

平成28年度第二回賛助会員交流会(利推協)

@計算科学センタービル

# FOCUSスパコン上での 流体解析における iconCFDの並列性能

2017年2月24日

三菱電機株式会社 先端技術総合研究所  
熱流体システム技術部 流体・伝熱制御グループ  
永安 哲也

# 目次

1. 目的
2. iconCFDとは
3. ベンチマーク方法
4. 解析対象
5. 解析条件
6. 解析メッシュ
7. 解析結果
8. ベンチマーク結果
9. まとめ

# 1. 目的

FOCUSスパコンのAシステムとDシステムを使用し、オープンソースベースの汎用CFDソフトウェアiconCFDの並列性能ベンチマークテストを実施する。

## 2. iconCFDとは

Icon Technology & Process Consulting Ltd.とIDAJ社により開発されたOpenFOAM®をベースに開発されたGPLに準ずる汎用のCFDプログラムである。(iconCFDの一部には、非GPLコードを含む。) 解析モデルの自動生成機能を有し、収束性が高く、OpenFOAMにおける既知のバグが修正されている。また、ツールとしての使いやすさに配慮した改良が加えられている。その他の主な特徴は、以下の通り。

- ・低コストでの大規模計算が実現可能
- ・ソースレベルでのカスタマイズが可能  
OpenFOAMをベースとした新たな物理モデルの組み込みや各種物理機能を組み合わせた新たなソルバーの作成が可能
- ・商用レベルのメンテナンスと技術サポート  
定期的なバージョンアップ、品質の保持、デバッグ、バージョン管理が実施される
- ・C++言語  
OpenFOAMと同じ言語での開発

## 2. iconCFDとは

※ 株式会社IDAJの許可を得て改編

### iconCFD のサービスパッケージ

#### モジュール

必要な機能毎にモジュールを選択  
以下のような改良が行われている

- メッシュ品質の向上
- ソルバーの収束性向上
- バグ修正
- 各種機能向上
- 使い勝手の向上



#### 保守

以下の内容が含まれる

- バグフィックス
- 機能向上(バージョンアップ)
- ドキュメントの提供



#### 技術サポート

プリペイドカード方式による技術サポート。以下の時間を選択

- 50時間・・・サポート申請者: 1人
- 100時間・・・サポート申請者: 2人
- 200時間・・・サポート申請者: 3人



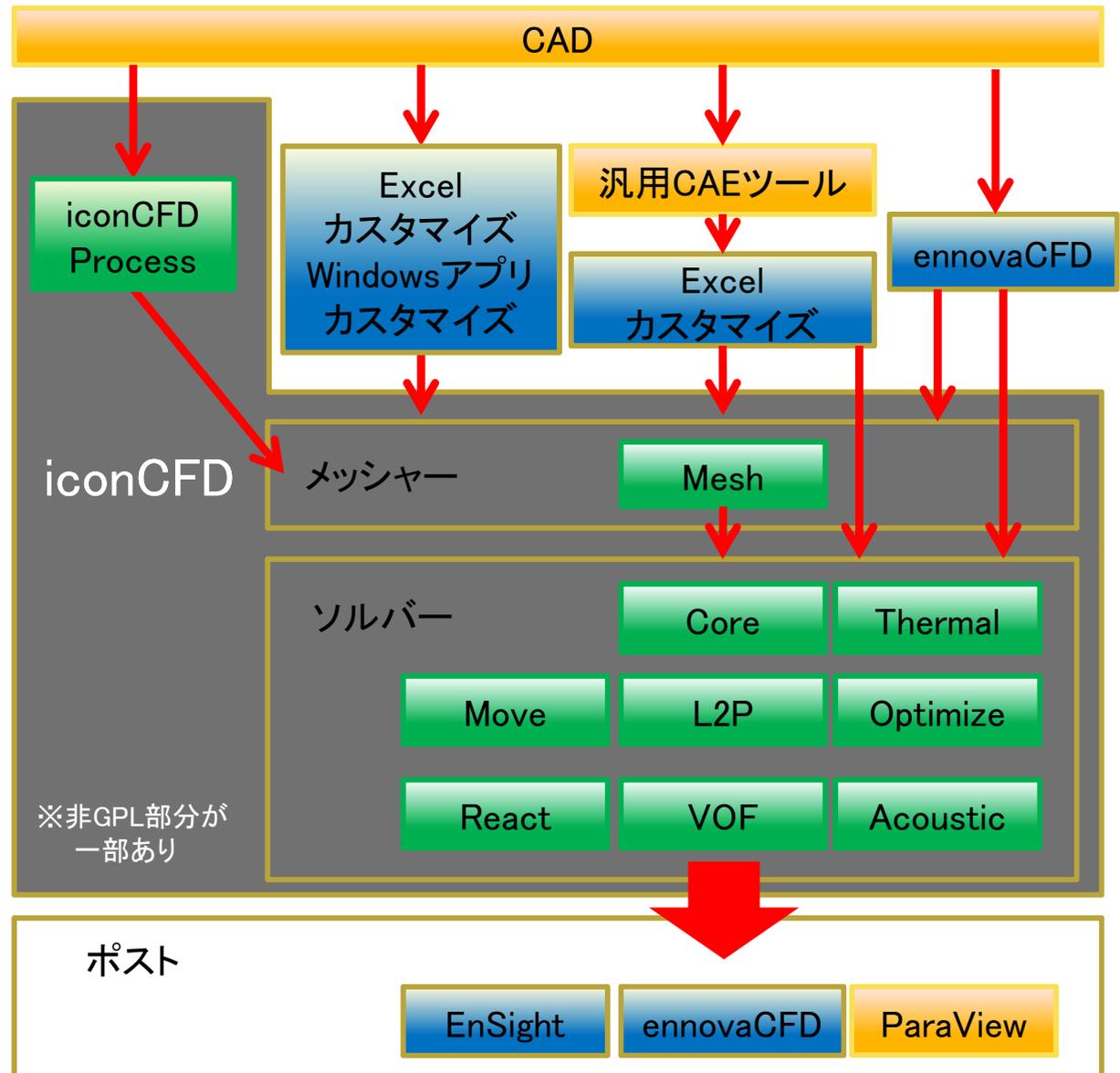
#### カスタマイズ

独自のカスタマイズが可能

- 物理モデルのカスタマイズ
- 他CAEシステムとの連携・連成
- 境界条件設定のGUIの作成
- 解析システムの構築



### サービス提供のモジュール構成図



### 3. ベンチマーク方法

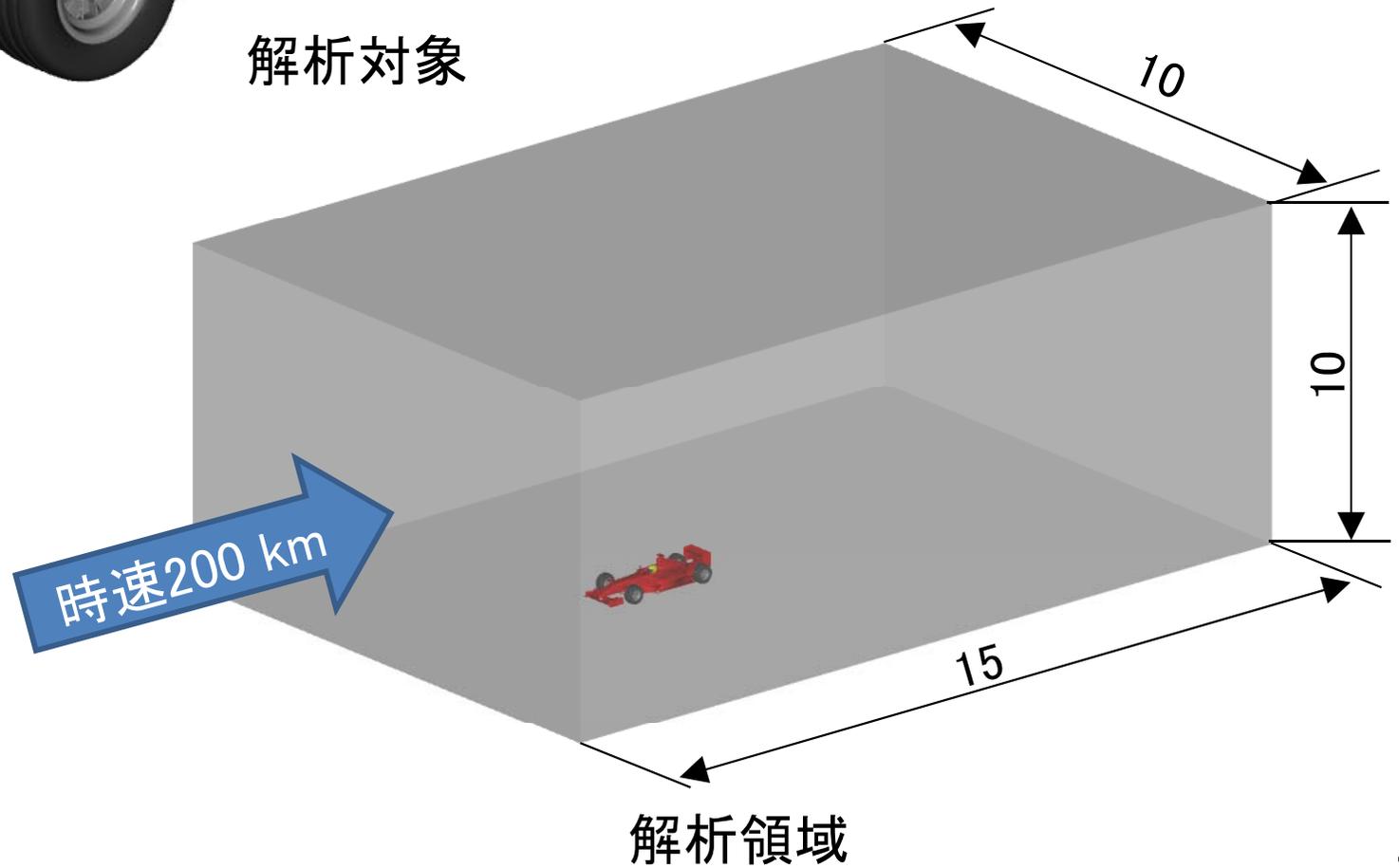
同一形状の流体解析モデルをメッシュ数・並列数を変更し、それぞれ iconCFDによる流体解析を実施する。収束までに要した時間ではなく、実行開始から500イタレーションの計算時間で比較する。

ここで、MPI(Message Passing Interface)は、iconCFD付属のintelMPIを使用した。環境は、通常運用中のFOCUSスパコン(Aシステム, Dシステム)とする。他のジョブの影響を小さくするため、原則3回の繰り返し測定を実施し、平均値を使用した。(極端に計算速度の遅いものについては除外した。) iconCFDのバージョンは、v3.2.8を使用した。

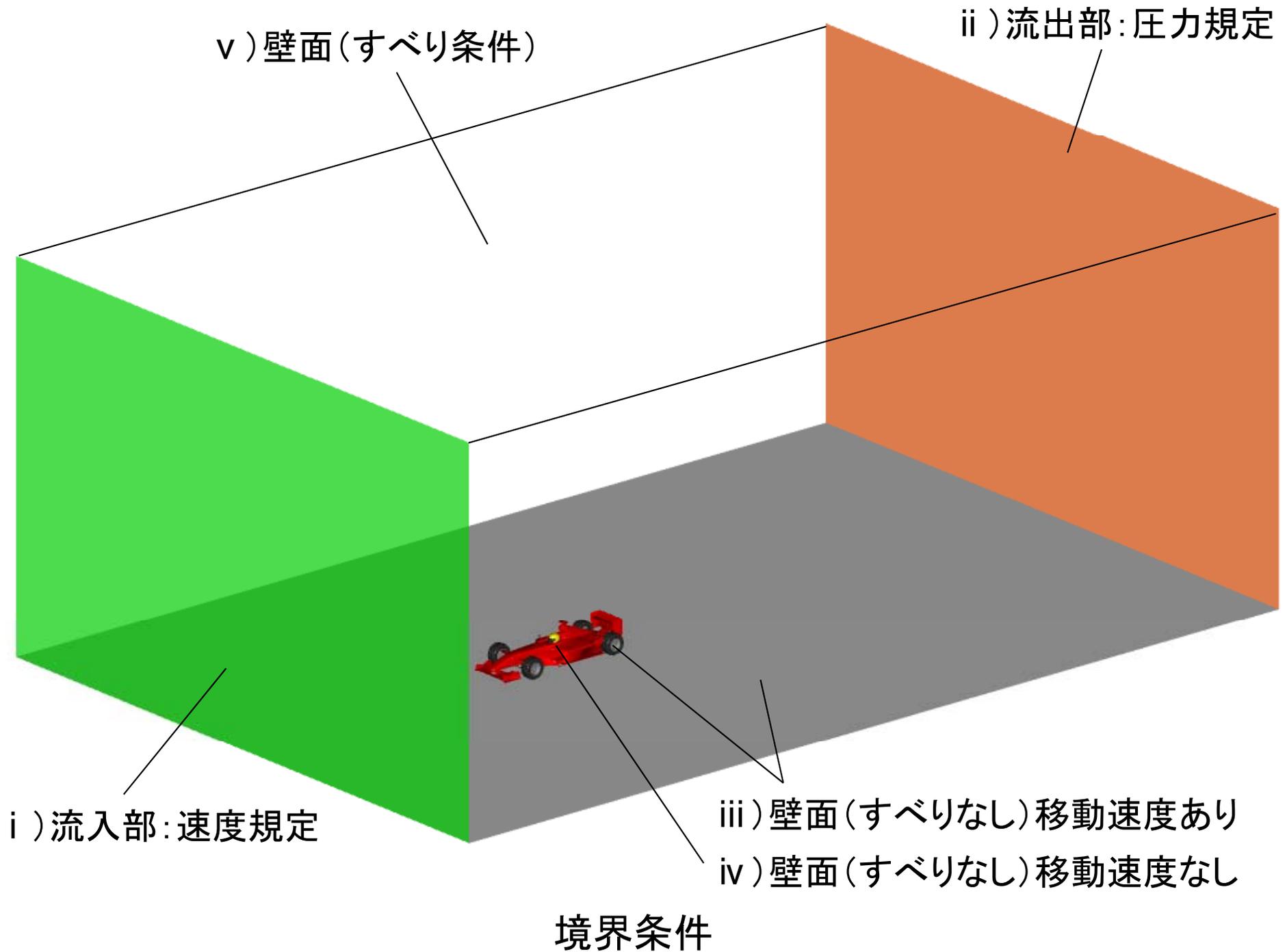
### 4. 解析対象

解析対象は、iconCFDの標準ベンチマークで用いられているレーシングカーモデルの外部流れとした。流れは、定常、非圧縮とし、乱流状態であるとした。200km/ hourの走行状態を仮定し、解析では、車体を固定、周囲流及び路面に相対的な速度を与えた。タイヤとホイールには、走行速度に合わせた回転速度を与えた。なお、流体の物性は、空気(密度: 1.205 [kg/m<sup>3</sup>], 動粘度:  $1.511 \times 10^{-5}$  [m<sup>2</sup>/s])とし、乱流モデルは、k- $\omega$  SSTを使用した。

## 4. 解析対象

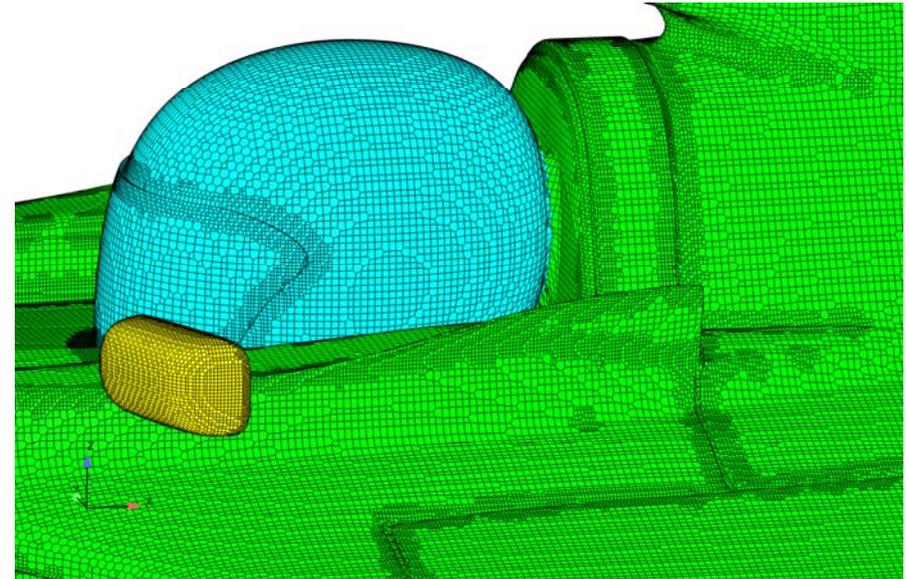
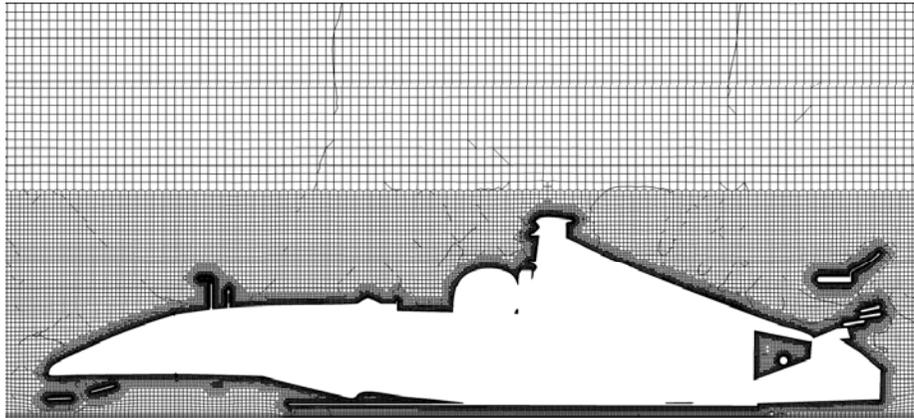


# 5. 解析条件

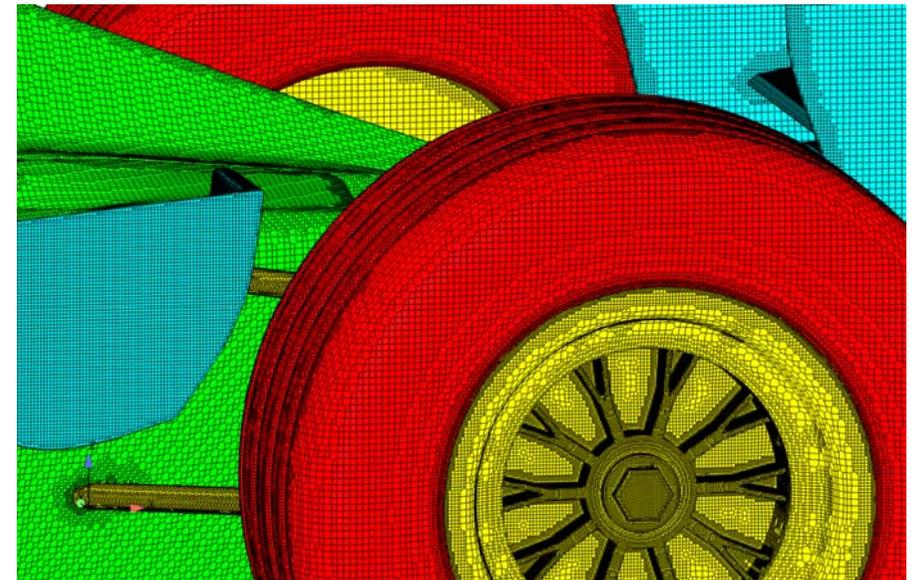
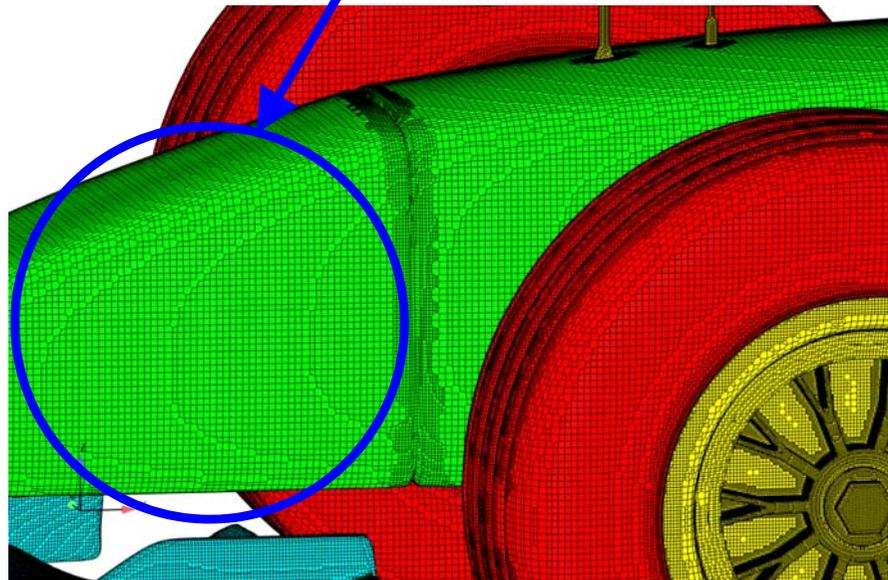


## 6. 解析メッシュ

① Coarse Mesh: 約2000万メッシュ (20,399,141メッシュ)

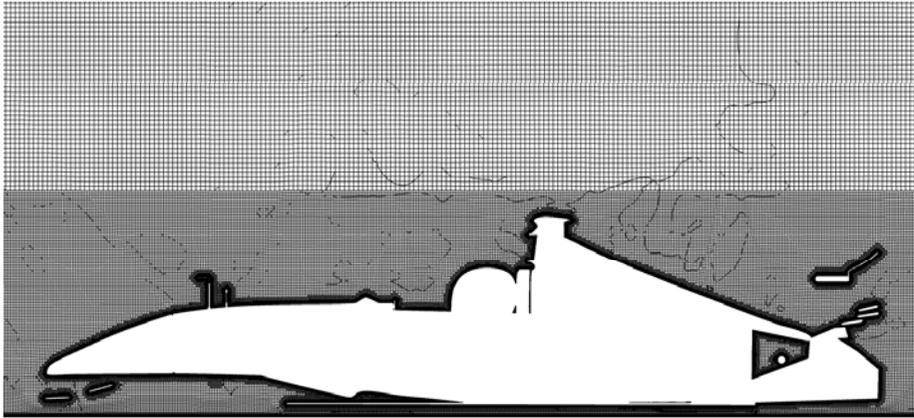


車体表面ベースサイズ:  
約2.8 [mm]

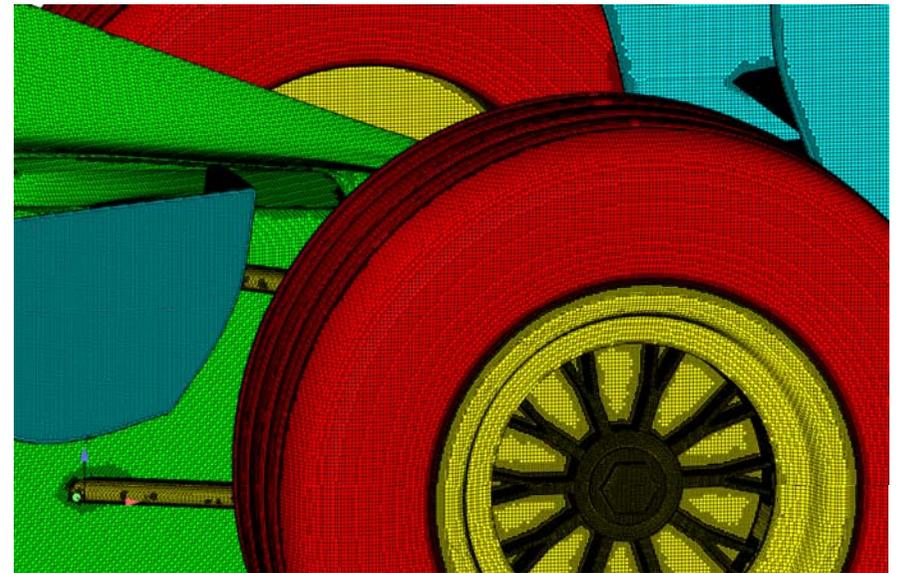
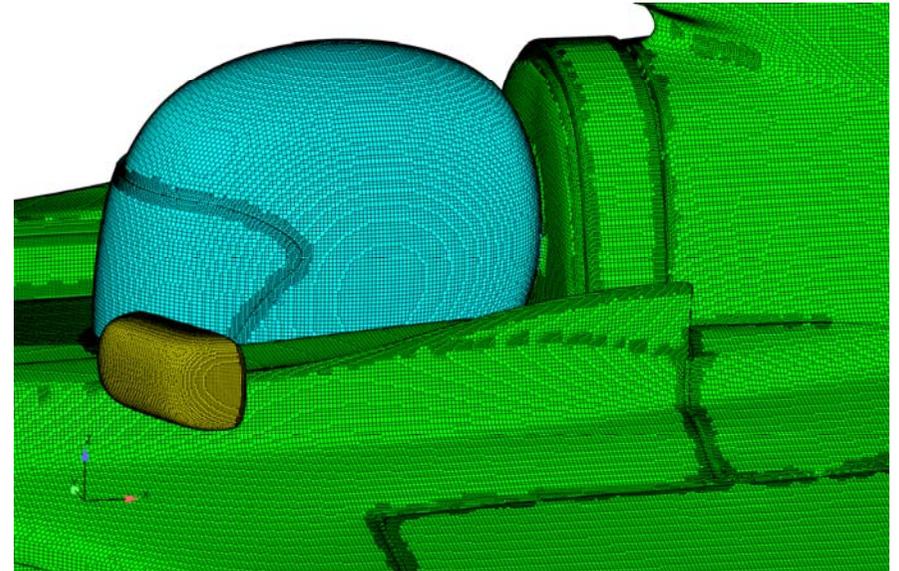
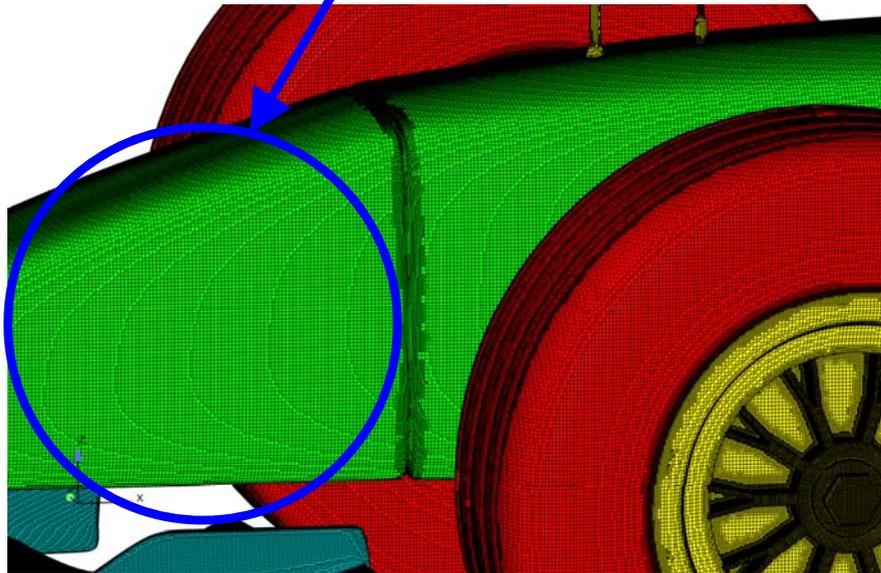


## 6. 解析メッシュ

② Standard Mesh: 約1億メッシュ (102,608,986メッシュ)

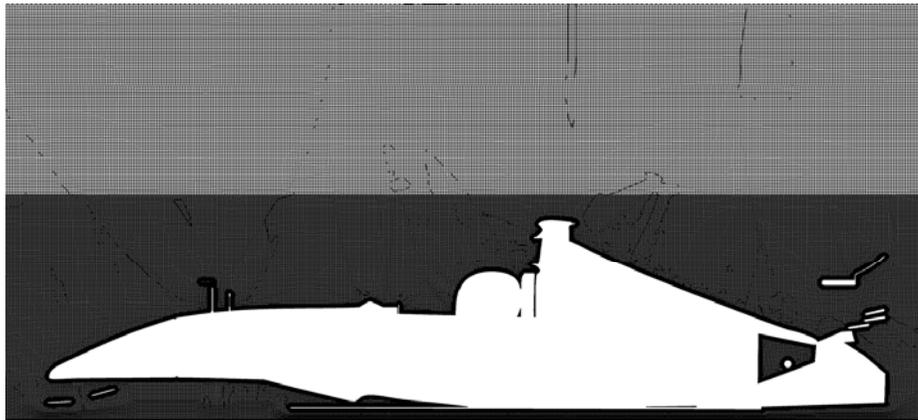


車体表面ベースサイズ:  
約1.6 [mm]

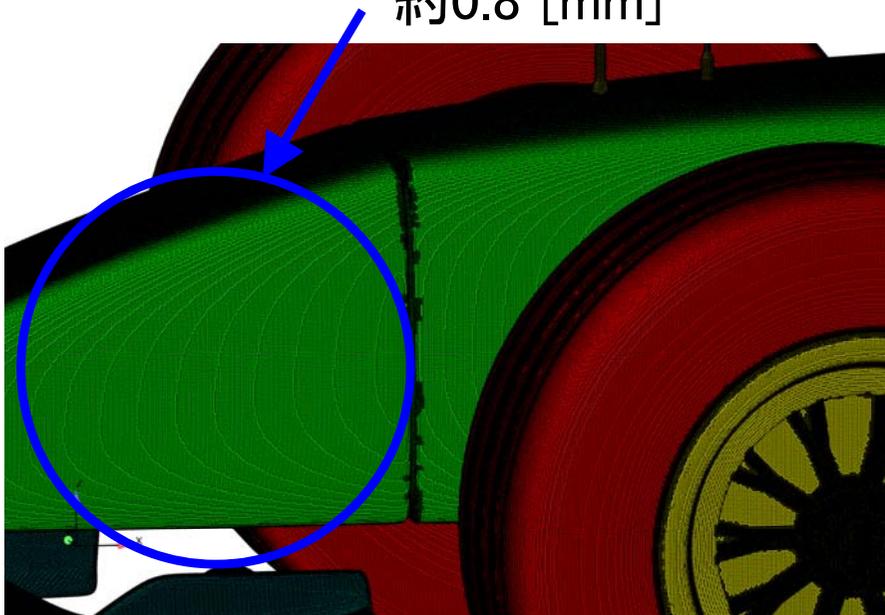
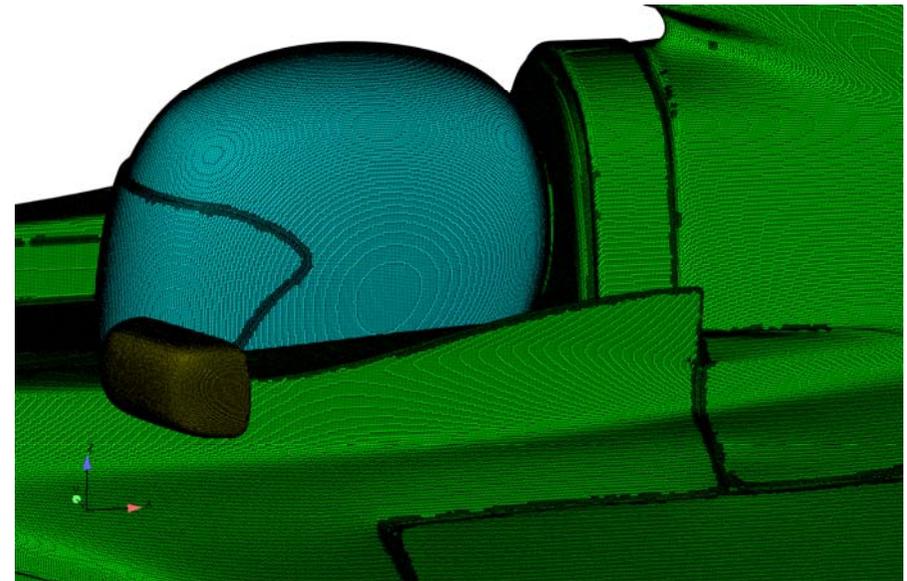


## 6. 解析メッシュ

③ Fine Mesh: 約5億メッシュ (501,384,138メッシュ)

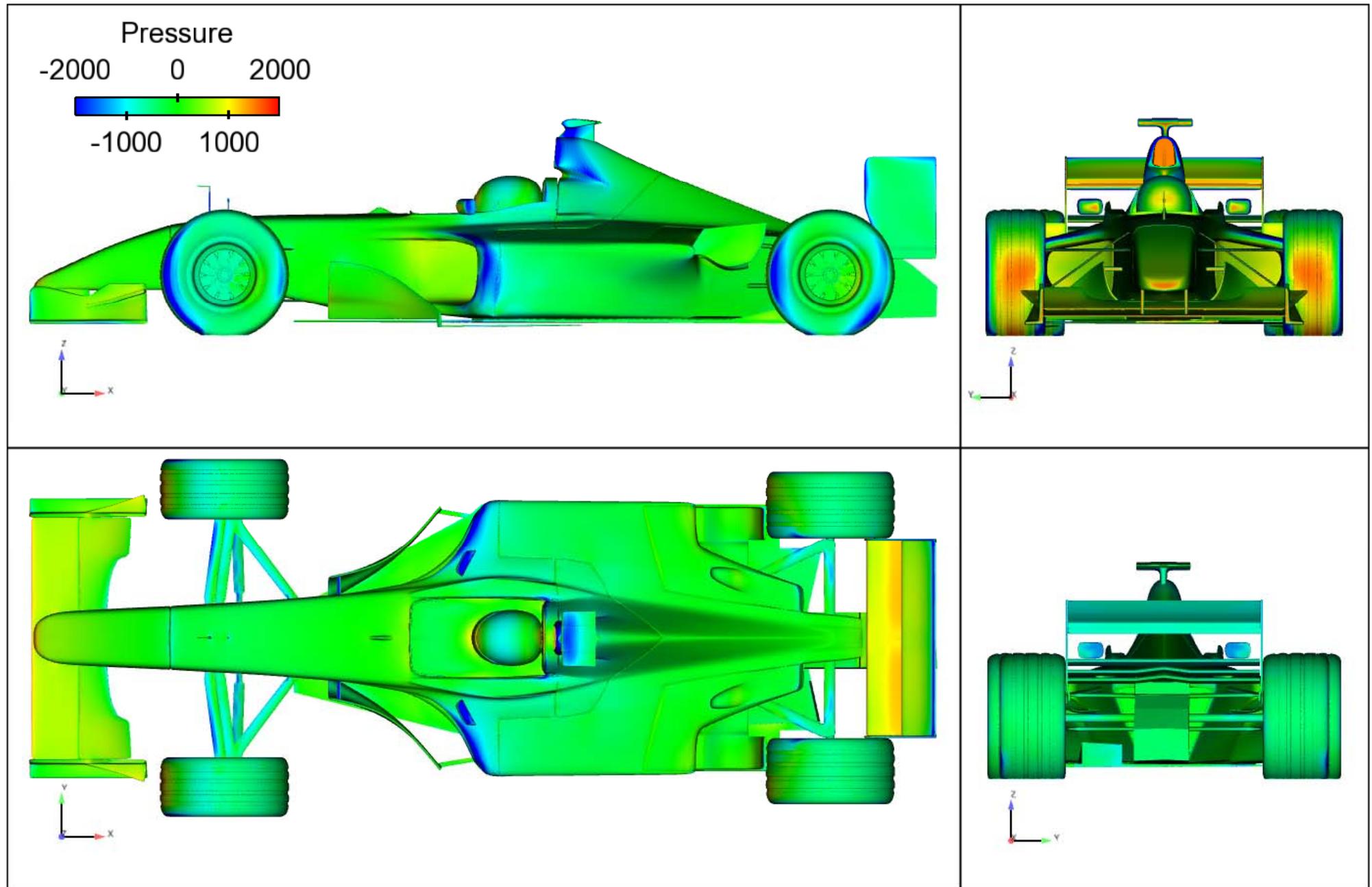


車体表面ベースサイズ:  
約0.8 [mm]



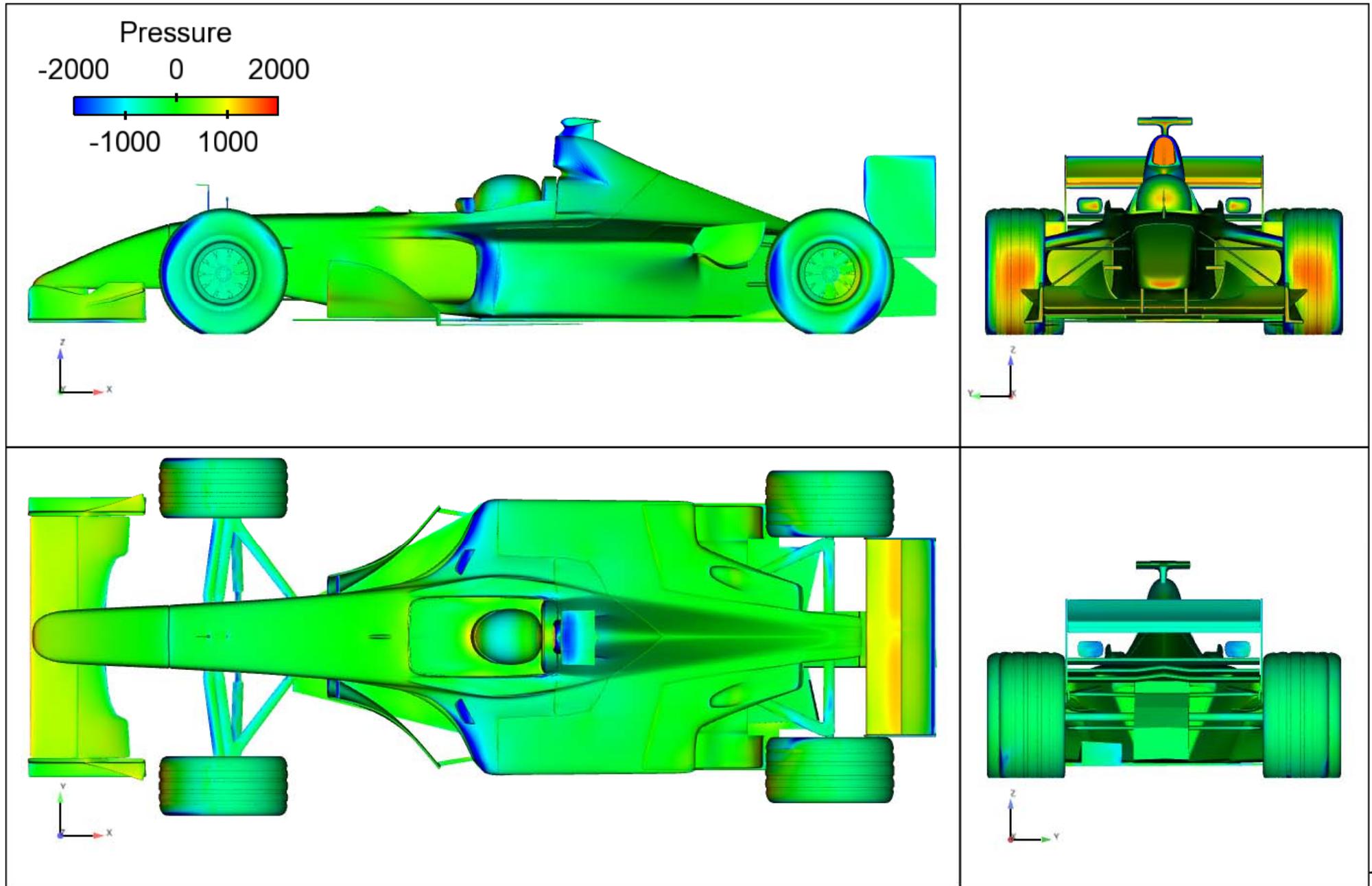
# 7. 解析結果(圧力分布)

① Coarse Mesh: 約2000万メッシュ(20,399,141メッシュ)



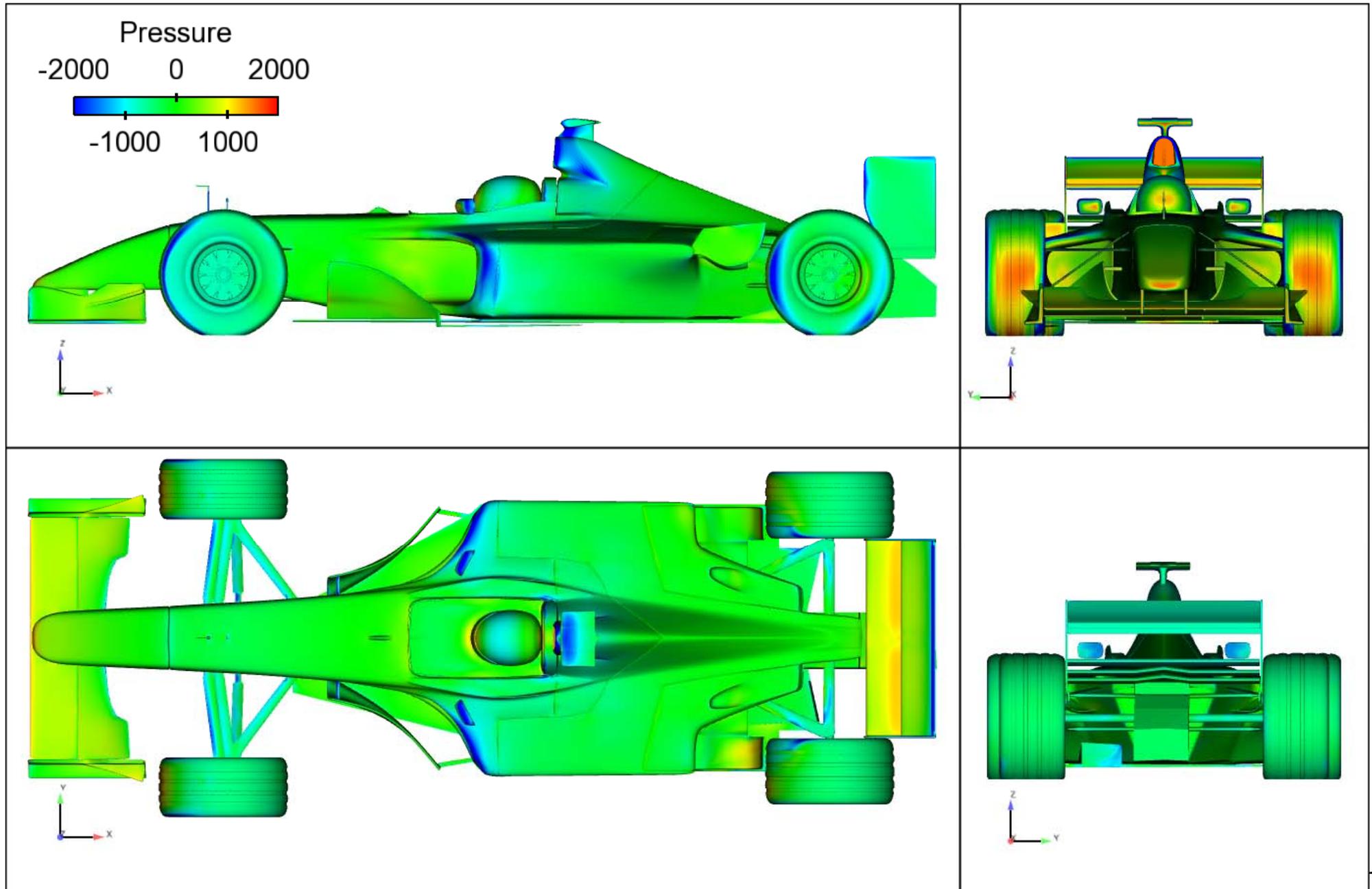
# 7. 解析結果(圧力分布)

② Standard Mesh: 約1億メッシュ(102,608,986メッシュ)



# 7. 解析結果(圧力分布)

③ Fine Mesh: 約5億メッシュ(501,384,138メッシュ)



## 8. ベンチマーク結果(計算時間)

実行開始から500イタレーションまでの平均時間 基準 [単位: sec]

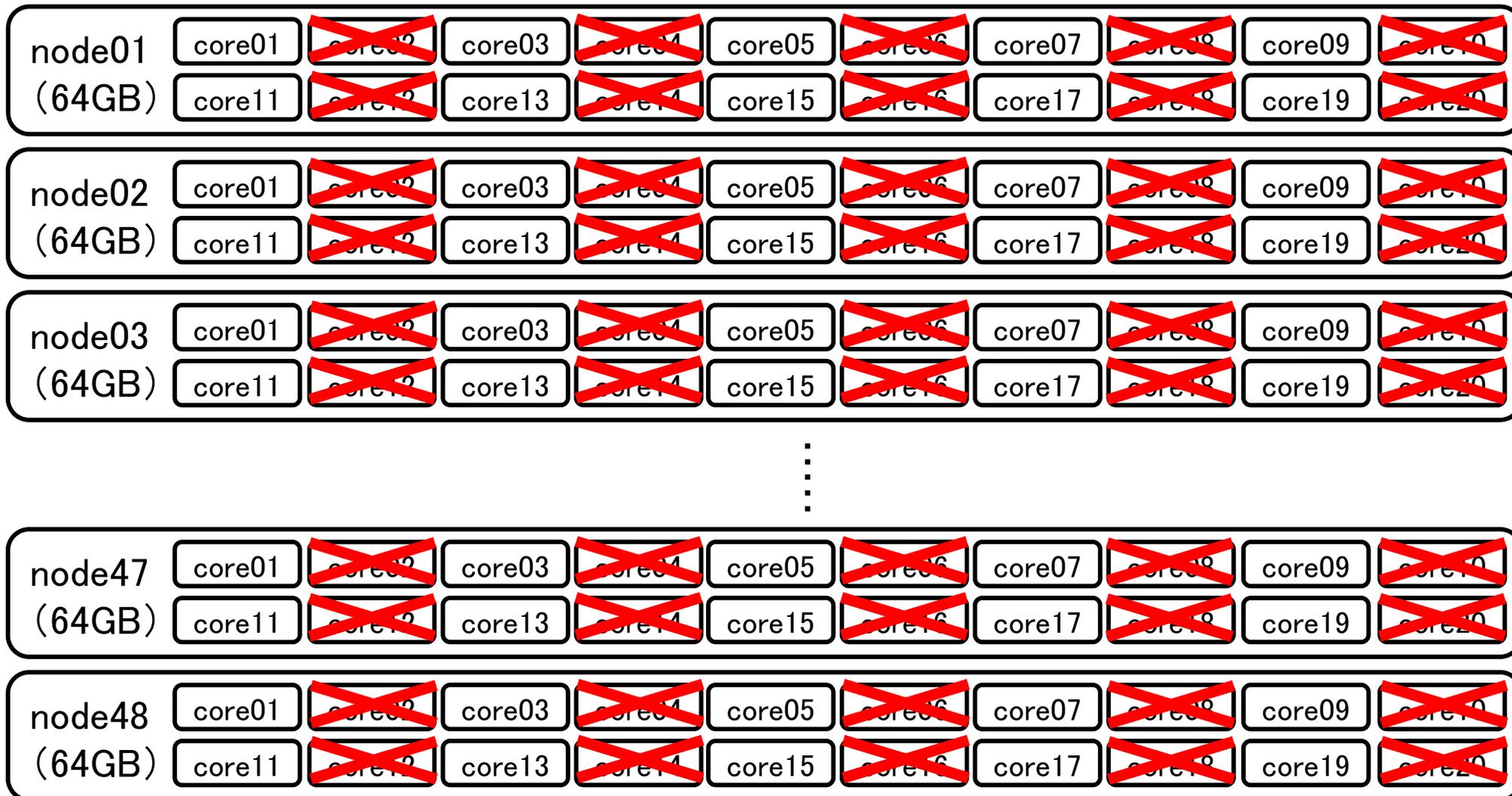
並列数		60	120	240	360	480	720	960	480 (1/2コア)
① Coarse Mesh (約2000万メッシュ)	Aシステム	4510 (0.5)	2335 (1.0)	1156 (2.0)	828.8 (2.8)	1103 (2.1)	1357 (1.7)	858.7 (2.7)	528.2 <u>(4.4)</u>
	Dシステム	2740 (0.5)	1413 (1.0)	720.3 (2.0)	526.5 (2.7)	428.3 (3.3)	413.6 (3.4)	392.9 (3.6)	242.3 <u>(5.8)</u>
② Standard Mesh (約1億メッシュ)	Aシステム		12554 (1.0)	6288 (2.0)	4576 (2.7)	3479 (3.6)	4323 (2.9)	3913 (3.2)	3246 <u>(3.9)</u>
	Dシステム	18801 (0.4)	7442 (1.0)	3770 (2.0)	2747 (2.7)	2130 (3.5)	1645 (4.5)	1538 (4.8)	1153 <u>(6.5)</u>
③ Fine Mesh (約5億メッシュ)	Aシステム		メモリ不足で 実行不可			16352 (1.0)	時間 不足	12619 (1.3)	時間 不足
	Dシステム				12532 (0.8)	9745 (1.0)		5944 (1.6)	

・(カッコ)内は, 120並列(③ Fine Meshについては, 480並列)を基準とした場合の計算速度(スケーラビリティ).

## 8. ベンチマーク結果

480並列については、使用するコア数を半分に減らし、ノード数を倍(960並列と同じ)としたケースも実行した。1コア当たりのメモリ使用量を増加させる狙い。

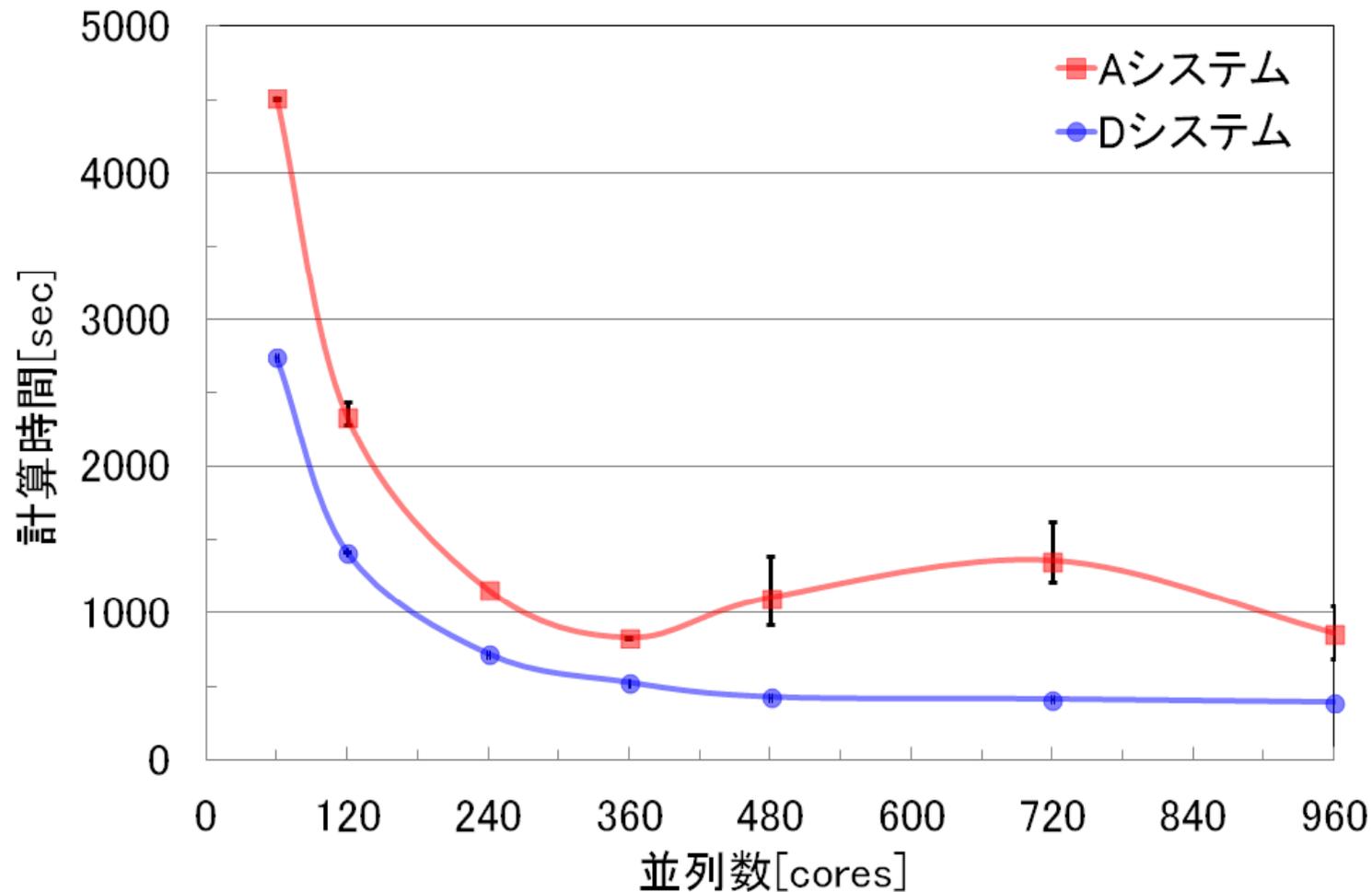
○ Dシステムの場合のイメージ



480並列 (~~20~~→10コア/ノード×48ノード)

## 8. ベンチマーク結果(計算時間)

① Coarse Mesh: 約2000万メッシュ(20,399,141メッシュ)

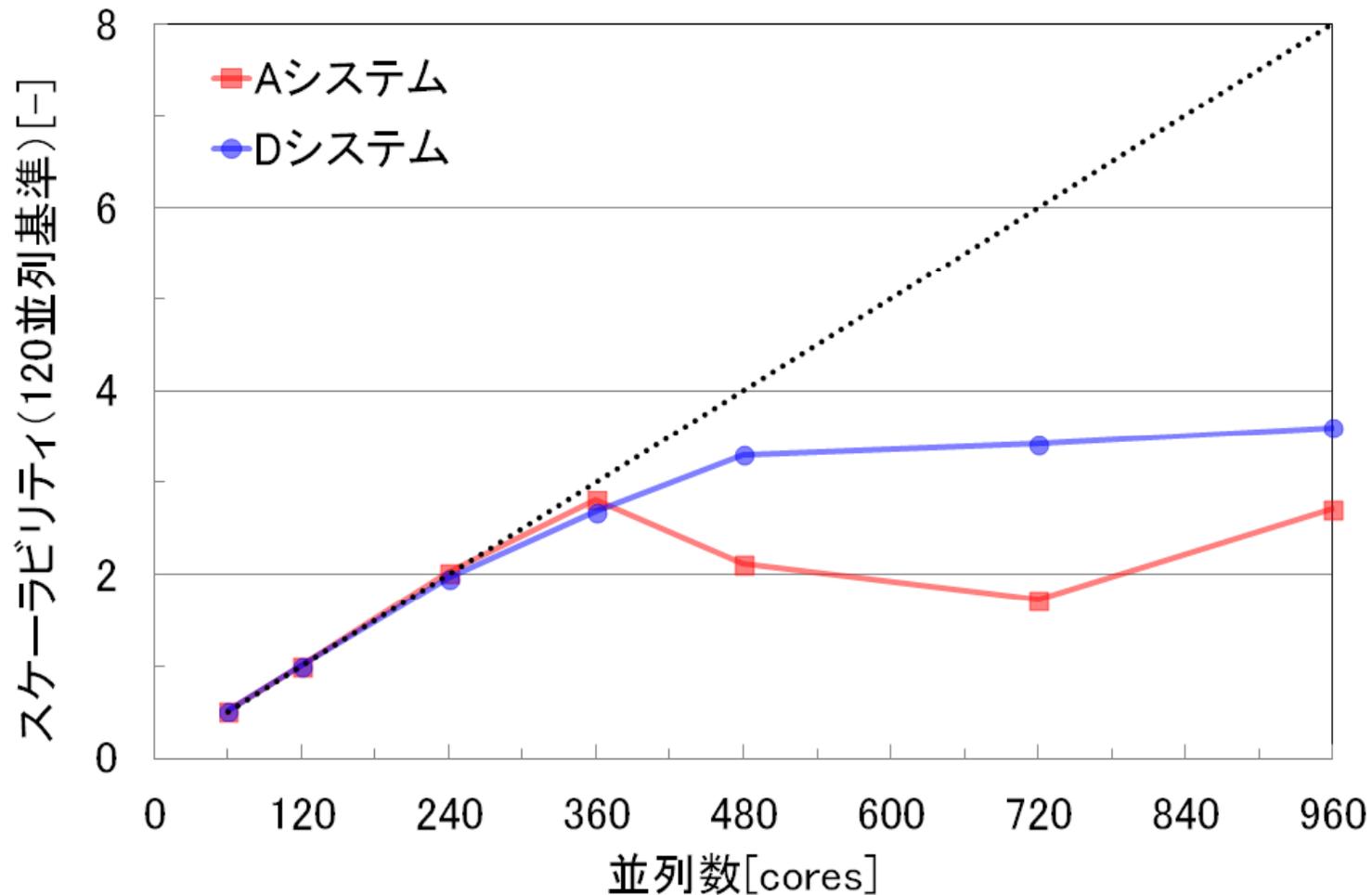


並列数と計算時間の関係

- ・Aシステムでは, 360並列時の計算時間が最も短い. 480並列以上は計算時間も長くなり, バラつき大.
- ・Dシステムでは, 並列数の増加に伴い, 計算時間も減少. バラつきなし.

## 8. ベンチマーク結果 (スケーラビリティ)

① Coarse Mesh: 約2000万メッシュ (20,399,141メッシュ)

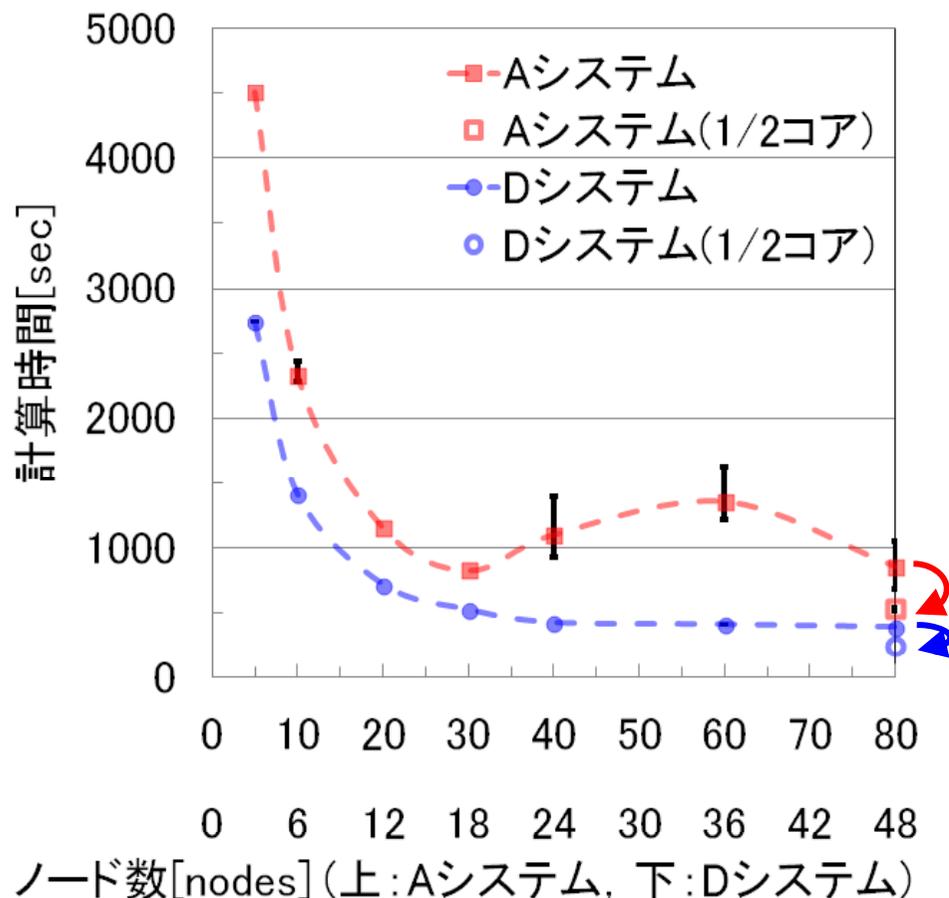


並列数とスケーラビリティ(120並列基準)の関係

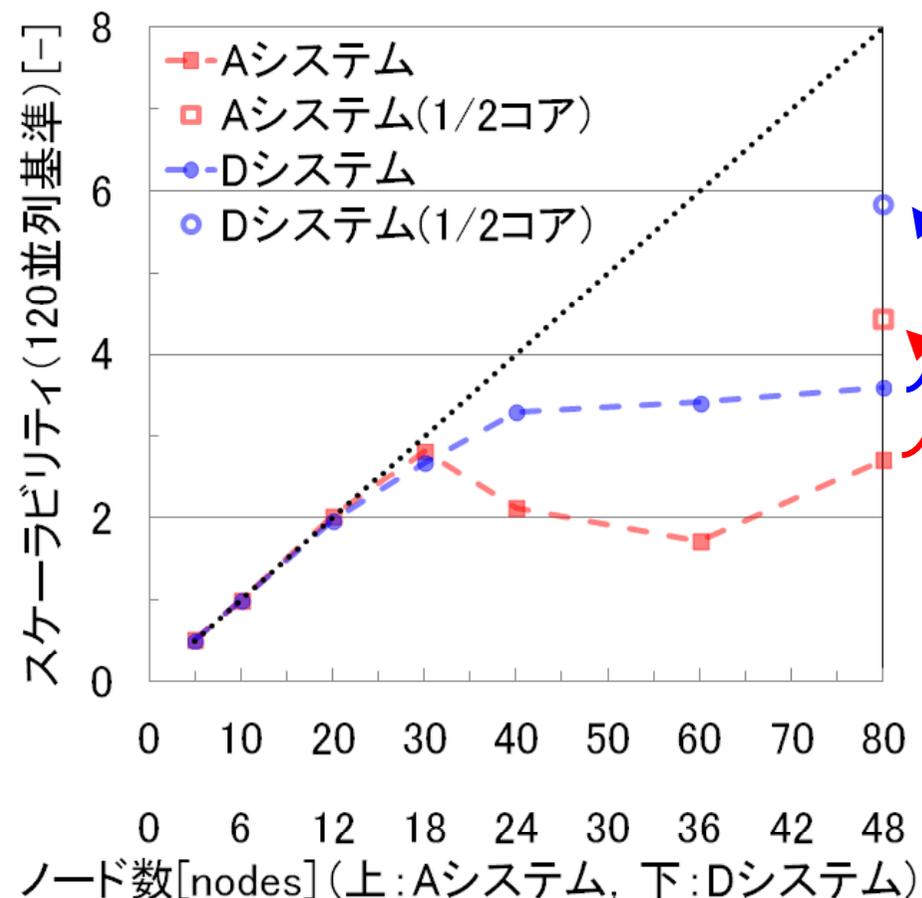
- ・Aシステムでは, 480並列以上でスケーラビリティが低下.
- ・Dシステムでも, 480並列以上ではスケーラビリティの増加は少ない.

## 8. ベンチマーク結果 (計算時間, スケーラビリティ)

① Coarse Mesh: 約2000万メッシュ (20,399,141メッシュ) 1/2コア



ノード数と計算時間  
の関係

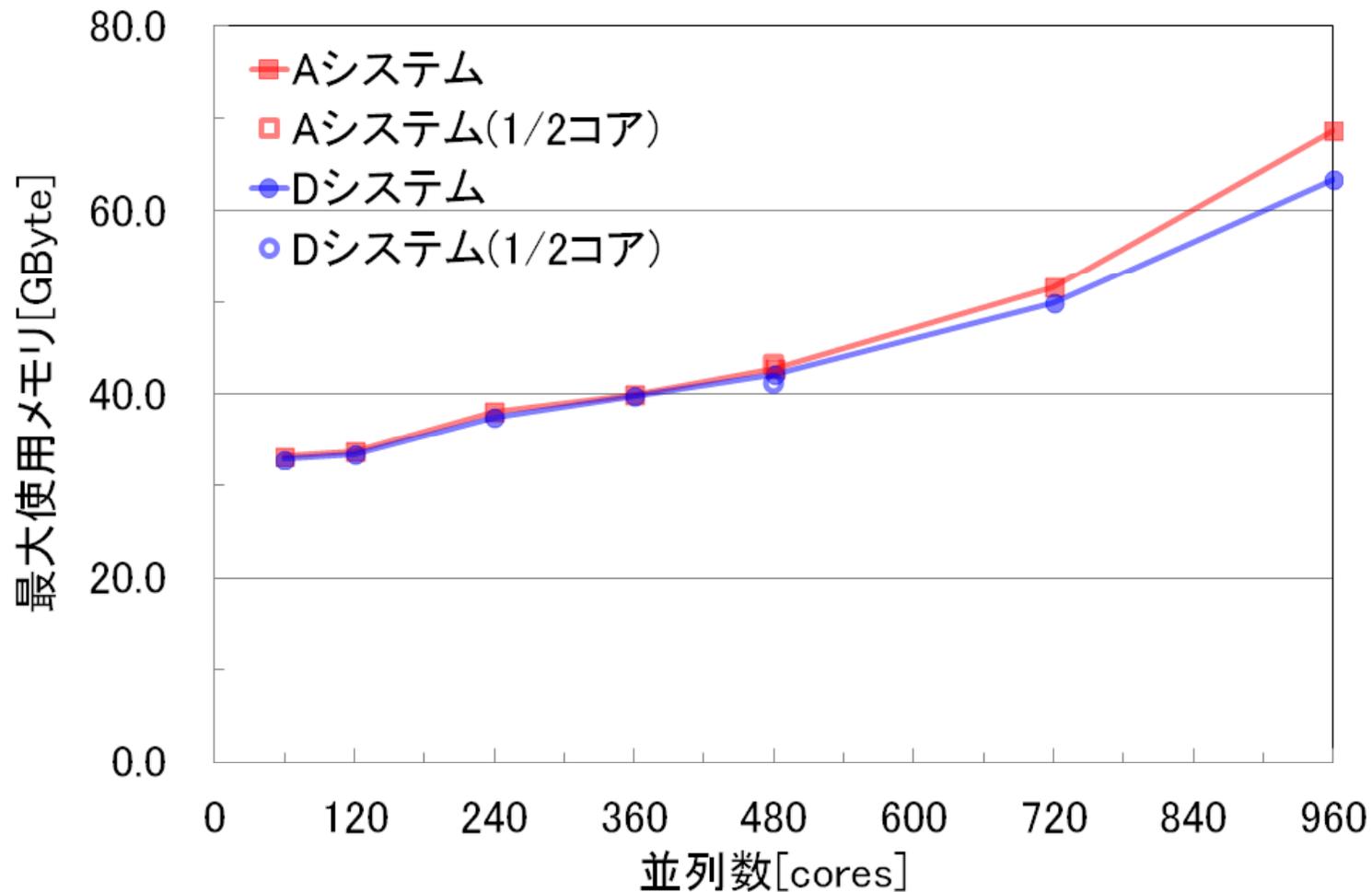


ノード数とスケーラビリティ  
(120並列基準)の関係

- ・同じノード数でも, 1ノード当たりの使用コア数を半分にする  
(Aシステム: 12→6コア, Dシステム: 20→10コア), 計算時間は  
短くなり, スケーラビリティも大幅に上昇する.

## 8. ベンチマーク結果(最大使用メモリ)

① Coarse Mesh: 約2000万メッシュ(20,399,141メッシュ)

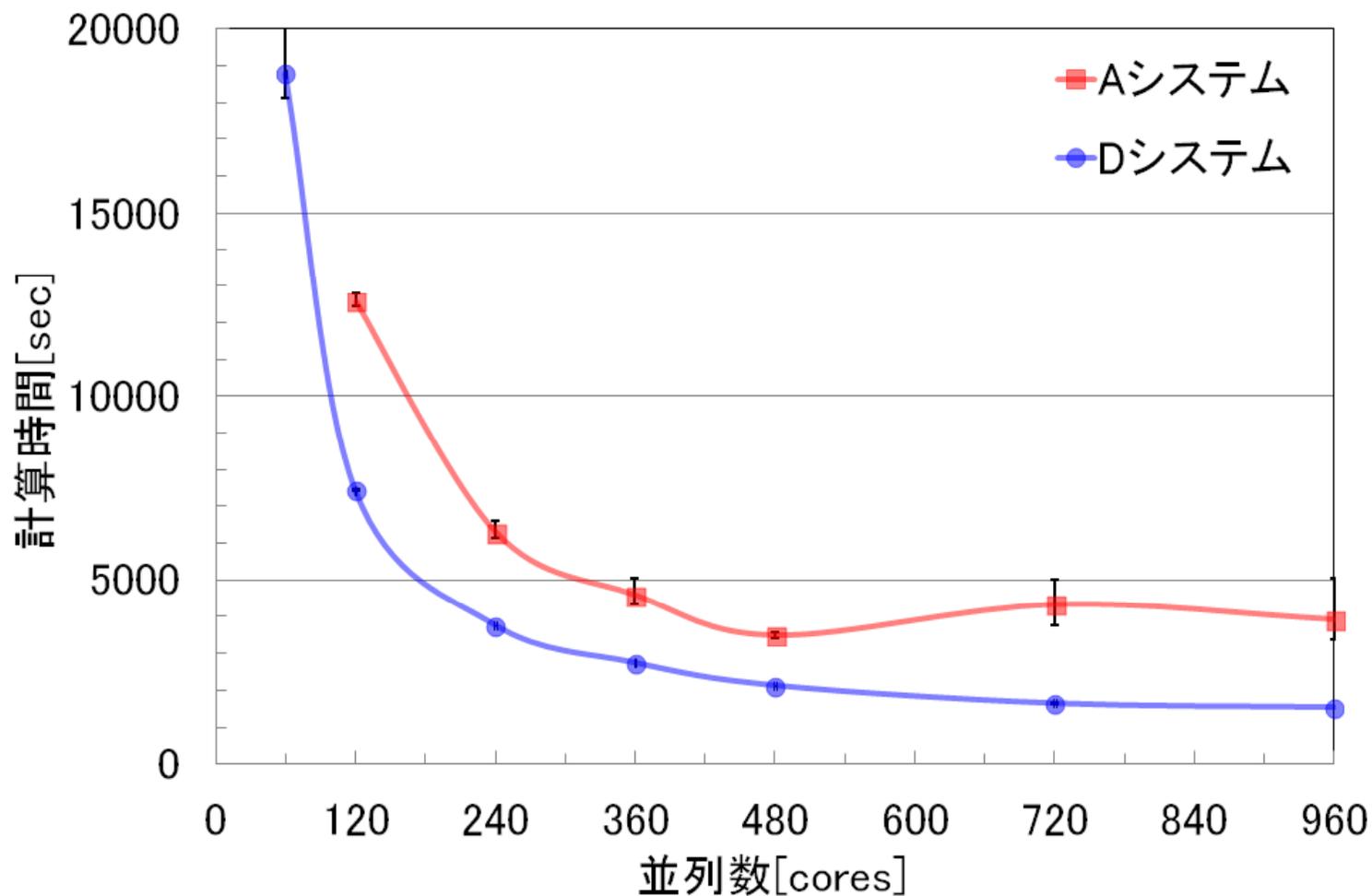


### 並列数と最大使用メモリの関係

- ・Aシステムでは, 120並列時の最大使用メモリ量に比べ, 960並列時の最大使用メモリ量は約2.0倍となった. Dシステムでも同様に, 120並列から960並列では, 約1.9倍の最大使用メモリ量となった.

## 8. ベンチマーク結果(計算時間)

② Standard Mesh: 約1億メッシュ(102,608,986メッシュ)

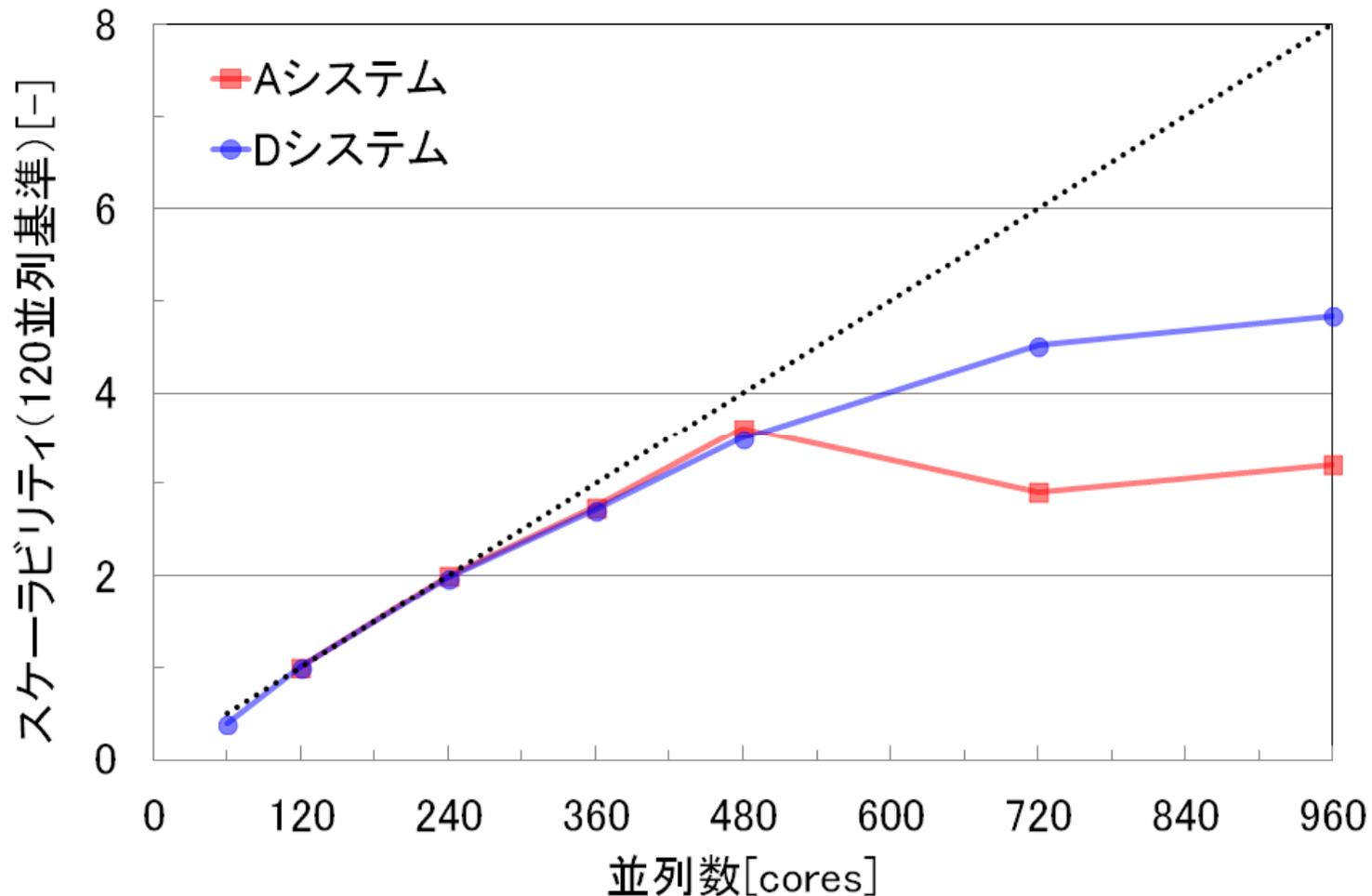


並列数と計算時間の関係

- ・Aシステムでは, 480並列時の計算時間が最も短い.
- ・Dシステムでは, 並列数の増加に伴い, 計算時間も減少. ただし, 60並列時にメモリ不足が原因と考えられるバラつきが発生. その他はバラつきなし.

## 8. ベンチマーク結果 (スケーラビリティ)

② Standard Mesh: 約1億メッシュ (102,608,986メッシュ)

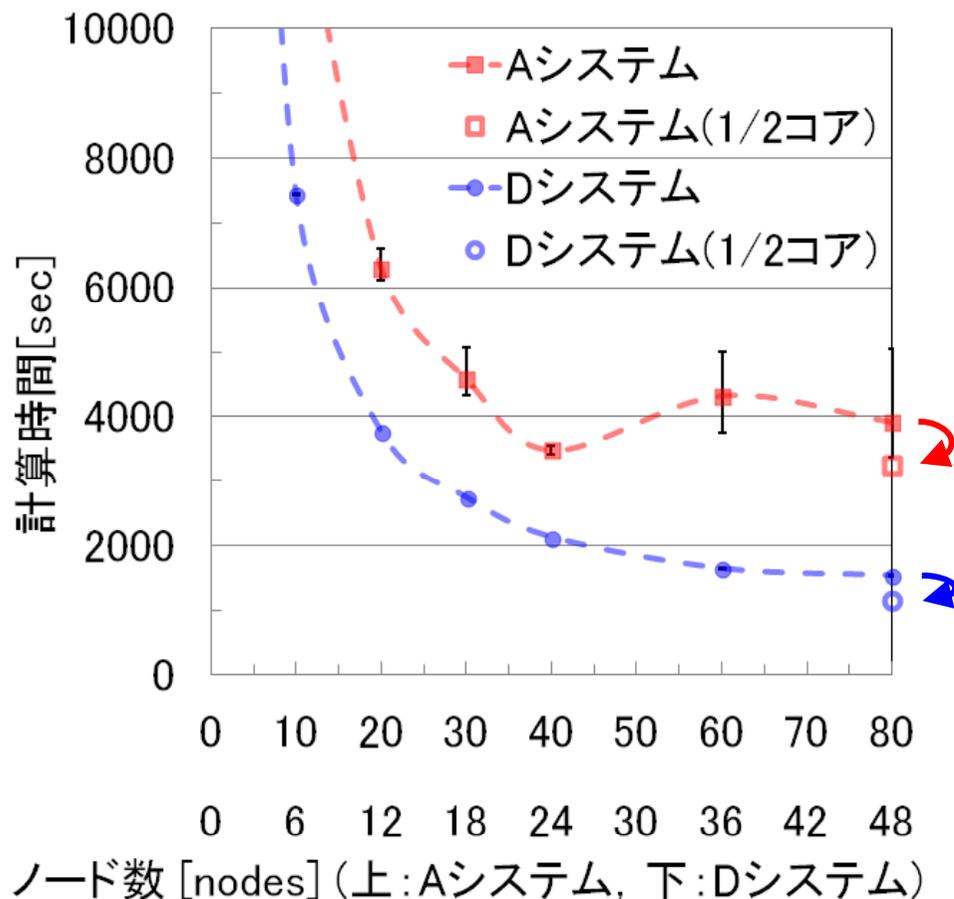


並列数とスケーラビリティ(120並列基準)の関係

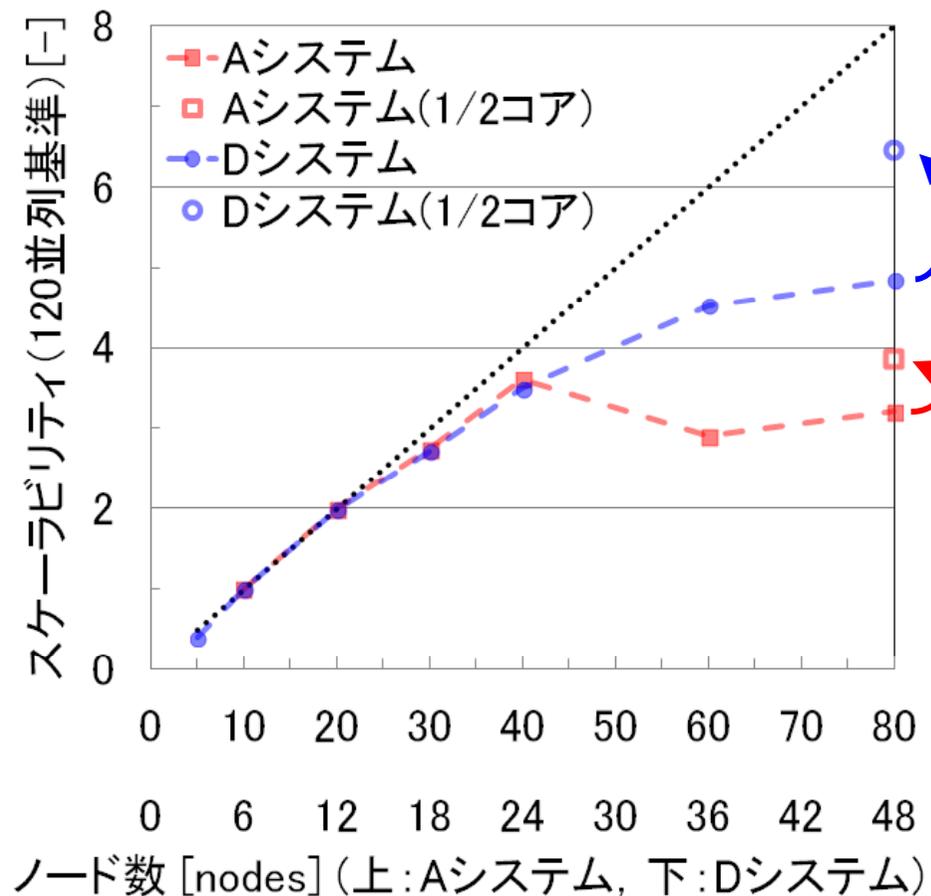
- ・Aシステムでは, 720並列以上でスケーラビリティが低下.
- ・Dシステムでは, 120→960並列 (線形性能8倍) で, スケーラビリティは約4.8 (並列効率約60%).

## 8. ベンチマーク結果 (計算時間, スケーラビリティ)

② Standard Mesh: 約1億メッシュ (102,608,986メッシュ) 1/2コア



ノード数と計算時間  
の関係

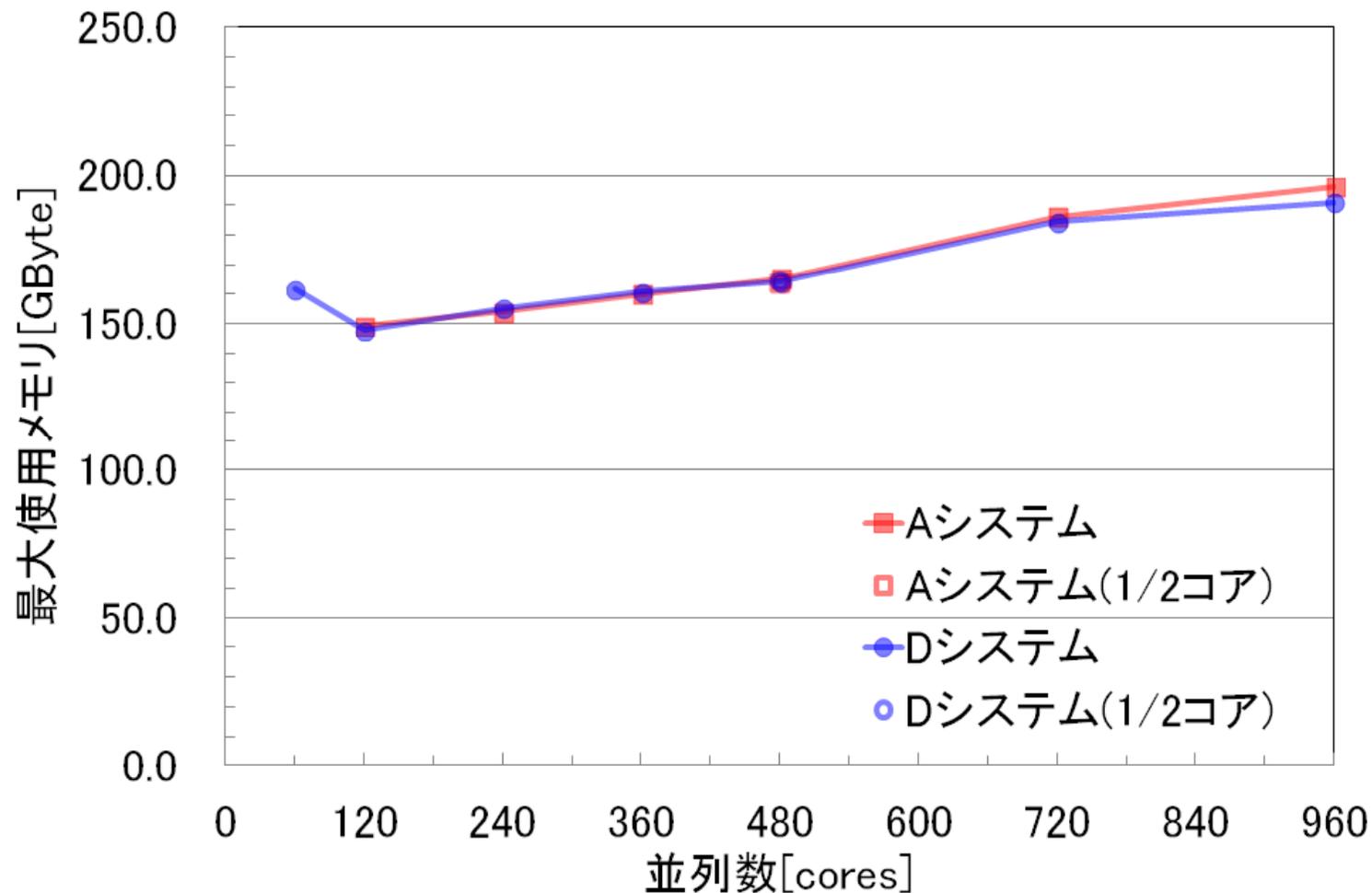


ノード数とスケーラビリティ  
(120並列基準)の関係

- ① Coarse Mesh: 約2000万メッシュと同様, 同じノード数でも, 1ノード当たりの使用コア数を半分にすると, 計算時間は短くなり, スケーラビリティも大幅に上昇する. (960並列時のDシステムの並列効率は約81%.)

## 8. ベンチマーク結果(最大メモリ使用量)

② Standard Mesh: 約1億メッシュ(102,608,986メッシュ)

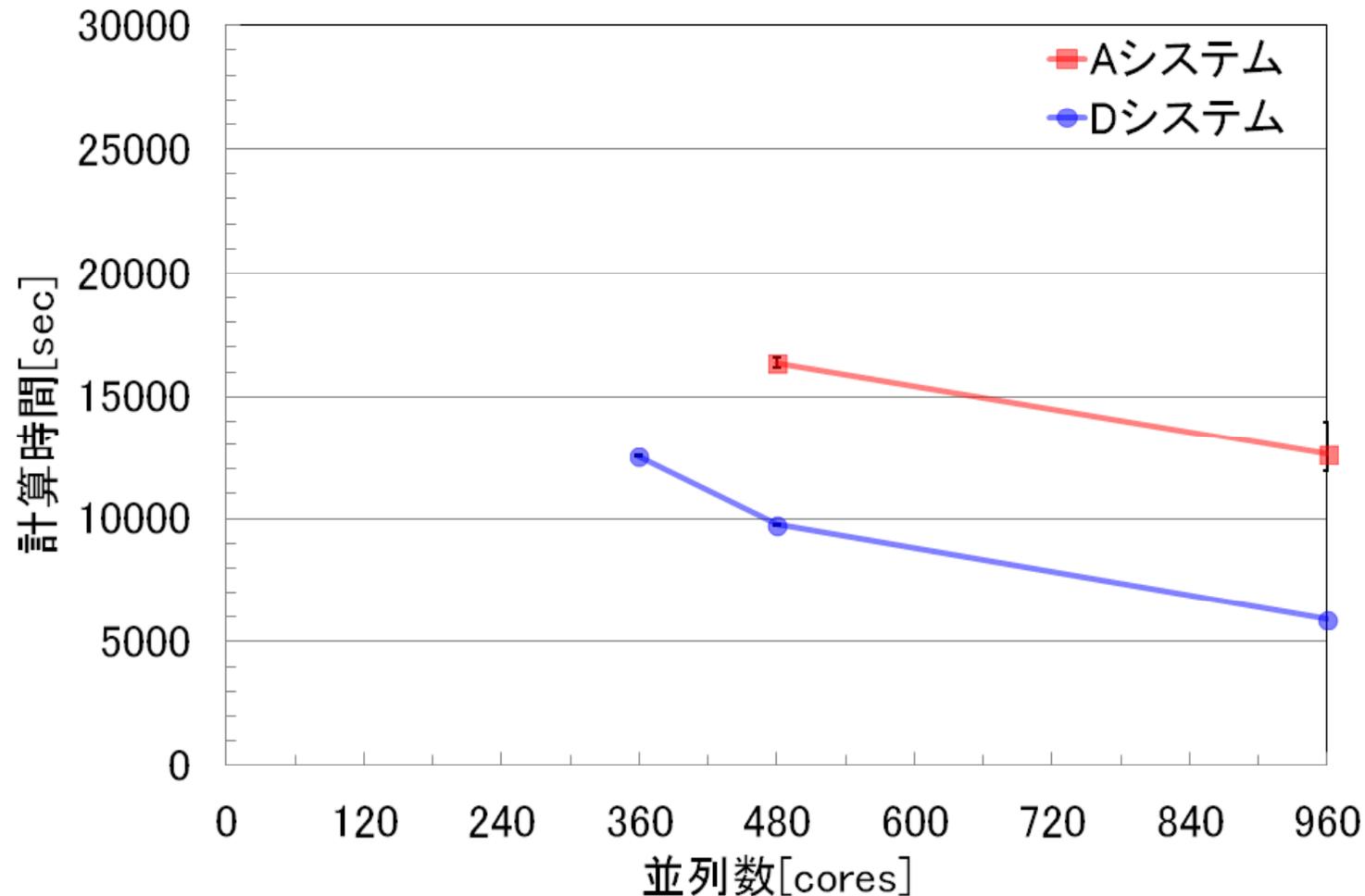


### 並列数と最大使用メモリの関係

- ・Aシステムでは, 120並列時の最大使用メモリ量に比べ, 960並列時の最大使用メモリ量は約1.3倍となった. Dシステムでも同様に, 120並列から960並列では, 約1.3倍の最大使用メモリ量となった.

## 8. ベンチマーク結果(計算時間)

③ Fine Mesh: 約5億メッシュ(501,384,138メッシュ)

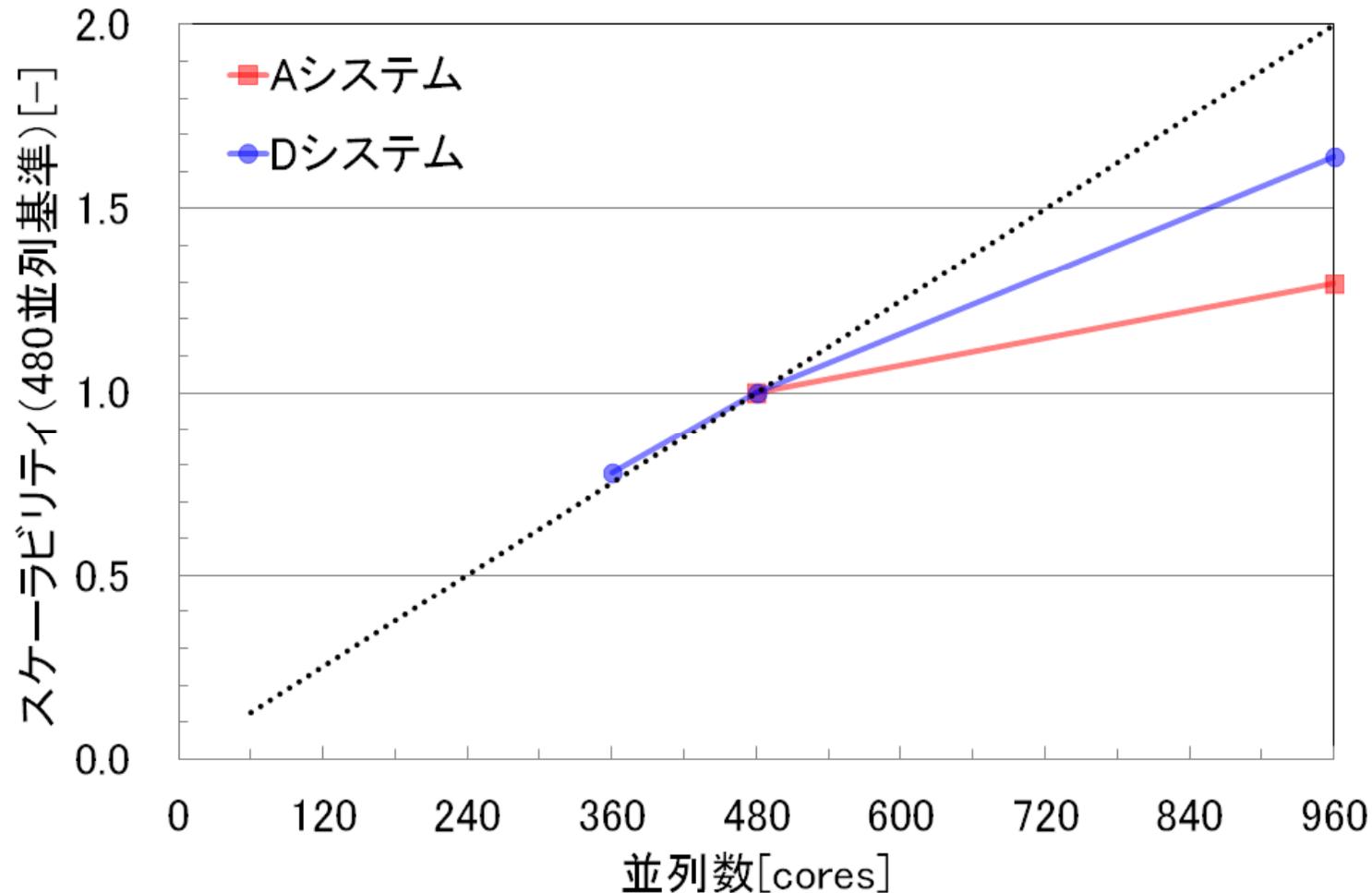


並列数と計算時間の関係

- ・Aシステム, Dシステムとも, 並列数の増加に伴い, 計算時間も減少.
- ・Aシステムでは, 960並列時に少しバラつきが見られた. Dシステムでは, バラつきは見られなかった.

## 8. ベンチマーク結果 (スケーラビリティ)

③ Fine Mesh: 約5億メッシュ (501,384,138メッシュ)

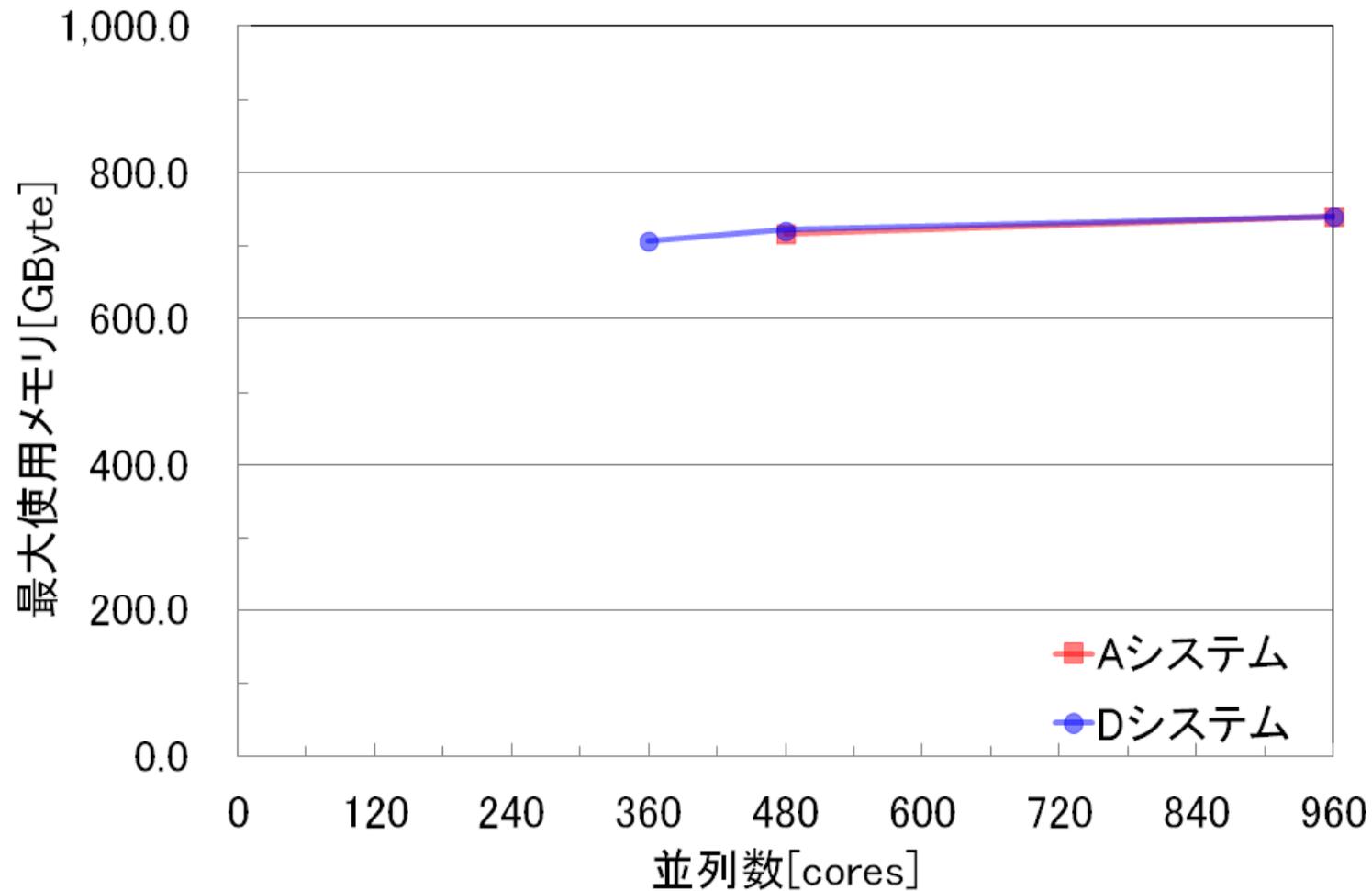


並列数とスケーラビリティ(120並列基準)の関係

- ・Aシステムでは, 480→960並列(線形性能2倍)でスケーラビリティは約1.3(並列効率約65%). Dシステムでは, 同じく480→960並列(線形性能2倍)で, スケーラビリティは約1.6(並列効率約80%).

## 8. ベンチマーク結果(最大メモリ使用量)

③ Fine Mesh: 約5億メッシュ(501,384,138メッシュ)



### 並列数と最大使用メモリの関係

- ・Aシステムでは, 480→960並列時の最大使用メモリ量は約1.03倍となった.
- ・Dシステムでは, 約1.02倍となった. 5億メッシュでは, 並列数の増加による最大使用メモリ量の増加はほとんどない.

## 9. まとめ

FOCUSスパコンのAシステムとDシステムを使用し、オープンソースベースの汎用CFDソフトウェアiconCFDの並列性能ベンチマークテストを実施した。

- FOCUSスパコンのAシステムでは、2000万メッシュと1億メッシュの解析の際に、ある一定の並列数を超えると、計算時間が長くなる傾向が見られた。また、複数回解析した場合のバラつきも大きい。
- FOCUSスパコンのDシステムでは、解析メッシュ数によらず、並列数の増加に伴い、計算時間も短くなった。また、複数回解析した場合のバラつきもほとんど見られなかった。
- FOCUSスパコンのAシステムにおいてもDシステムにおいても、同じノード数の並列計算を実施する場合、1ノード当たりの全コア(12コア[Aシステム], 20コア[Dシステム])を使用するよりも、半分のコア(6コア[Aシステム], 10コア[Dシステム])を使用して計算した方が、計算時間が短い。
- 解析メッシュ数が少ない場合に並列数を増加すると、最大の使用メモリ量は大幅に増加する。

# 謝辞

本ベンチマークは、公益財団法人 計算科学振興財団 (FOCUS) の協力を得て実施した。