

1000億格子規模におけるOpenFOAMの 超大規模数値流体解析

清水建設(株) 技術研究所

○ PHAM VAN PHUC

内山 学

内容

- OpenFOAMの活用例
 - 1千億格子規模の計算への期待
- 1千億格子の計算における取組み
 - OpenFOAMの保有課題・改良策
 - 1千億格子の計算とその性能評価

最近の取組み

「京」における実行効率の向上

Weak Scaling

性能
2倍向上

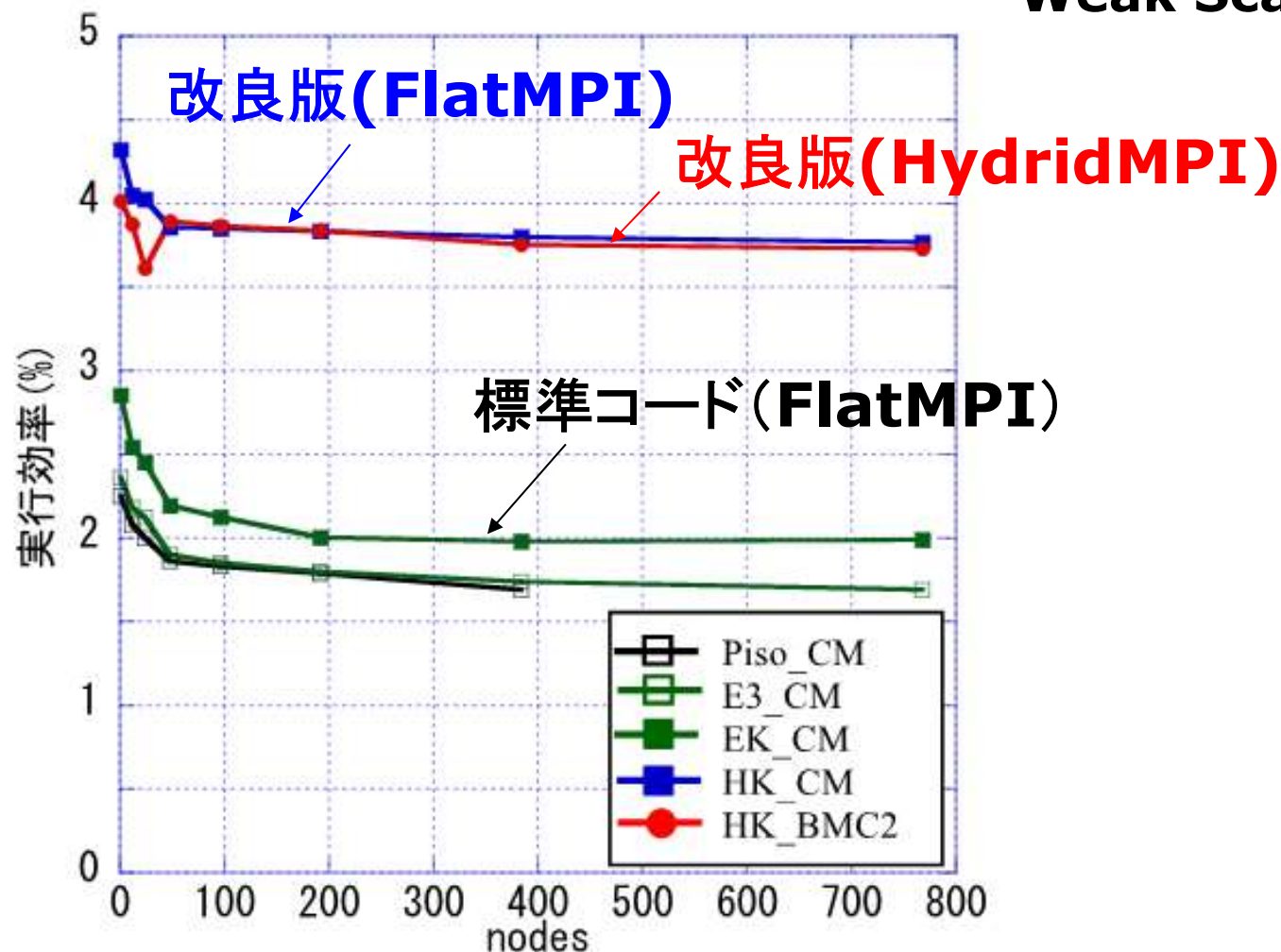


Figure 14 実行効率 vs. ノード数

20億格子

40億格子

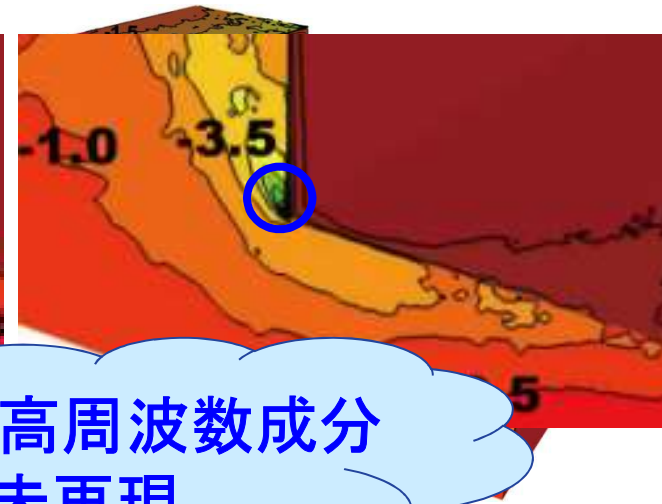
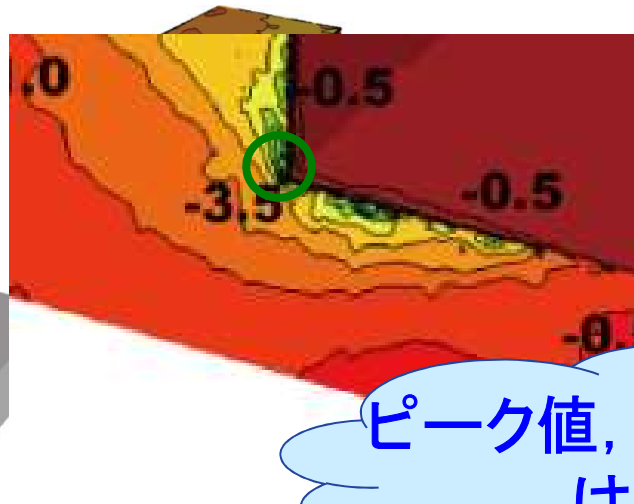
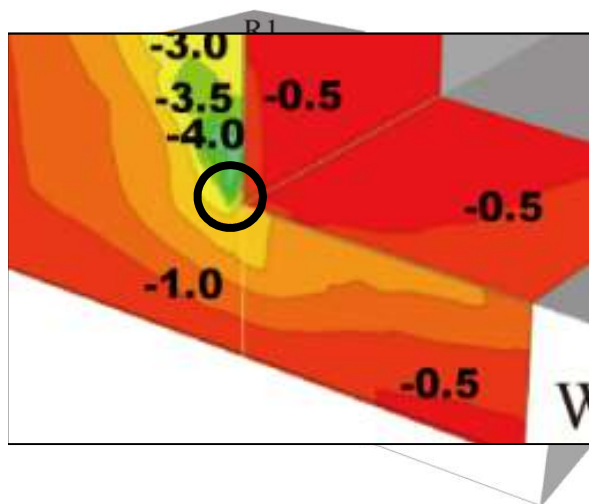
計算規模による精度の変化

建物風圧の予測

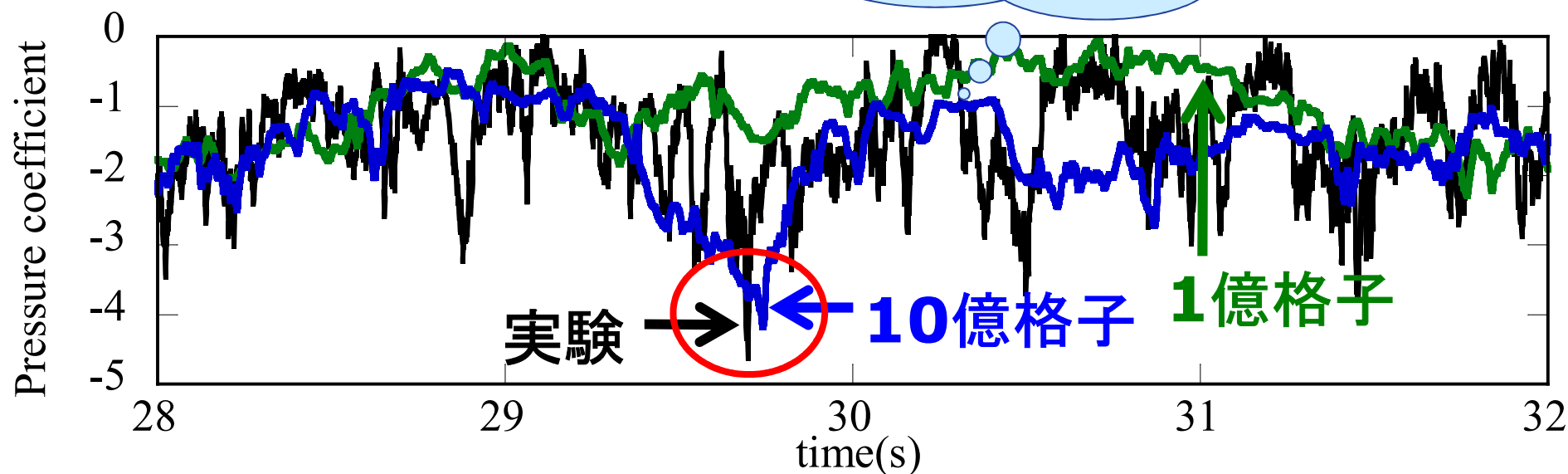
実験

解析(1億格子)

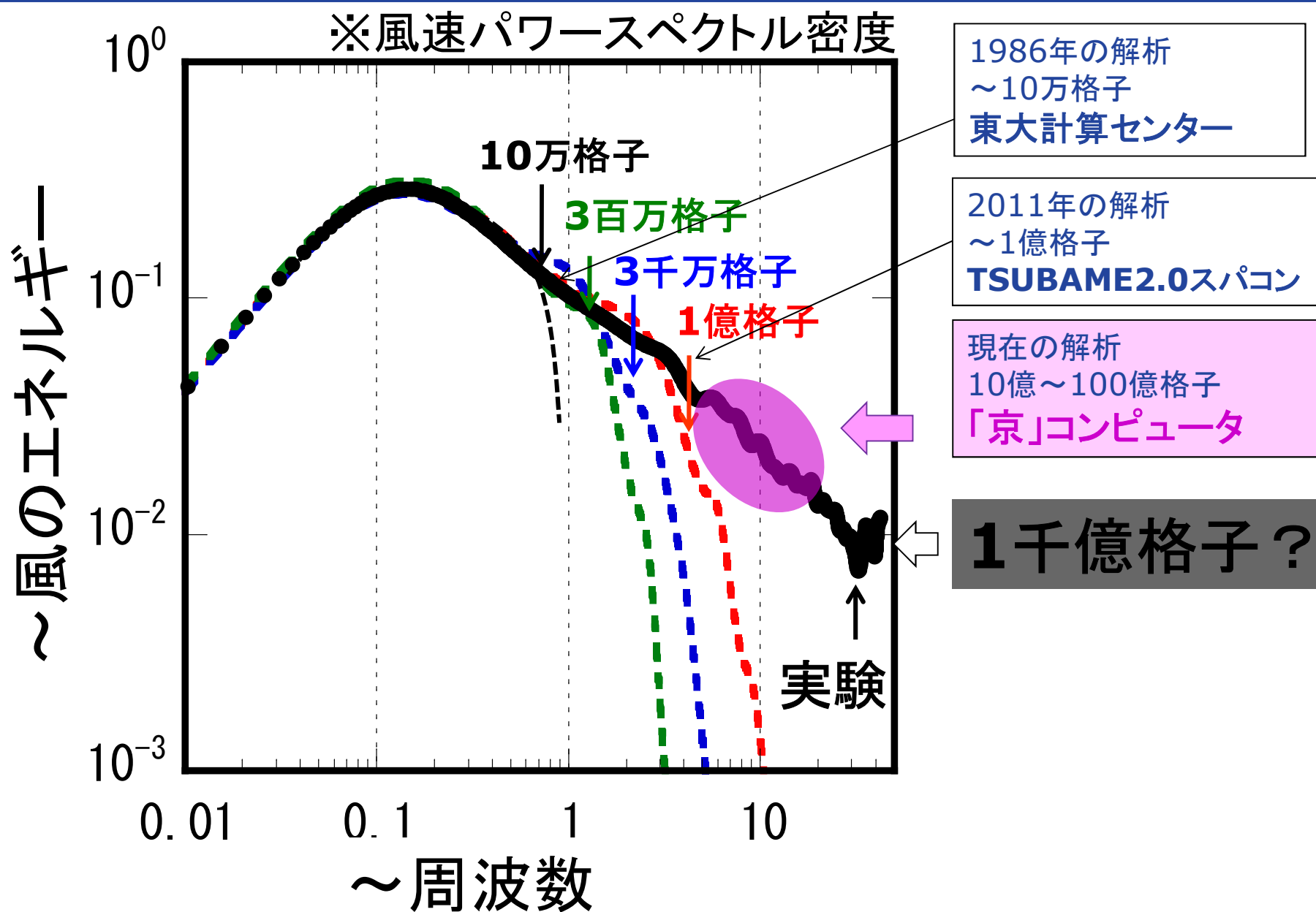
解析(10億格子)



ピーク値, 高周波数成分
は未再現



1千億格子の計算規模への期待



1千億格子規模の計算における課題

① 整数32bitの壁

- 20億以上数値の表現

② 解析モデル作成の壁

- 10億格子以上のモデルの作成

③ 超並列処理の壁

- 1万並列以上の計算

オープンCAE
シンポジウム2013
講演資料

① 整数32bitの壁

20億以上数値の表現

整数型一覧

符号	基本型	limits.h	保証値
signed	char	SCHAR_MIN	-127
		SCHAR_MAX	+127
unsigned		UCHAR_MAX	255
signed	short int	SHRT_MIN	-32,767
		SHRT_MAX	+32,767
unsigned		USHRT_MAX	65,535
signed	int	INT_MIN	-32,767
		INT_MAX	+32,767
unsigned		UINT_MAX	65,535
signed	long int	LONG_MIN	-2,147,483,647
		LONG_MAX	+2,147,483,647
unsigned		ULONG_MAX	4,294,967,295
signed	long long int	LLONG_MIN	-9,223,372,036,854,775,807
		LLONG_MAX	+9,223,372,036,854,775,807
unsigned		ULLONG_MAX	18,446,744,073,709,551,615

OpenFOAMコード

整数64bit等の対応
へ改良

```
Found 0 zoned faces to keep together.  
Refined mesh: cells:801600009 faces:-1880129018 points:811616410  
Cells per refinement level:  
0 700537152  
1 87924507  
2 13138350  
Writing mesh to time 2  
Wrote mesh in = 1023.69 s.  
Mesh refined in = 3518.65 s.  
Finished meshing in = 3841.13 s.  
End  
Finalising parallel run  
-----postsim-----
```

面の数(あり得ない):
-1,880,129,018
計算の誤り!

20億まで
900兆まで

コードの修正
(整数64bit対応)

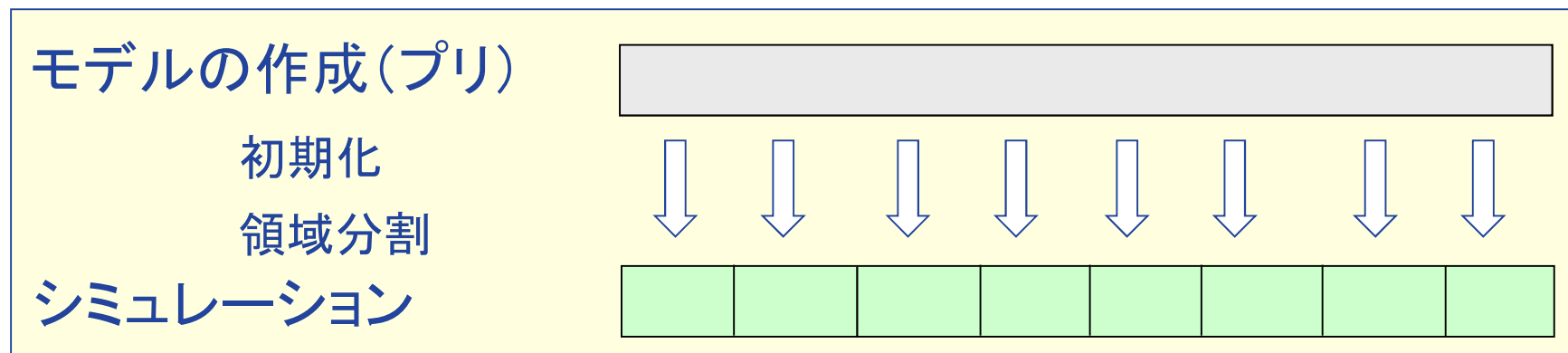
```
Determining initial surface intersections  
-----  
Edge intersection testing:  
Number of edges : 17742344704  
Number of edges to retest : 17742344704  
Number of intersected edges : 52608  
Calculated surface intersections in = 478.26 s  
Initial mesh: cells:5908037632 faces:17742344704 points:5926248121  
Cells per refinement level:  
0 5908037632  
Adding patches to surface calculation  
-----  
Patch Type Region  
-----  
wall_blk03:
```

面の数: 17,742,344,704 ~ 177億
格子数: 5,908,037,632 ~ 59億
計算は正しい

② 解析モデル作成の壁

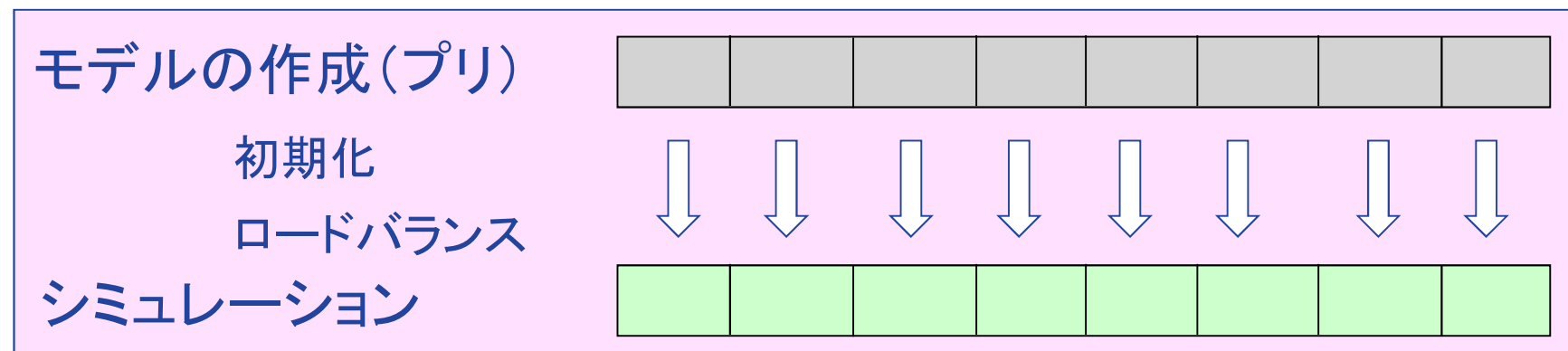
■ 従来の方法

■ シリアル処理 (メモリ不足, 1TB~10億格子限界)



■ 新たな方法

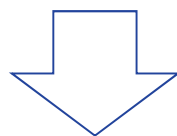
■ 分散処理 (既存SnappyHexMesh等の工夫で可能)



③ 超並列処理の壁

必要な並列数

- 通常計算: 10万 ~ 100万格子/並列



1千億格子

- 並列数: 10万 ~ 100万

※「京」コンピュータ

- 計算ノード : 82,944ノード (8コア/ノード)
- 処理可能並列数 : 最大 165,888 (MPI)

「京」には、1千億格子の計算リソースがある。
OpenFOAMには、10万並列処理の実績がない。

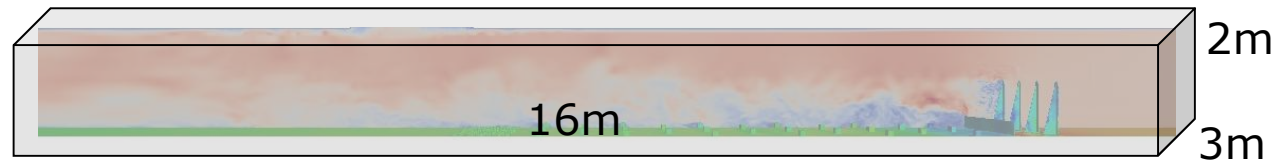
OpenFOAMの並列性能の評価

■ 解析手法

- コード : OpenFOAM-2.2.x
- ソルバ : pisoFoam
- 数値解法 : U:BiCG, P:PCG

■ 解析モデル

風洞測定部

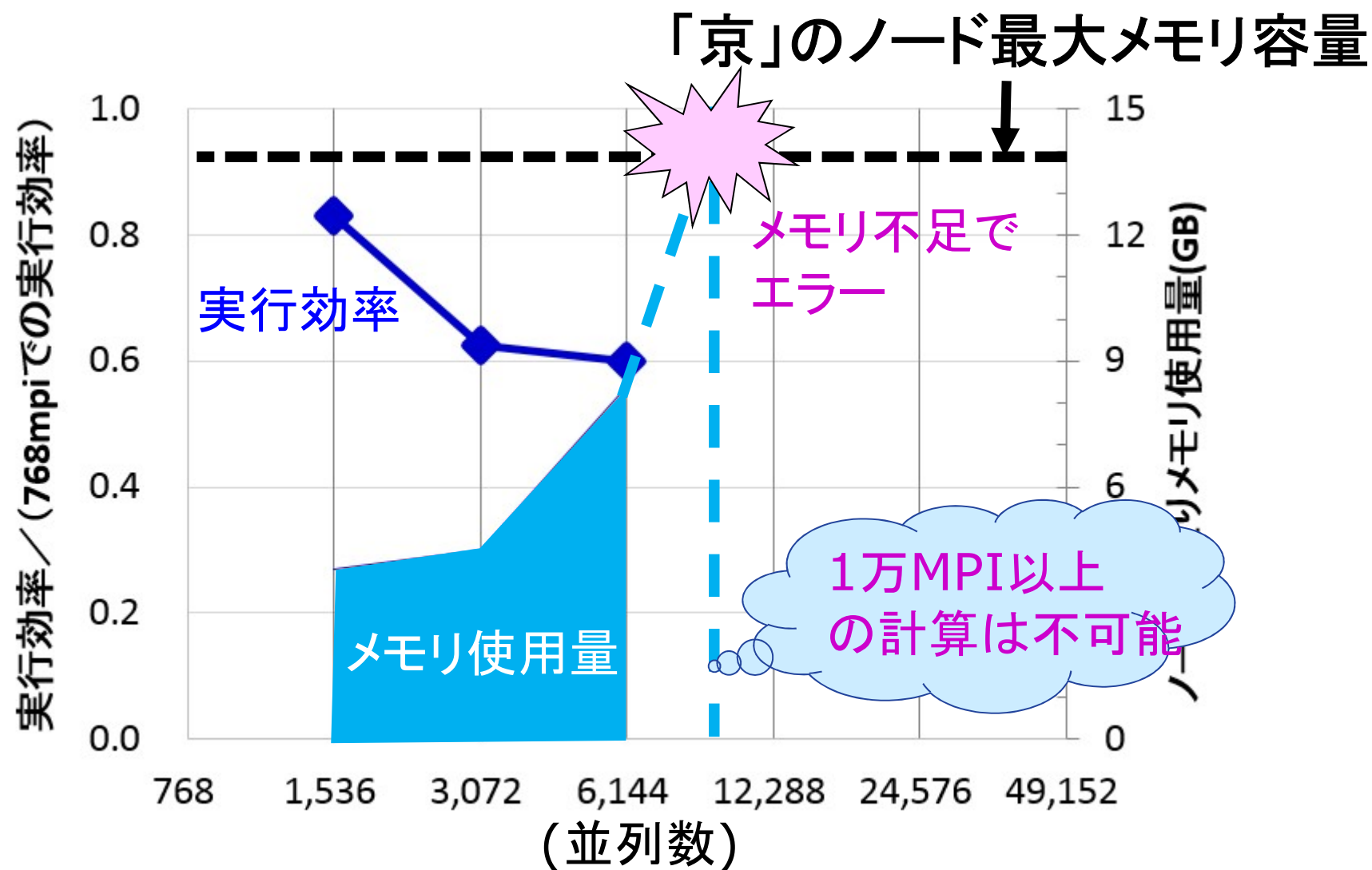


■ 実行環境

- 計算機 : 「京」コンピュータ
- ネットワーク: Tofuインターコネクト

実行効率・使用メモリ量

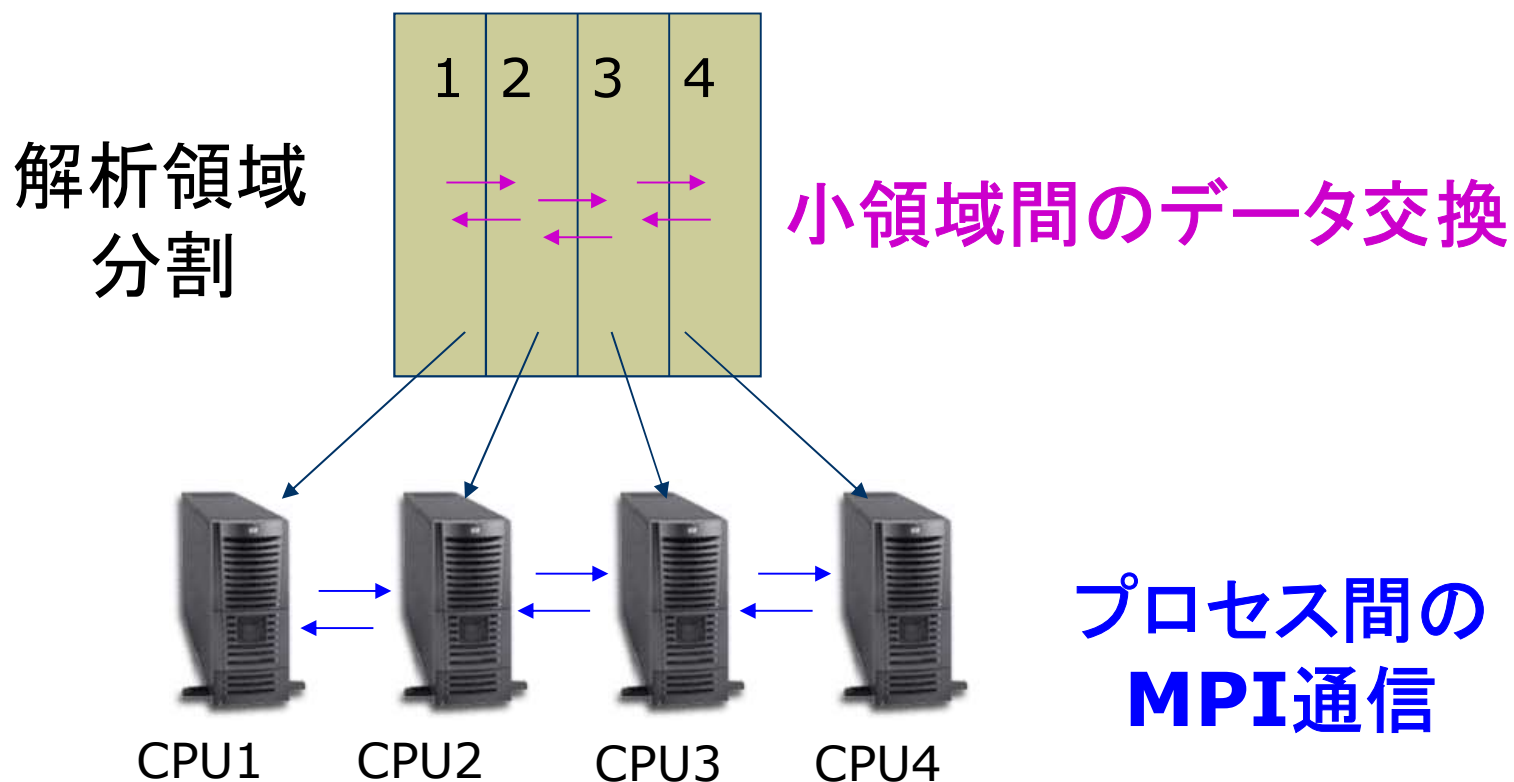
Weak Scaling



実行効率は急低下、使用メモリ量は爆発的増加

OpenFOAMの並列処理プラットフォーム

■ 領域分割手法

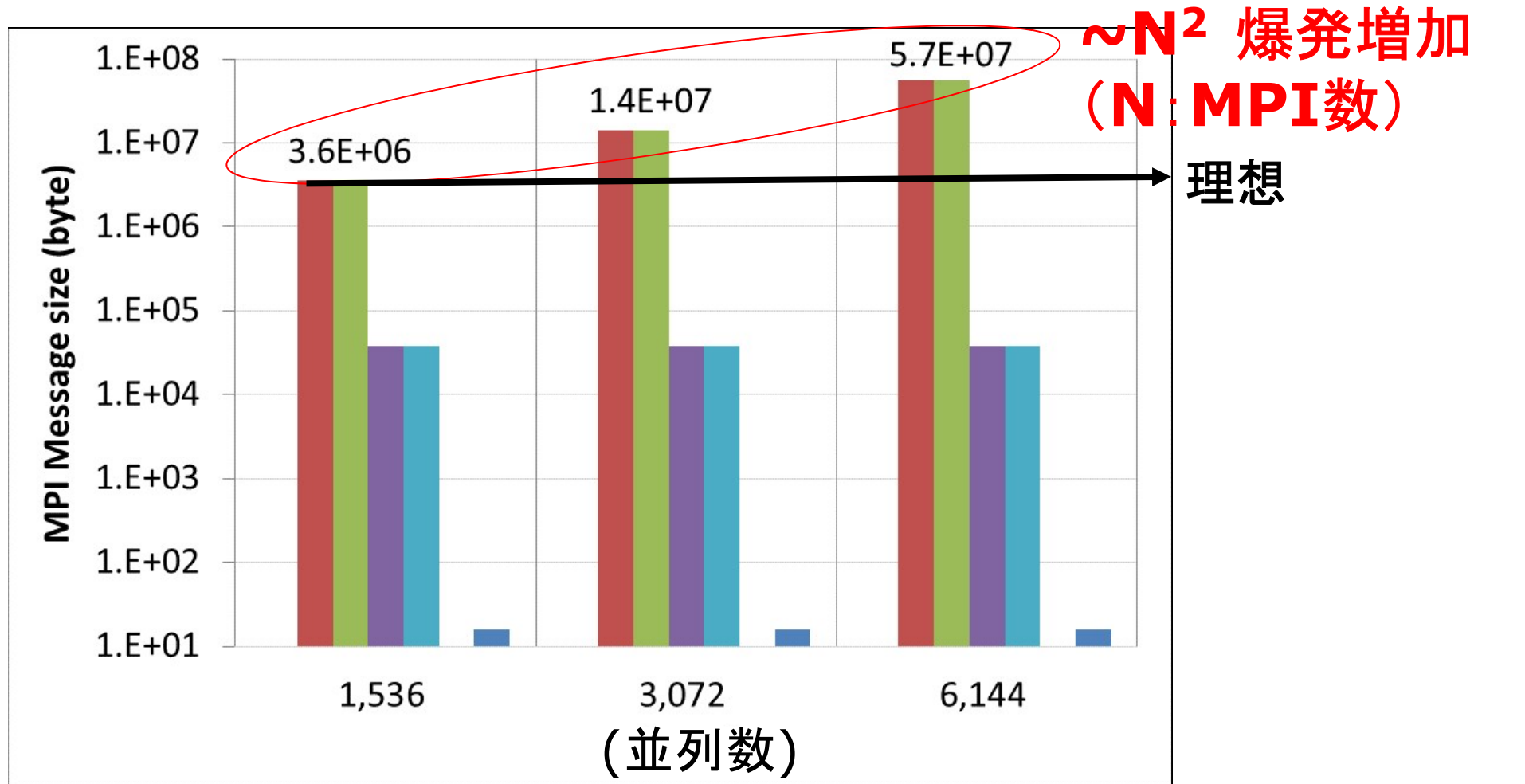


MPI通信の測定結果

Weak Scaling

標準版

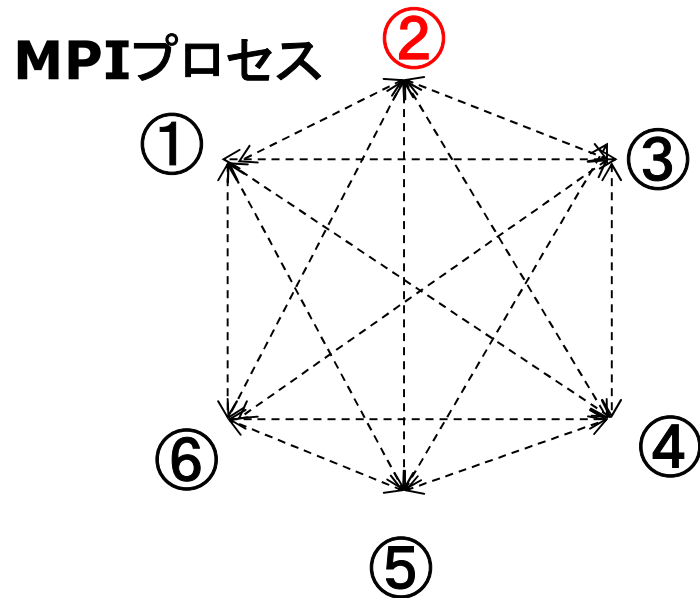
■ Probe ■ Send ■ Recv ■ Isend ■ Irecv ■ Waitall ■ Allreduce



一部のMPIサイズは爆発的に増大

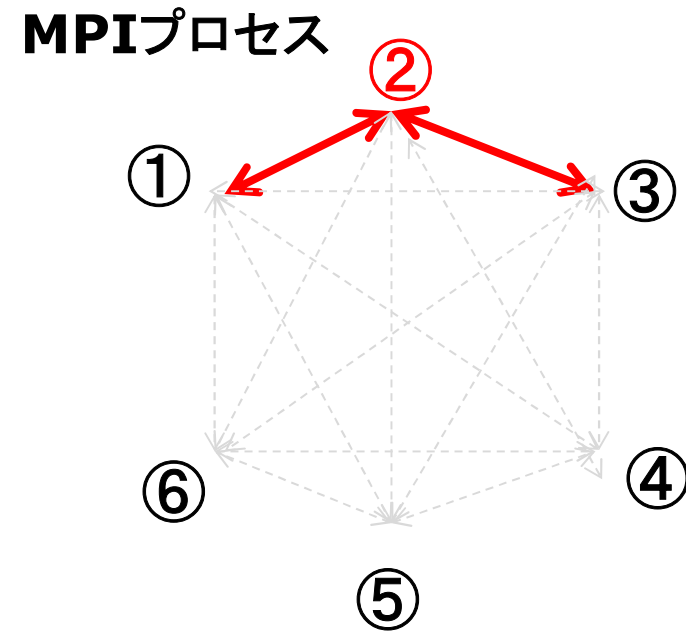
MPI通信形態の改良

exchange関数



全体MPI通信形態
となっているコード

標準版



隣接MPI通信形態
になるコード

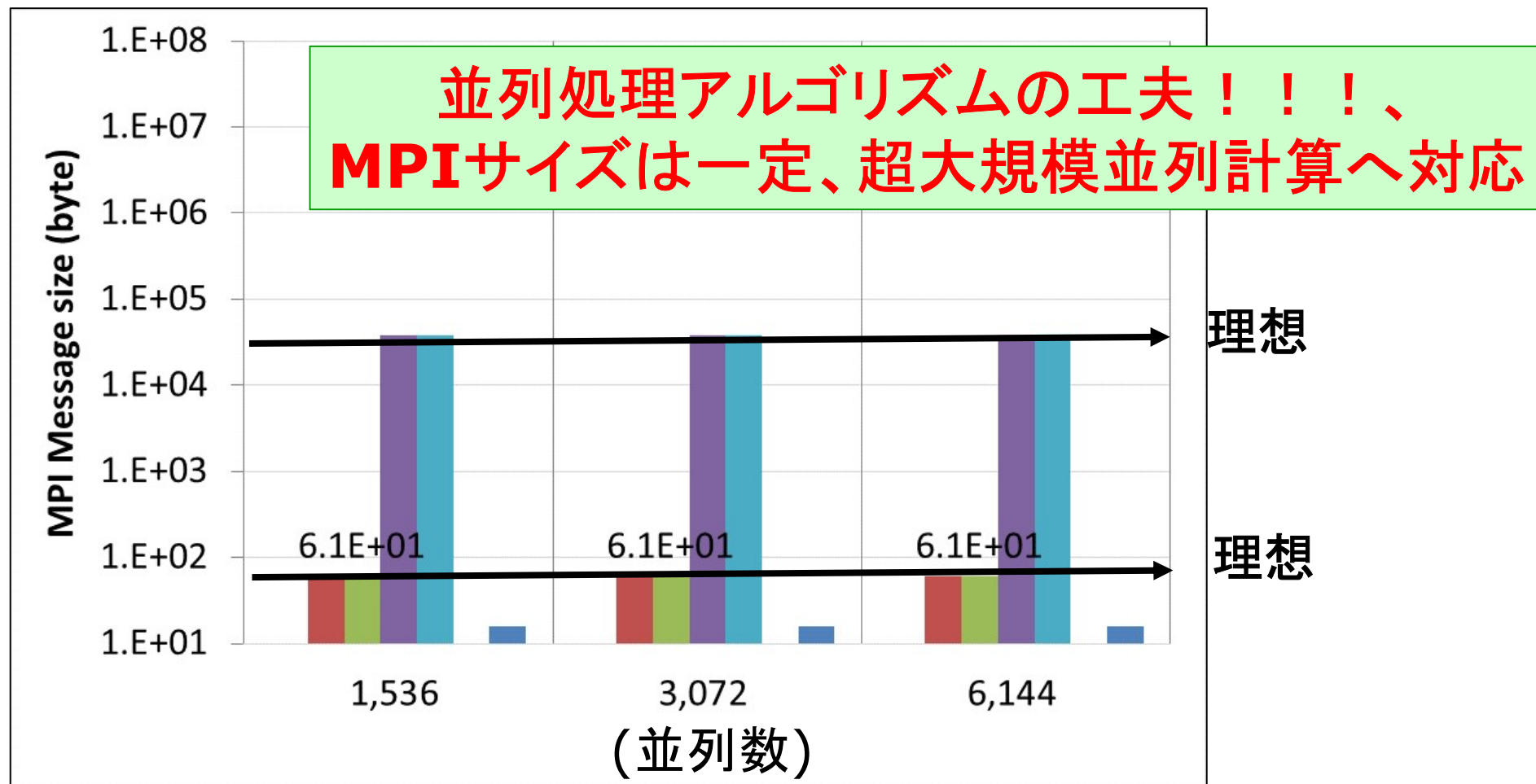
改良版

OpenFOAMの課題解決

Weak Scaling

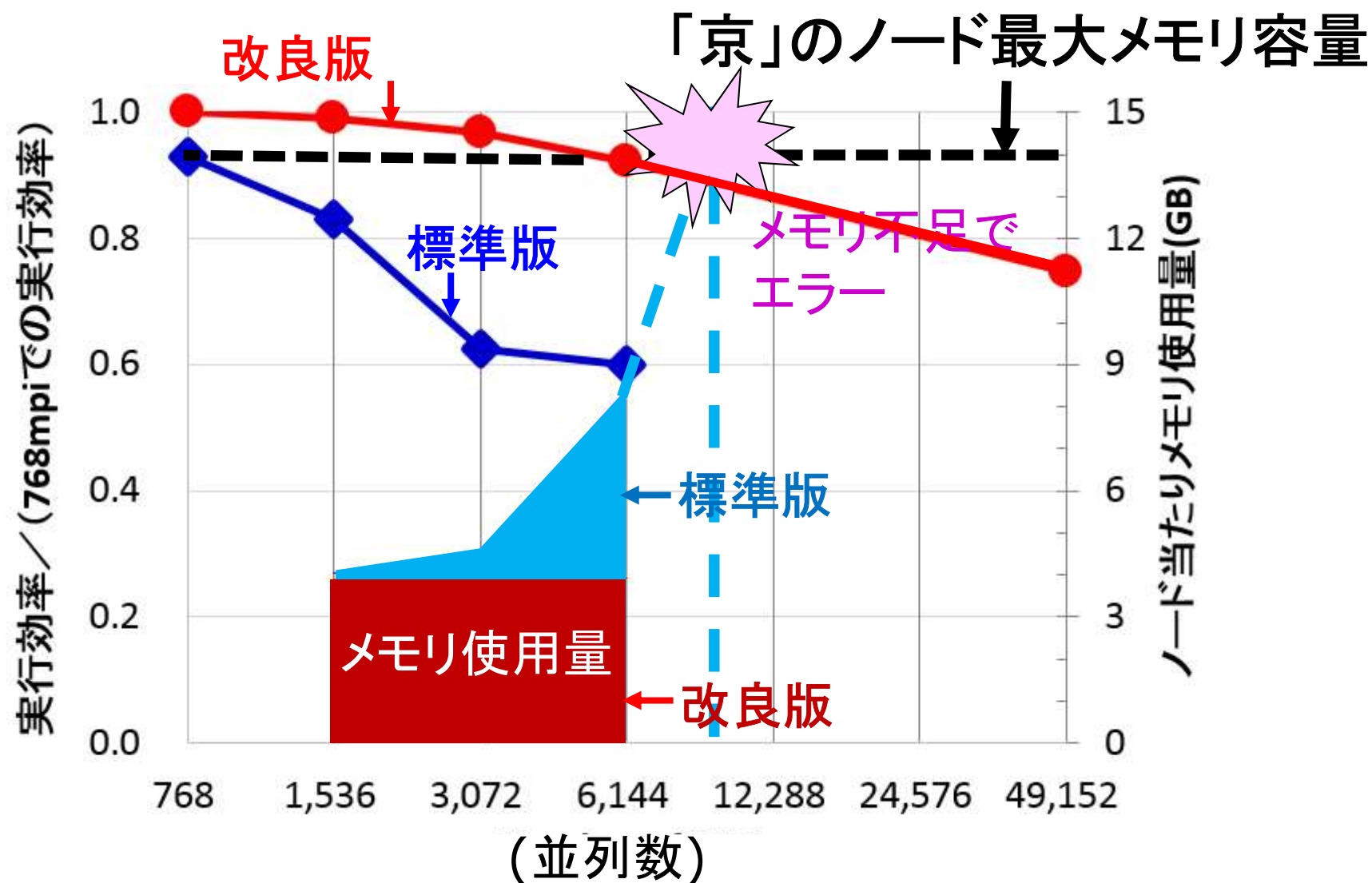
改良版

■ Probe ■ Send ■ Recv ■ Isend ■ Irecv ■ Waitall ■ Allreduce



実行効率・使用メモリ量

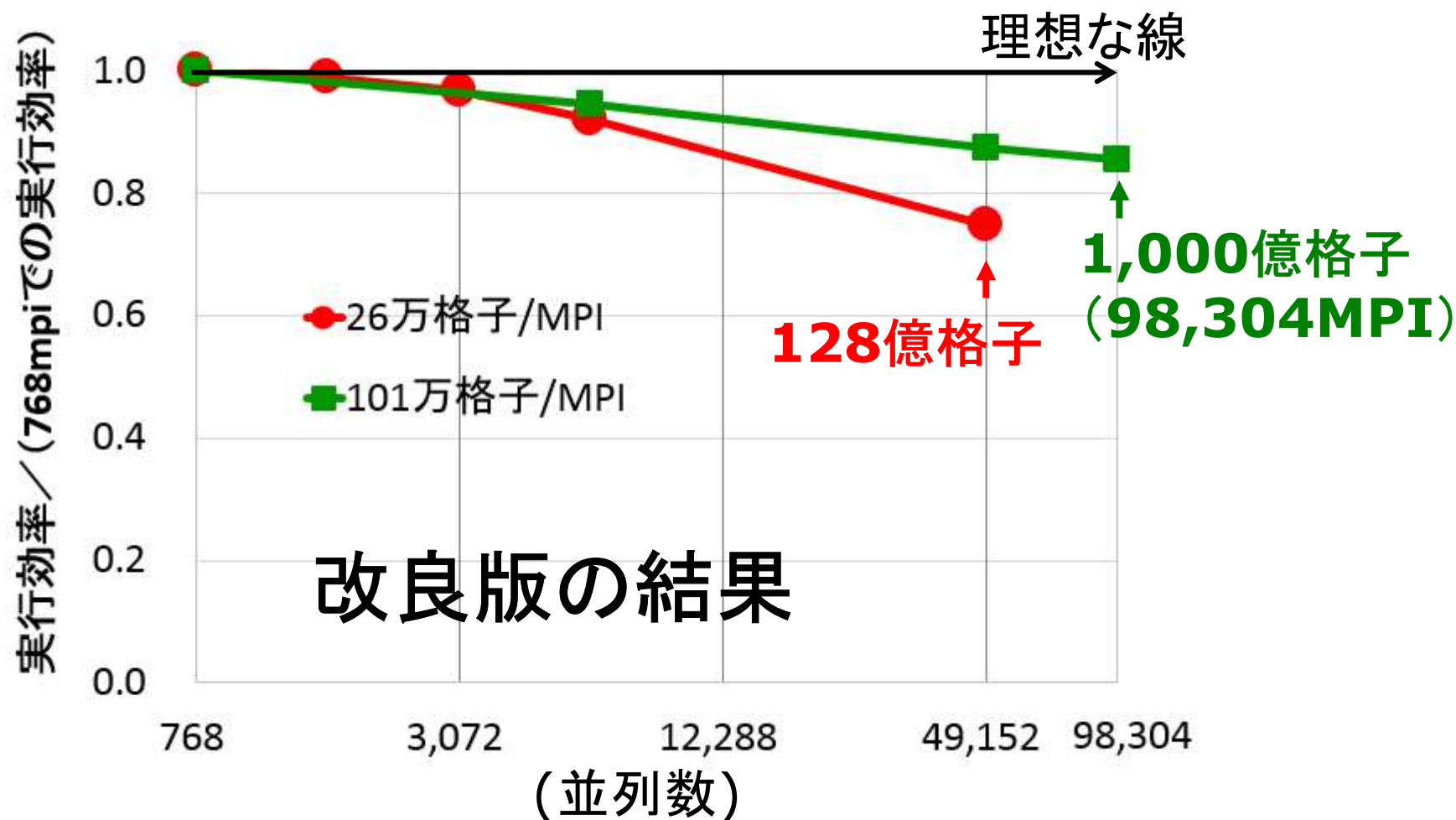
Weak Scaling



実行効率は大幅改善。使用メモリ量はほぼ一定

計算規模による性能の変化

Weak Scaling



高い並列性能を持つ1千億格子規模の計算を確認

まとめ・謝辞

■ まとめ

- 以下の対応により, OpenFOAMを用いた1千億格子規模の計算を可能にした。
 - 整数64bitコードへの改良
 - モデル作成の並列処理
 - 隣接MPI通信形態への改良

■ 謝辞

- 本検討は, 富士通(株)・**RIST**のご協力、理化学研究所のご助言、また、理化学研究所のスーパーコンピュータ「京」の利用より得られたものである(課題番号: hp150031). ここに記して謝意を表する.

END.