2 間距離
ラップ管対面間距離[m]	$O-G_2$ 間距離 $\times 2$
ラップ管肉厚[m]	G_2-G_3 間距離
インターラップ領域ギャップ幅[m]	G_3-G_4 間距離

図 A.1 燃料集合体仕様値の定義



ギャップ線メッシュ分割数	$B_1-B_2 = B_2-B_3 = \dots = B_6-B_1$ 間メッシュ数 (=12)
燃料ピン周辺領域・周方向メッシュ分割数	$B_1-B_2-B_3-B_4-B_5-B_6-B_1$ 間メッシュ数 (=72)
燃料ピン周辺領域・径方向メッシュ分割数	$A_1-B_1 = A_2-B_2 = \dots = A_6-B_6$ 間メッシュ数 (=6)
ワイヤ周辺領域・周方向メッシュ分割数	$C-D$ 間メッシュ数 (=4)
ワイヤ周辺領域・径方向メッシュ分割数	A_0-C 間メッシュ数 (=4)

図 A.2.1 流体領域メッシュ分割数の定義



被覆管領域・径方向メッシュ分割数	F_1 - F_2 間メッシュ数 (=2)
燃料ギャップ領域・径方向メッシュ分割数	F_2 - F_3 間メッシュ数 (=1)
燃料ペレット領域・径方向メッシュ分割数	F_3 - F_4 間メッシュ数 (=6)
ワイヤ領域・径方向メッシュ分割数	F_1 - F_5 間メッシュ数 (=4)
ラップ管周辺領域・径方向メッシュ分割数	G_1 - G_2 間メッシュ数 (=6)
ラップ管領域・径方向メッシュ分割数	G_2 - G_3 間メッシュ数 (=2)
インターラップ領域・径方向メッシュ分割数	G_3 - G_4 間メッシュ数 (=2)

図 A.2.2 構造材領域メッシュ分割数の定義

