

文部科学省次世代IT基盤構築のための研究開発
「革新的シミュレーションソフトウェアの研究開発」

RSS21 フリーソフトウェア

HEC ミドルウェア (HEC-MW)

HEC-MW 領域分割ユーティリティ

(**hecmw_part**) バージョン 2.01

領域分割ユーティリティ ユーザーズマニュアル

本ソフトウェアは文部科学省次世代IT基盤構築のための研究開発「革新的シミュレーションソフトウェアの研究開発」プロジェクトによる成果物です。本ソフトウェアを無償でご使用になる場合「RSS21 フリーソフトウェア使用許諾条件」をご了承頂くことが前提となります。営利目的の場合には別途契約の締結が必要です。これらの契約で明示されていない事項に関して、或いは、これらの契約が存在しない状況においては、本ソフトウェアは著作権法など、関係法令により、保護されています。

お問い合わせ先

(公開／契約窓口) (財)生産技術研究奨励会

〒153-8505 東京都目黒区駒場4-6-1

(ソフトウェア管理元) 東京大学生産技術研究所 計算科学技術連携研究センター

〒153-8505 東京都目黒区駒場4-6-1

Fax : 03-5452-6662

E-mail : software@rss21.iis.u-tokyo.ac.jp

目 次

1. ソフトウェア概要.....	2
1.1. はじめに.....	2
1.2. 領域分割手法.....	3
1.3. 領域分割タイプ.....	4
1.4. 可変オーバーラップ深さ.....	12
1.5. 領域分割イメージ図生成用 UCD ファイルの生成.....	12
2. 実行に必要なファイル.....	13
2.1 はじめに.....	13
4.2 領域分割する単一領域のメッシュデータファイル.....	13
4.3 全体制御ファイル.....	13
4.4 領域分割ユーティリティ制御ファイル.....	14
3. 実行方法.....	17
4. 適用例.....	18
4.1 単純立方体モデル (48 節点×48 節点×48 節点).....	18
4.2 赤門 (東京大学本郷キャンパス) モデル.....	20
4.3 本州モデル.....	22
4.4 黒鉛ブロックモデル.....	24
5. エラーメッセージ.....	26

1. ソフトウェア概要

1.1. はじめに

本ソフトウェアは、単一領域のメッシュデータを部分領域に領域分割し、並列有限要素法の計算に使用する分散メッシュデータを作成するユーティリティソフトウェアである。データ入力に HEC-MW ライブラリのデータ入力機能を用いており、同機能で読み込みが可能な単一領域のメッシュデータ「NASTRAN, ABAQUS, FEMAP Neutral または GeoFEM フォーマットメッシュデータ」「単一領域メッシュデータ」を任意の数の部分領域に領域分割する。また、HEC-MW ライブラリのデータ出力機能を用いて「分散メッシュデータ」の出力を行っており、HEC-MW を用いて開発したソフトウェアにおいては、本ソフトウェアにより作成した分散メッシュデータを用いて解析を行うことが可能である。

本ソフトウェアでは、単一領域メッシュの領域分割の実施の仕方として、節点単位で領域分割を行う「節点ベース分割」、および、要素単位で領域分割を行う「要素ベース分割」の 2 種類の領域分割タイプを実装している。領域分割手法としては、座標値を基準として部分領域への分割を行う RCB 法を用いた領域分割手法を実装しており、また、グラフ理論を領域分割に応用した MeTiS のライブラリへのインターフェイスを実装することで、同ライブラリにおける kMeTiS と pMeTiS を用いた領域分割も可能にしている。さらに、節点ベース分割においては、部分領域間のオーバーラップ深さを任意に設定することが可能であり、また、MicroAVS など利用可能な UCD ファイルフォーマットで領域分割イメージを表示するためのファイルを出力機能も実装されている。

以下、各機能について述べる。

1.2. 領域分割手法

本ソフトウェアにおいては、以下の領域分割手法が利用可能である。

- RCB

Recursive Coordinate Bisection の略であり、座標値の大小を基準に領域分割を行う方法である。高速で安定した手法であるが、部分領域数が 2^n に限定される。単純な形状においては、有効な手法である。

- MeTiS

高速、かつ安定した手法であり、複雑形状においても良好な領域分割が得られることから、世界中で広く利用されているオープンソースのフリーソフトウェアである。本ソフトウェアにおいては、この MeTiS のライブラリへのインターフェイスを備えており、pMeTiS と kMeTiS を直接利用して領域分割を行うことが可能である。但し、MeTiS が予め利用環境にインストールされていることが前提となる。MeTiS は以下の URL よりダウンロード可能である。

<http://www-users.cs.umn.edu/~karypis/metis/index.html>

1.3. 領域分割タイプ

本ソフトウェアにおいては、単一領域メッシュの部分領域への分割の仕方として、以下の2種類の領域分割タイプに対応している。

- 節点単位での領域分割（節点ベース分割）

図 1.3-1 に示すように、節点単位で領域分割を行う方法であり、この場合、全ての節点に対して唯一の所属部分領域が決定し、隣接する部分領域間でオーバーラップする要素が生じる。ゆえに、節点ベース分割においては、各部分領域は、以下に示す節点および要素の情報を所持する（図 1.3-2）。

- その部分領域に属している節点（内部節点）
- 内部節点を含む要素
- 内部節点を含む要素を構成する節点

また、隣接する部分領域間での通信に関する情報（通信テーブル）として、以下に示す情報を所持する。

- 輸入節点：部分領域内の節点のうち、他の部分領域に属する節点
- 輸出節点：他の部分領域の輸入節点となっている内部節点
- 共有要素：他の部分領域と共有する要素

図 1.3-1 の第 2 番の部分領域における輸入節点、輸出節点、および共有要素を図 1.3-3、図 1.3-4、図 1.3-5 に示す。

- 要素単位での領域分割（要素ベース分割）

図 1.3-6 に示すように、要素単位で領域分割を行う方法であり、この場合、全ての要素に対して唯一の所属部分領域が決定し、隣接する部分領域間でオーバーラップする節点が生じる。ゆえに、要素ベース分割においては、各部分領域は、以下に示す節点および要素の情報を所持する（図 1.3-7）。

- その部分領域に属している要素（内部要素）
- 内部要素を構成する節点
- 内部要素を構成する節点を含む要素

また、隣接する部分領域間での通信に関する情報（通信テーブル）として、以下に示す情報を所持する。

- 輸入要素：部分領域内の要素のうち、他の部分領域に属する要素
- 輸出要素：他の部分領域の輸入要素となっている内部要素
- 共有節点：他の部分領域と共有する節点

図 1.3-7 の第 2 番の部分領域における輸入要素, 輸出要素, および共有節点を図 1.3-8, 図 1.3-9, 図 1.3-10 に示す。

どちらの領域分割タイプにおいても、その通信テーブルは領域分割ユーティリティが自動的に作成し、分散メッシュデータに書き込むため、ユーザは通信に関して考慮する必要はない。

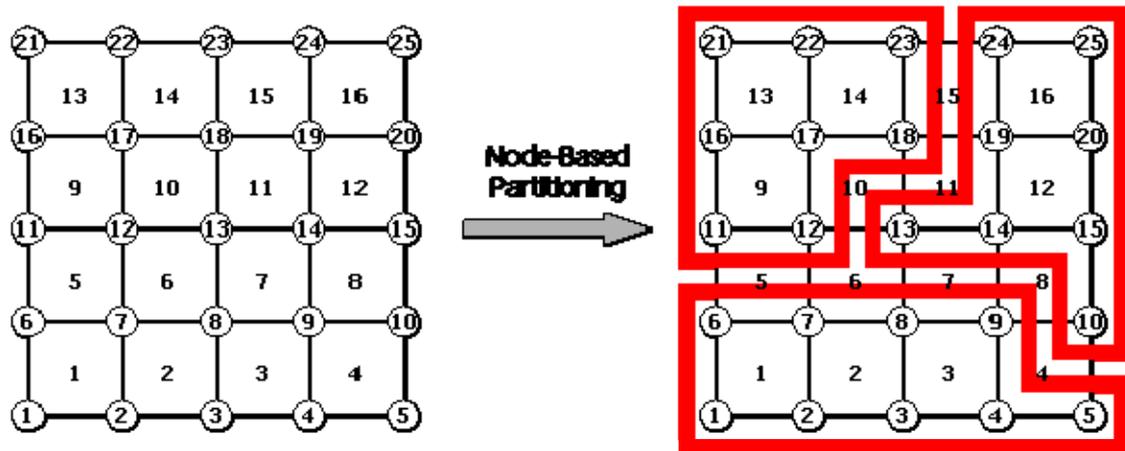


図 1.3-1 節点単位での領域分割

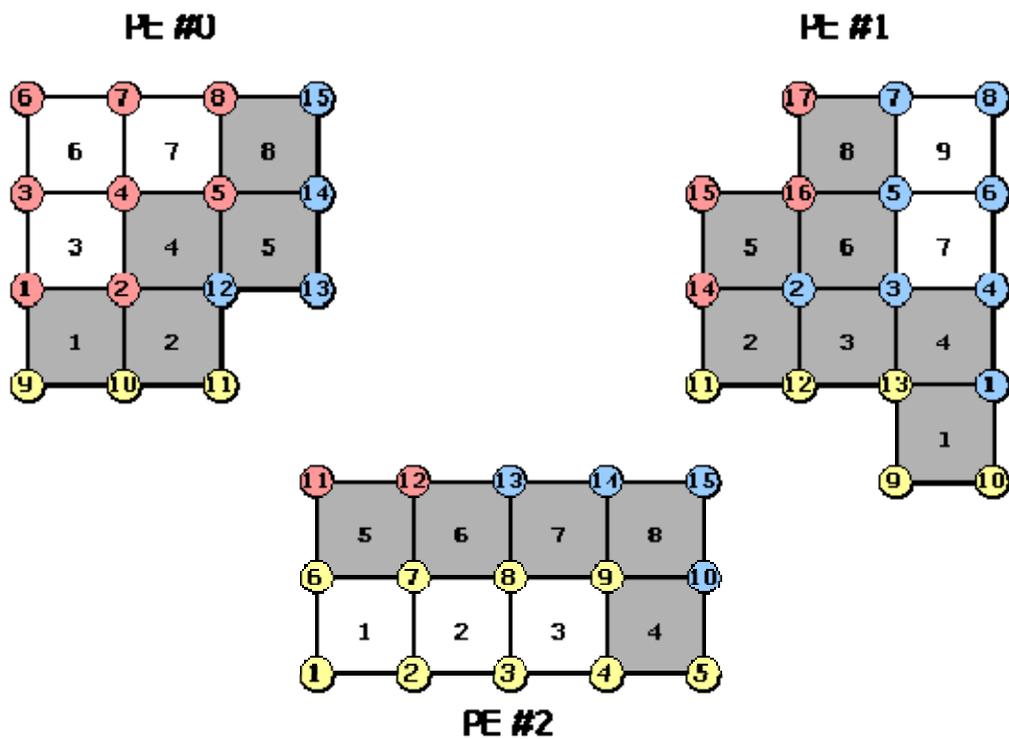


図 1.3-2 各部分領域が保持する節点および要素 (節点ベース分割)

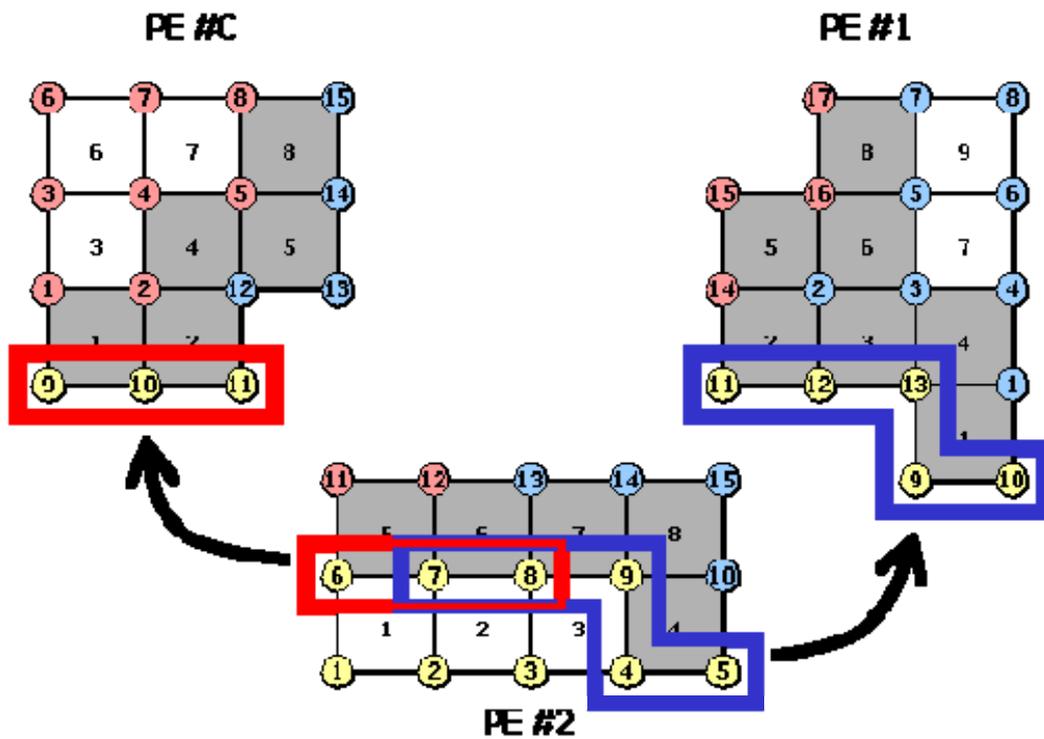


図 1.3-3 第 2 番の部分領域における入力節点

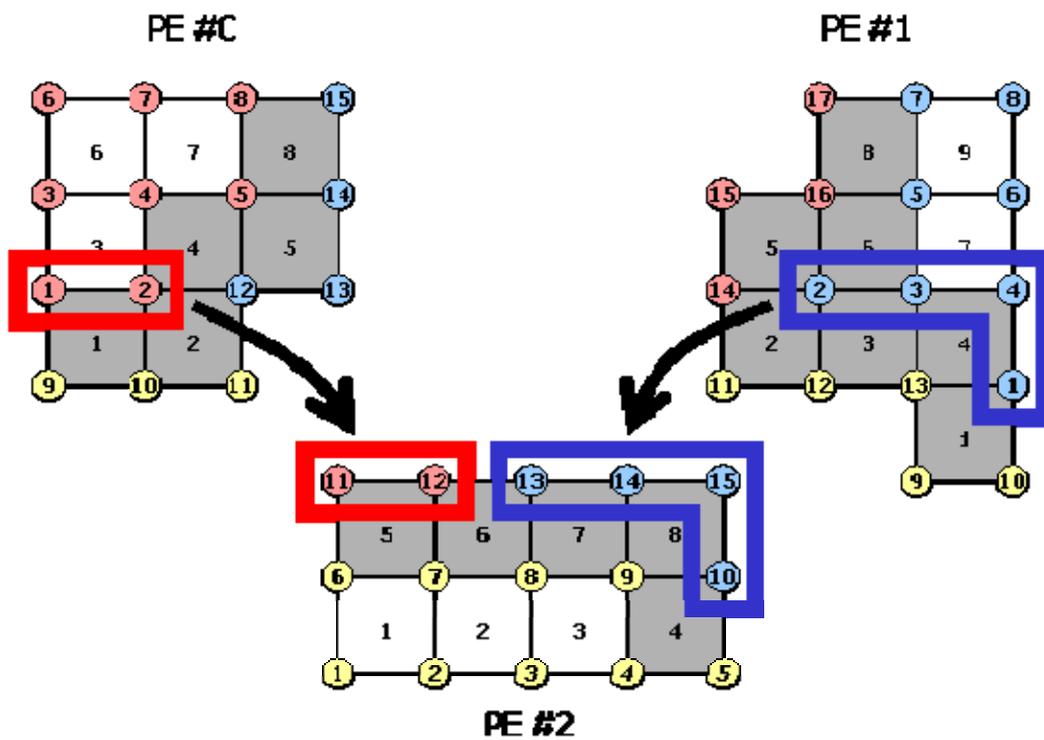


図 1.3-4 第 2 番の部分領域における出力節点

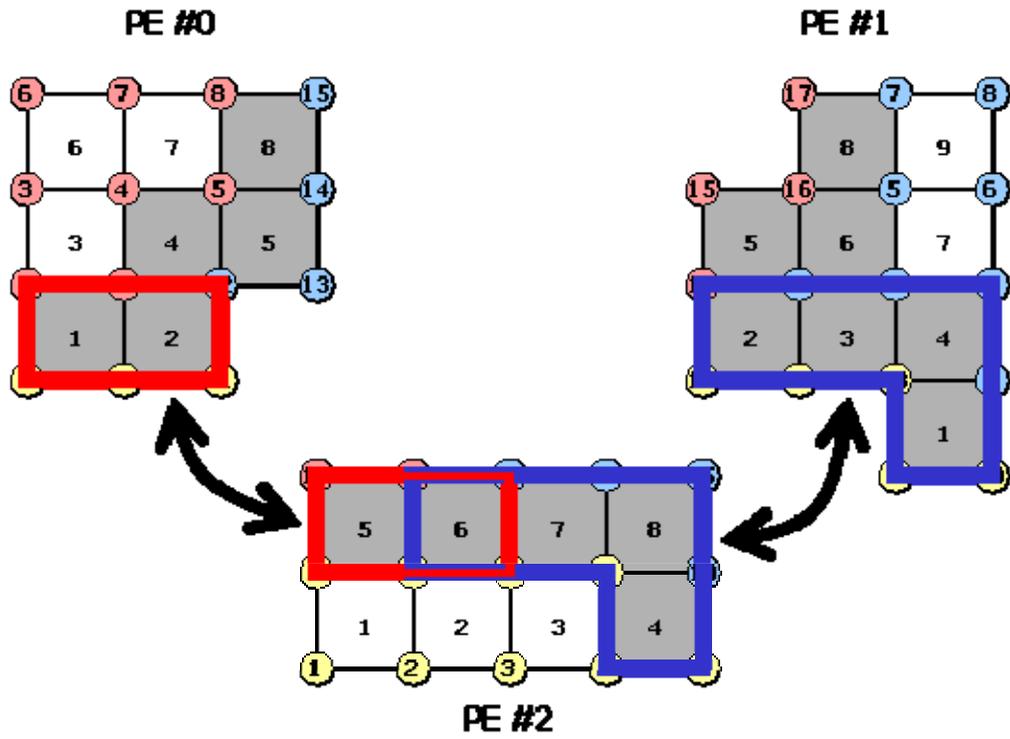


図 1.3-5 第 2 番の部分領域における共有要素

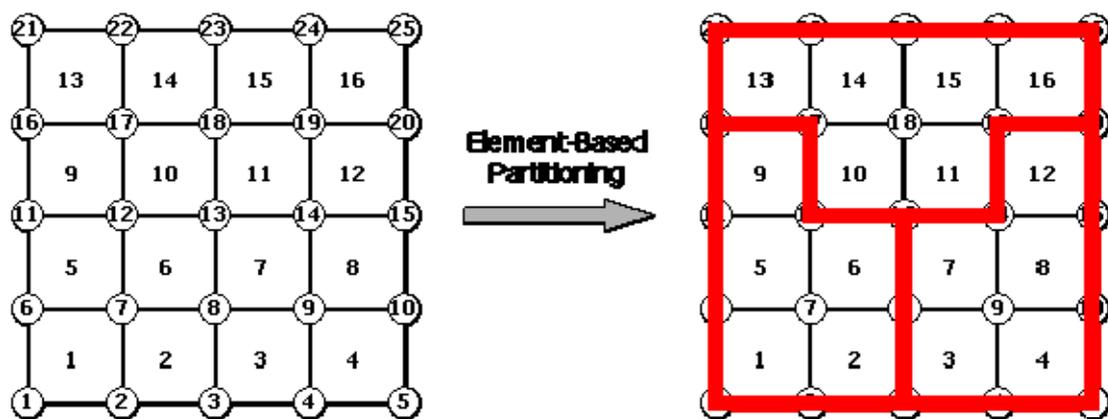


図 1.3-6 要素単位での領域分割

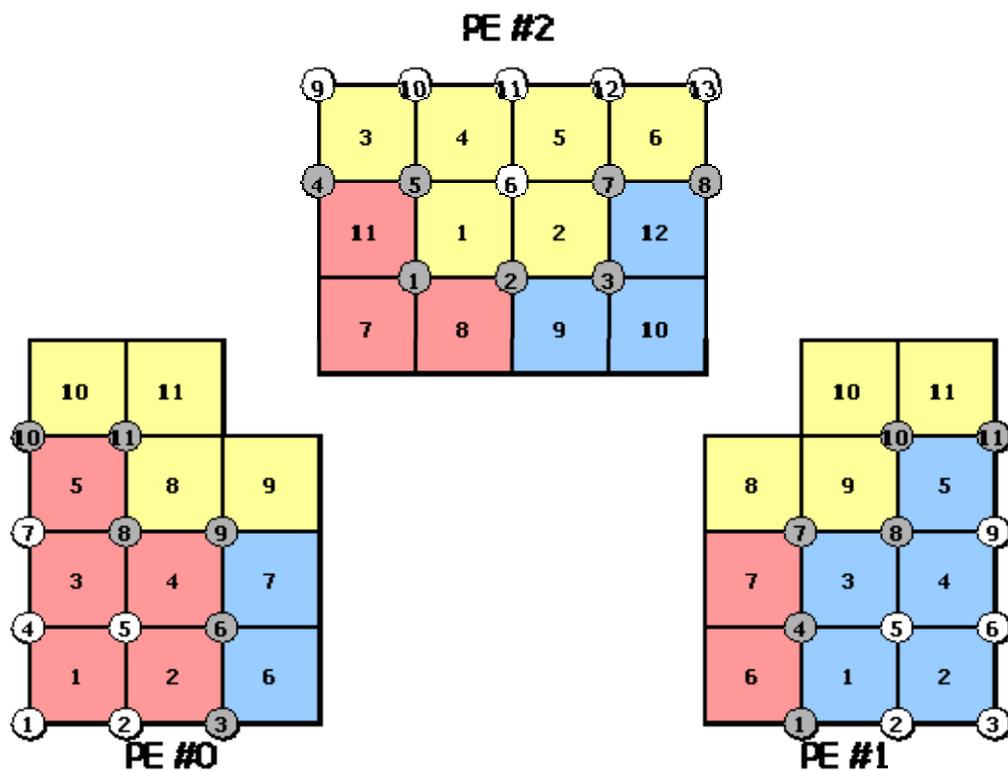


図 1.3-7 各部分領域が保持する節点および要素 (要素ベース分割)

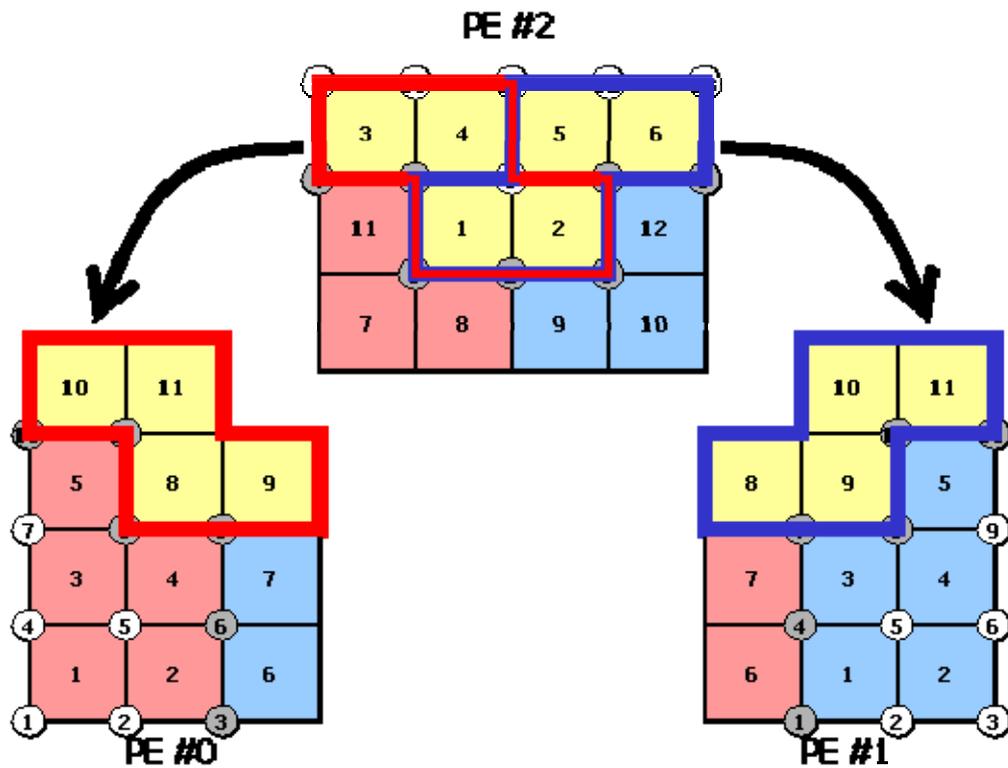


図 1.3-8 第 2 番の部分領域における入力要素

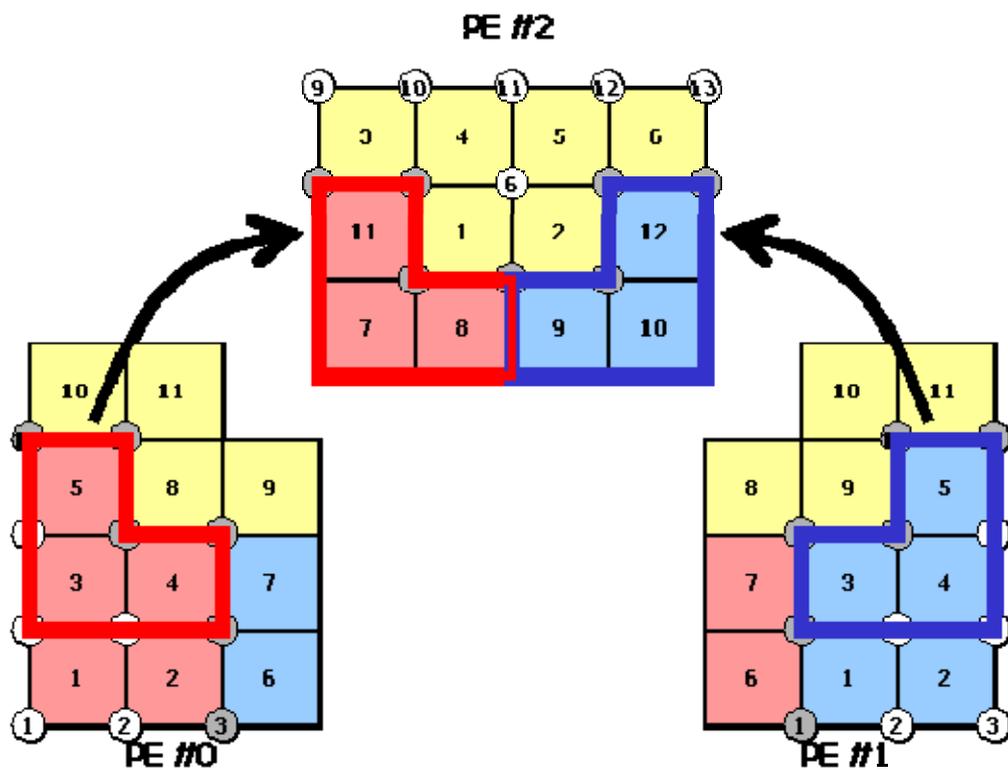


図 1.3-9 第 2 番の部分領域における出力要素

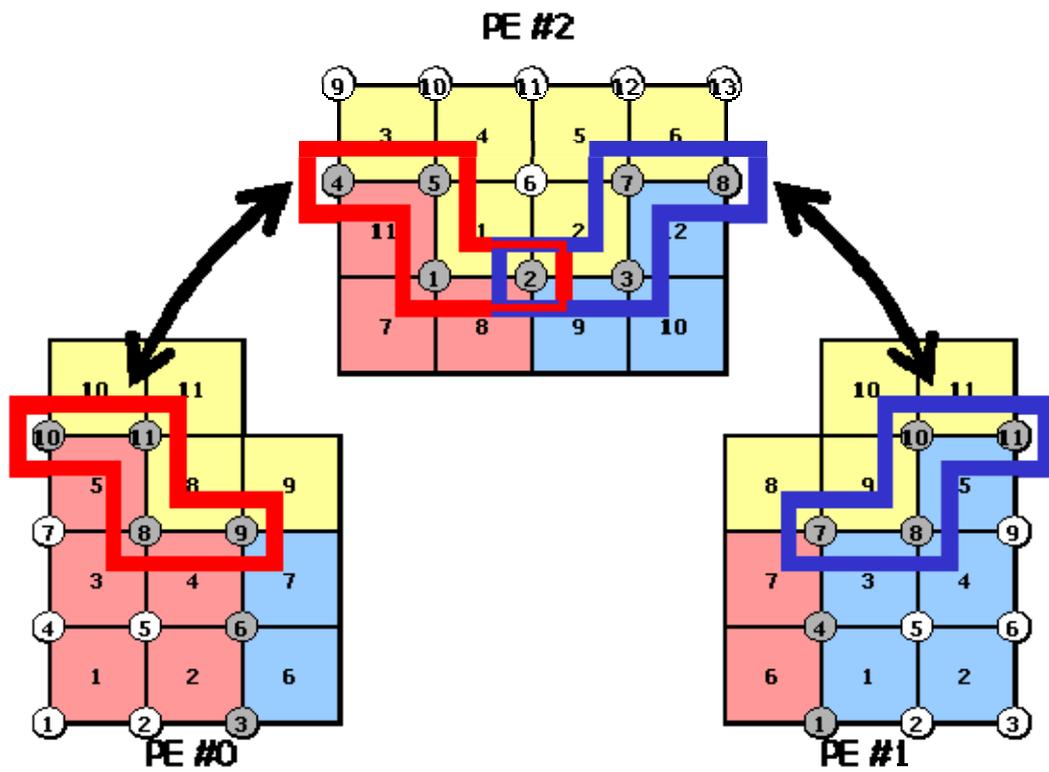


図 1.3-10 第 2 番の部分領域における共有節点

1.4. 可変オーバーラップ深さ

本ソフトウェアにおいては、部分領域間のオーバーラップ深さを任意に設定可能である（但し、領域分割タイプが節点単位での領域分割の場合に限られる）。通常、オーバーラップ深さは 1 で構わないのであるが、前処理に SAI (Sparse Approximate Inverse) を用いて接触問題や MPC 拘束問題等を解く際には、オーバーラップ深さを深くする必要が生じる。

1.5. 領域分割イメージ図生成用 UCD ファイルの生成

本ソフトウェアにおいては、MicroAVS 等において領域分割イメージ図を表示させるための UCD ファイルを出力することが可能である。

2. 実行に必要なファイル

2.1 はじめに

本ソフトウェアの実行にあたっては、以下のファイルが必要である。

- 領域分割する単一領域のメッシュデータファイル
- 全体制御ファイル
- 領域分割ユーティリティ制御ファイル

以下、これらのファイルについて述べる。

4.2 領域分割する単一領域のメッシュデータファイル

本ソフトウェアのデータ入力機能については HEC-MW ライブラリのデータ入力機能を利用しており、同機能が入力可能な単一領域のメッシュデータを入力データとして領域分割を行うことが可能である。

HEC-MW データ入力機能については「I/O, 全体制御プログラム使用説明書」を参照のこと。

4.3 全体制御ファイル

本ソフトウェアは、表示するメッシュデータの指定に HEC-MW の全体制御ファイルを用いている。本ソフトウェアの実行にあたっては、全体制御ファイルに以下の指定が必要である。

- 領域分割する単一領域のメッシュデータのファイル名
- 作成する分散メッシュデータのファイル名のヘッダー

以下に、全体制御ファイルの記述例を示す。

mesh.dat というファイル名の単一領域メッシュデータをを領域分割し、mesh.dist.0, mesh.dist.1, … というファイル名の分散メッシュデータを作成する場合、その全体制御

ファイルの記述は以下の通りである。

```
# 領域分割する単一領域メッシュデータの定義
!MESH, NAME=part_in, TYPE=HECMW-ENTIRE
mesh.dat

# 作成する分散メッシュデータの定義
!MESH, NAME=part_out, TYPE=HECMW-DIST
mesh.dist
```

ここで、領域分割する単一領域メッシュデータの定義（記述例 1 つ目の **!MESH**）において、その **NAME** オプションに指定する文字列は **part_in** に固定されている。また、作成する分散メッシュデータの定義（記述例 2 つ目の **!MESH**）において、その **NAME** オプションに指定する文字列は **part_out** に、**TYPE** オプションに指定するファイルフォーマットの指定子は **HECMW-DIST** に固定されている。これらのオプションに指定の文字列等が記述されていない場合にはエラーとなる。

なお、全体制御ファイルの詳細については「**I/O, 全体制御プログラム使用説明書**」を参照のこと。

4.4 領域分割ユーティリティ制御ファイル

本ソフトウェアにおいては、領域分割手法、領域分割数などの領域分割条件を「領域分割ユーティリティ制御ファイル」にて設定する。この制御マニュアルの記述法を以下に示す。

なお、記述上の注意（ファイル名で利用できる文字など）は **HEC-MW** 単一領域フォーマットのそれに従っているため「**I/O, 全体制御プログラム使用説明書**」も併せて参照されたい。また、この制御ファイルのデフォルトのファイル名は **hecmw_part_ctrl.dat** である。

適用例の章において、各例題の領域分割ユーティリティ制御ファイルの内容を示してある。記述の際に参考にされたい。

1 行目

!PARTITION, TYPE=<type>, METHOD=<method>, DOMAIN=<domain> [, optional parameter]

パラメータ	
TYPE	領域分割タイプ (必須)
METHOD	領域分割手法 (必須)
DOMAIN	領域分割数 (必須)
DEPTH	オーバーラップ深さ (省略可)
UCD	領域分割イメージ用 UCD ファイルの出力 (省略可)

パラメータ名	パラメータ値	内 容
TYPE	NODE-BASED	節点ベース分割
	ELEMENT-BASED	要素ベース分割
METHOD	RCB	RCB 法, 2 行目に分割基準軸の指定が必須
	KMETIS	kMETIS
	PMETIS	pMETIS
DOMAIN	<domain>	領域分割数
DEPTH	<depth>	部分領域のオーバーラップ深さ (省略時は DEPTH=1) TYPE=ELEMENT-BASED の時は指定不可
UCD	<filename>	UCD ファイル名 (省略可)

2 行目 (METHOD=RCB の場合のみ必須)

(2 行目) DIR1, DIR2, DIR3, ... (分割回数分指定)

変数名	属性	内容
DIRX	C	分割基準軸 (小文字の x, y, z で指定)

本制御ファイルの実例を以下に示す。

実例 1

```
!PARTITION, TYPE=NODE-BASED, METHOD=RCB, DOMAIN=8, DEPTH=1, UCD=mesh.inp  
x, y, z
```

実例 2

```
!PARTITION, TYPE=ELEMENT-BASED, METHOD=PMETIS, DOMAIN=32
```

3. 実行方法

本ソフトウェアの起動コマンドは以下の通り。

```
hecmw_part [-f <part_ctrl_filename>]
```

ここで、`-f` オプションにより、領域分割ユーティリティ制御ファイルのファイル名の指定が可能である。このオプションが省略された場合には、実行ディレクトリにあるデフォルトのファイル名 (`hecmw_part_ctrl.dat`) のファイルから領域分割ユーティリティ制御情報を読み込む。

また、本ソフトウェアの実行時には、全体制御ファイルは実行ディレクトリに、領域分割する単一領域メッシュデータファイルは全体制御ファイルで指定したディレクトリに配置する。

4. 適用例

4.1 単純立方体モデル (48 節点 × 48 節点 × 48 節点)

- 要素タイプ : 六面体 1 次要素
- 節点数 : 110,592
- 要素数 : 103,823

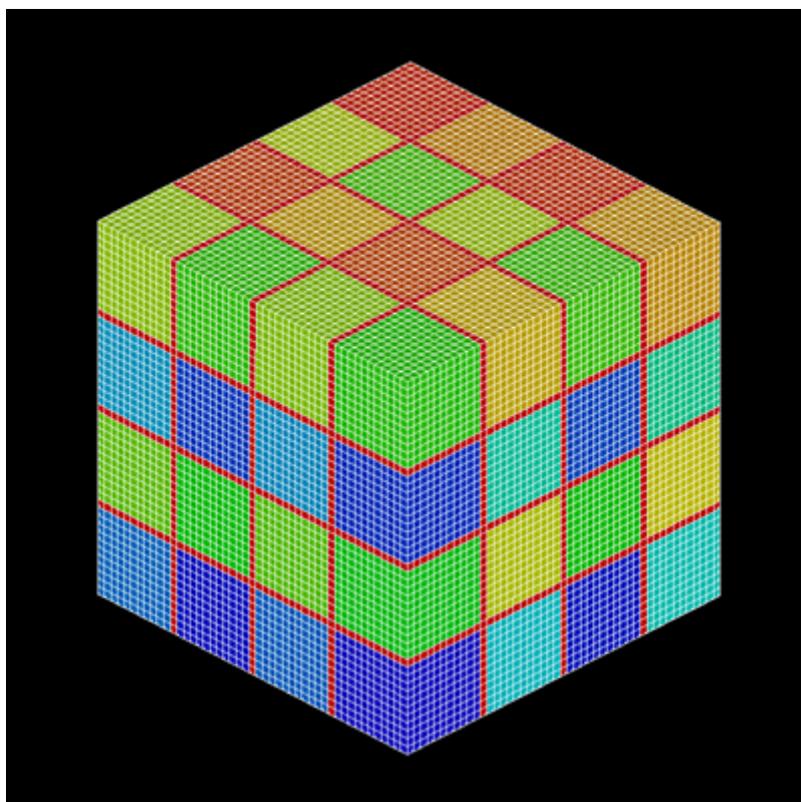


図 4.1-1 領域分割ユーティリティ適用例 1 : 単純立方体モデル
(節点ベース分割, RCB 法, edgecut 数 : 20,736/324,864)

- 領域分割ユーティリティ制御ファイル記述例

```
!PARTITION, TYPE=NODE-BASED, METHOD=RCB, DOMAIN=64, UCD=cube.rcb.inp  
x, y, z, x, y, z
```

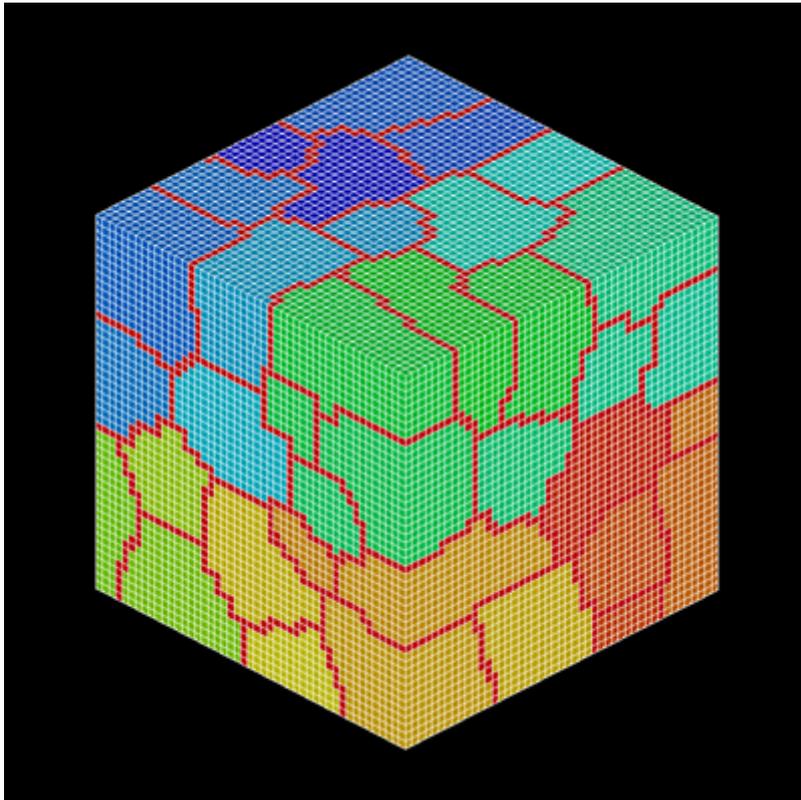


図 4.1-2 領域分割ユーティリティ適用例 1 : 単純立方体モデル
(節点ベース分割, kMETIS, edgcut 数 : 26,160/324,864)

- 領域分割ユーティリティ制御ファイル記述例

```
!PARTITION, TYPE=NODE-BASED, METHOD=KMETIS, DOMAIN=64, UCD=cube.kmetis.inp
```

4.2 赤門（東京大学本郷キャンパス）モデル

- 要素タイプ : 四面体一次要素
- 節点数 : 3,550
- 要素数 : 10,156



図 4.2-1 領域分割ユーティリティ適用例 2 : 赤門モデル
(節点ベース分割, RCB 法, edgecut : 4,396/17,180)

- 領域分割ユーティリティ制御ファイル記述例

```
!PARTITION, TYPE=NODE-BASED, METHOD=RCB, DOMAIN=32, UCD=akamon.rcb.inp  
x, y, z, x, y
```

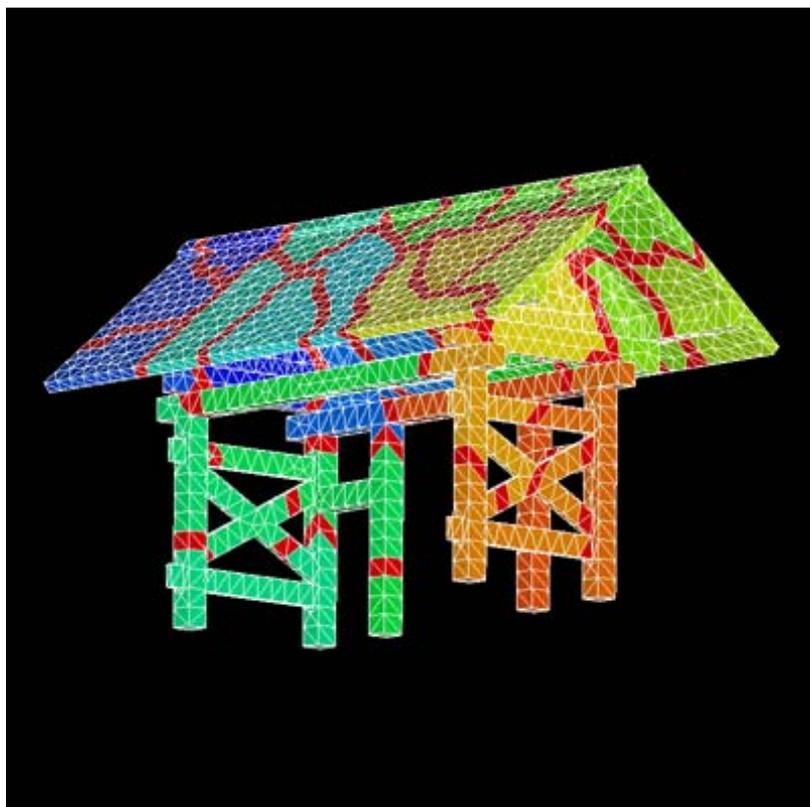


図 4.2-2 領域分割ユーティリティ適用例 2 : 赤門モデル
(節点ベース分割, pMETIS, edgecut : 2,119/17,180)

- 領域分割ユーティリティ制御ファイル記述例

```
!PARTITION, TYPE=NODE-BASED, METHOD=PMETIS, DOMAIN=32, UCD=akamon.kmetis.inp
```

4.3 本州モデル

- 要素タイプ : 三角形一次要素
- 節点数 : 21,285
- 要素数 : 40,548

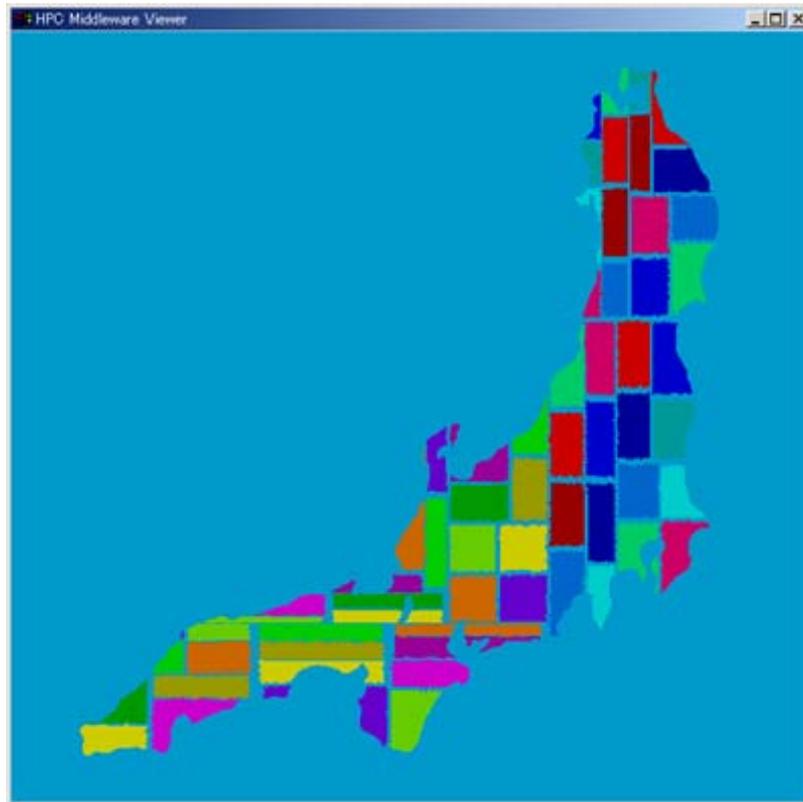


図 4.3-1 領域分割ユーティリティ適用例 3 : 本州モデル
(要素ベース分割, RCB 法, edgecut : 17,241/237,627)

- 領域分割ユーティリティ制御ファイル記述例

```
!PARTITION, TYPE=ELEMENT-BASED, METHOD=RCB, DOMAIN=64, UCD=honshu.rcb.inp  
x, y, x, y, x, y
```

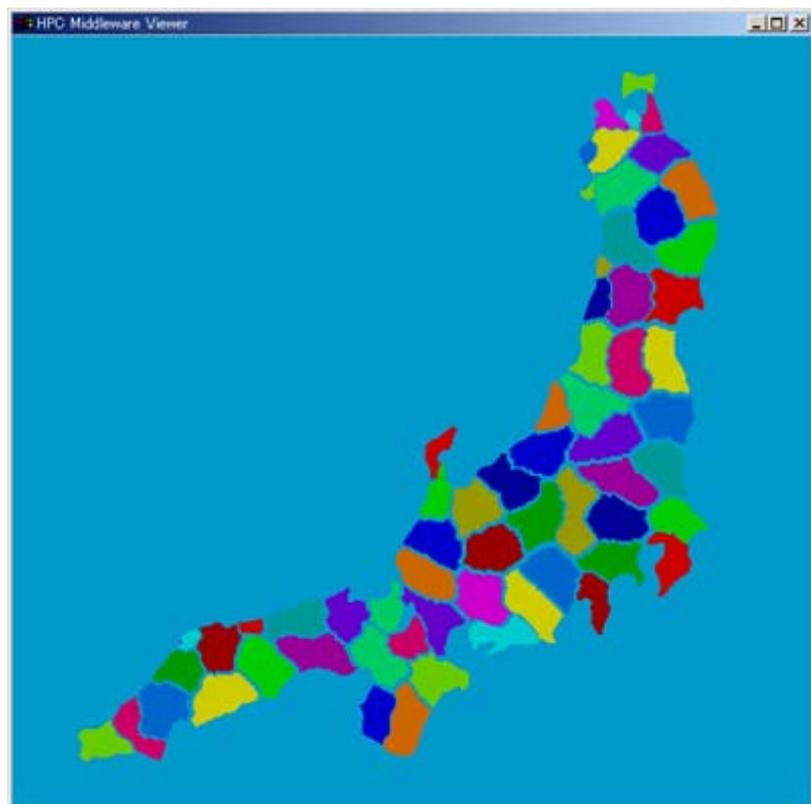


図 4.3-2 領域分割ユーティリティ適用例 3 : 本州モデル
(要素ベース分割, kMETIS, edgcut : 11,657/237,627)

- 領域分割ユーティリティ制御ファイル記述例

```
!PARTITION, TYPE=ELEMENT-BASED, METHOD=KMETIS, DOMAIN=64, UCD=honshu.kmetis.inp
```

4.4 黒鉛ブロックモデル

- 要素タイプ : 六面体一次要素
- 節点数 : 1,308
- 要素数 : 795

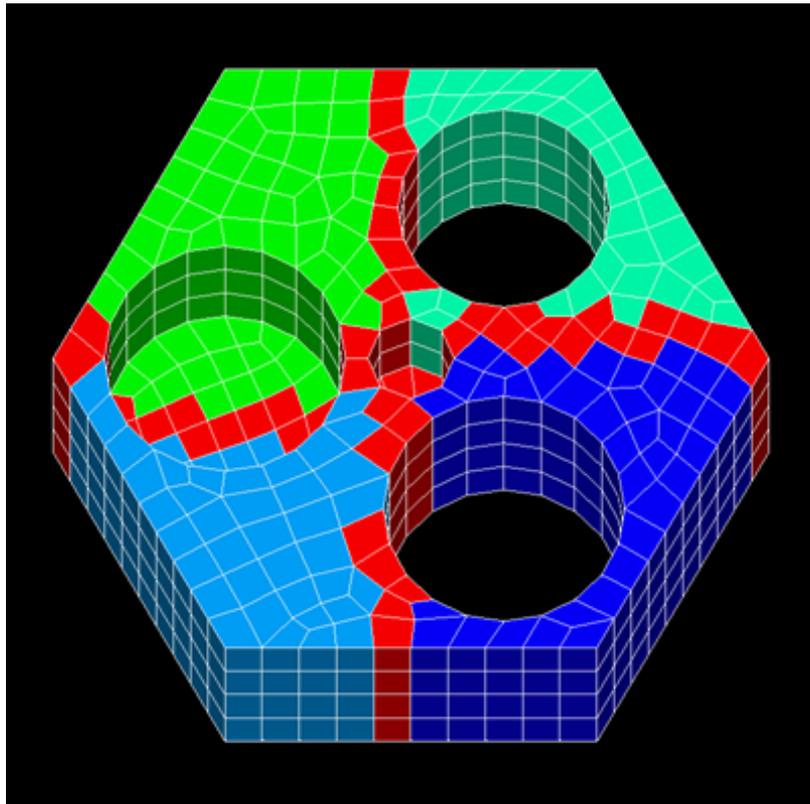


図 4.4-1 領域分割ユーティリティ適用例 4 : 黒鉛ブロックモデル
(オーバーラップ深さ変更例, DEPTH=1)

- 領域分割ユーティリティ制御ファイル記述例

```
!PARTITION, TYPE=NODE-BASED, METHOD=RCB, DOMAIN=4, DEPTH=1, UCD=carbon.1.inp  
x, y
```

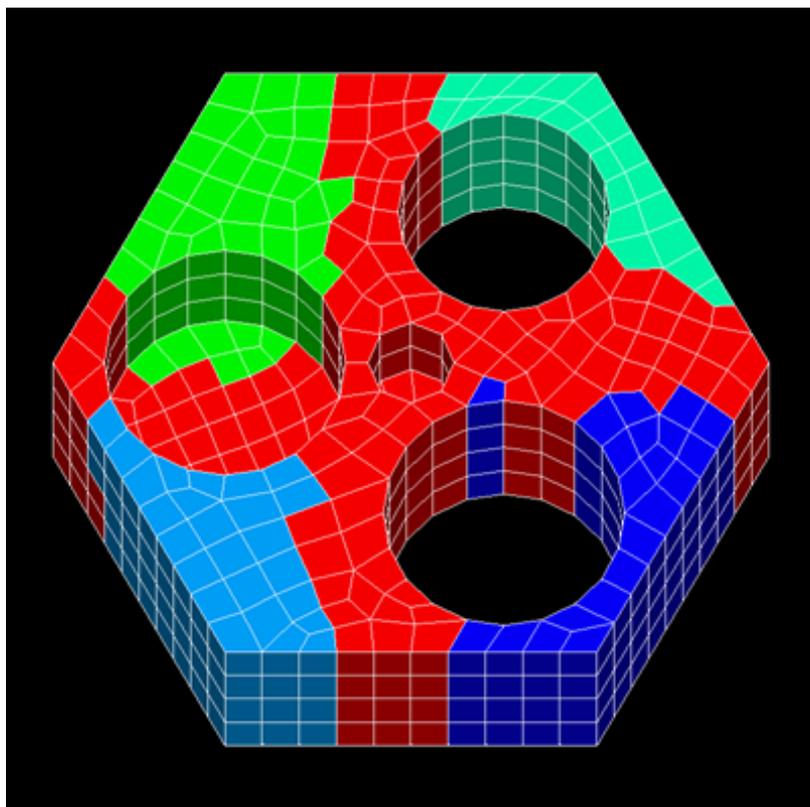


図 4.4-2 領域分割ユーティリティ適用例 4：黒鉛ブロックモデル
(オーバーラップ深さ変更例, DEPTH=2)

- 領域分割ユーティリティ制御ファイル記述例

```
!PARTITION, TYPE=NODE-BASED, METHOD=RCB, DOMAIN=4, DEPTH=2, UCD=carbon. 2. inp  
x, y
```

5. エラーメッセージ

HECMW-PART-E0001: No such file or directory

指定したファイルが存在しない

HECMW-PART-E0002: File close error

ファイルを閉じる際に予期せぬエラーが発生した

HECMW-PART-E0003: Too long file name

ファイル名が長すぎる

HECMW-PART-E0004: NULL pointer was found

不正なヌルポインタを検出した

HECMW-PART-E0005: Invalid EOF is found

予期せぬ EOF を検出した

HECMW-PART-E0006: Invalid argument is found

不正な引数を検出した

HECMW-PART-E0101: Invalid token found

無効なトークンを検出した

HECMW-PART-E0111: 'TYPE' must not be omitted

!PARTITION の必須のオプションである'TYPE'が見つからない

HECMW-PART-E0121: 'METHOD' must not be omitted

!PARTITION の必須のオプションである'METHOD'が見つからない

HECMW-PART-E0131: 'DOMAIN' must not be omitted

!PARTITION の必須のオプションである'DOMAIN'が見つからない

HECMW-PART-E0134: 'METHOD' is 'RCB', but 'DOMAIN' is not n-th power of 2

'METHOD'に'RCB'が指定されているが'DOMAIN'の値が2のべき乗でない

HECMW-PART-E0161: Partitioning directions are not enough

分割軸方向の指定が足りない

HECMW-PART-E0201: Invalid partitioning type is found

不正な領域分割タイプを検出した

HECMW-PART-E0202: Invalid partitioning method is found

不正な領域分割法を検出した

HECMW-PART-E0203: Invalid element type is found

不正な要素タイプを検出した

HECMW-PART-E0204: Invalid partitioning direction is found

不正な分割軸方向を検出した

HECMW-PART-E0205: Invalid number of sub-domains is found

不正な部分領域数を検出した

HECMW-PART-E0206: Invalid overlapping depth is found

不正な部分領域間オーバーラップ深さを検出した

HECMW-PART-E0211: Stack overflow

スタック配列がオーバーフローした

HECMW-PART-E0301: Domain number is outside of the range

部分領域番号が部分領域数を超えている

HECMW-PART-E0311: Number of nodes is outside of the range

不正な節点数を検出した

HECMW-PART-E0312: Number of internal nodes is outside of the range

不正な内部節点数を検出した

HECMW-PART-E0321: Number of elements is outside of the range

不正な要素数を検出した

HECMW-PART-E0322: Number of internal elements is outside of the range

不正な内部要素数を検出した

HECMW-PART-E0331: Number of neighboring sub-domains must be greater than or equal 0

隣接部分領域数が 0 より小さい

HECMW-PART-E0501: Log file has not initialized yet

ログファイルの初期化処理が行われていない

HECMW-PART-E9999: Fatal error

深刻なエラーが発生した

HECMW-PART-W0162: Too many partitioning directions are specified

必要数以上の分割軸方向が指定された

HECMW-PART-W0401: EQUATION_BLOCK is not found

EQUATION_BLOCK が見つからない

HECMW-PART-W0502: Log file for partitioner has already initialized

ログファイルの初期化処理が複数回呼び出された