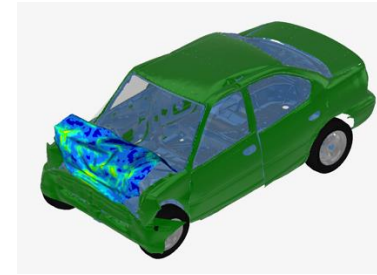


OpenRadioss (20240621) ベンチマーク @FOCUSスパコン

公益財団法人 計算科学振興財団

2024年8月

ベンチマーク設定



- ベンチマークモデル (motorbikeS)
OpenRadioss HPC Benchmark Models
(<https://openradioss.org/models/>)
Chrysler Neon 1m element HPC Benchmark (1million finite elements)
- 実行バージョン
 - OpenRadioss 20240621
(コンパイラ : Intel OneAPI 2024, @Sシステムにてコンパイル)
 - FOCUSスパコン共用領域に導入済み
/home1/share/x86_64/el8/OpenRadioss/OpenRadioss-20240621/
- 実行システム
F,H,R,S(032c,092c,192c),Z (次ページ 概要 (スペック、料金) 参照)
- 並列数
各システム 1,2,4,8,・・・ノードにて計算
並列数 = 各システムノードあたり搭載コア数 x ノード数
 - **ハイブリッド並列** : プロセスあたり4スレッド
並列数=プロセス数 x 4スレッド
 - **MPI並列** : 並列数=プロセス数
- 各システムについて、実行時間と利用料金を比較

各システム概要

システムスペック

システム	ノード内 コア数	最大ノード数	ベース 周波数(GHz)
A	12	208	2.26
B	16	2	1.86
F	40	60	2.2
H	8	136	2.1
Q	64	1	2
R	32	10	2.8
S032c	32	39	2.4
S092c	92	16	2.4
S192c	192	16	2.4
V	20	2	2.4
W	40	2	2.4
X	48	6	1.8
Z	40	24	2.1

ノード単価料金(円) 2024年度

1~	5~	9~	13~	17~	21~
75	70	65	60	55	50
100	---	---	---	---	---
400	360	320	280	240	200
75	70	65	60	55	50
600	---	---	---	---	---
400	360	320	---	---	---
320	288	256	224	192	160
800	720	640	560	480	400
1600	---	---	---	---	---
300	---	---	---	---	---
600	---	---	---	---	---
100	---	---	---	---	---
400	360	320	280	240	200

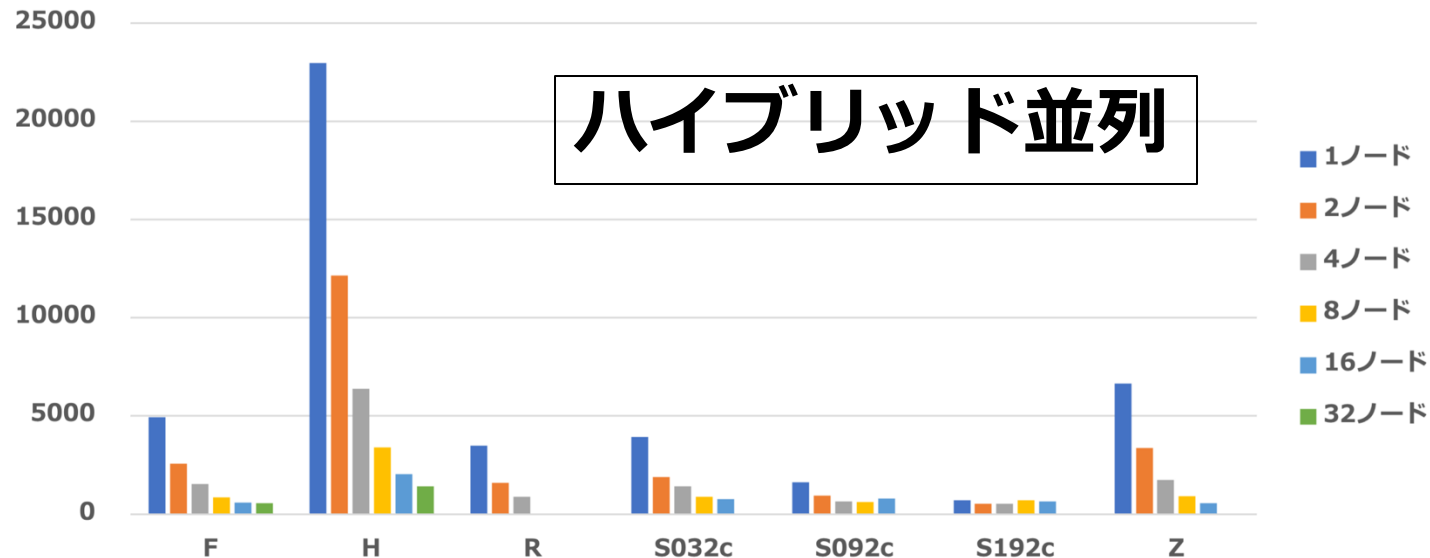
実行時間(ノード数依存)

このケースでは

- ・ハイブリッド並列の方が並列効率が高い。
- ・MPI並列の場合、並列数が大きすぎると逆に遅くなる場合があるため注意が必要

計算時間 (秒)

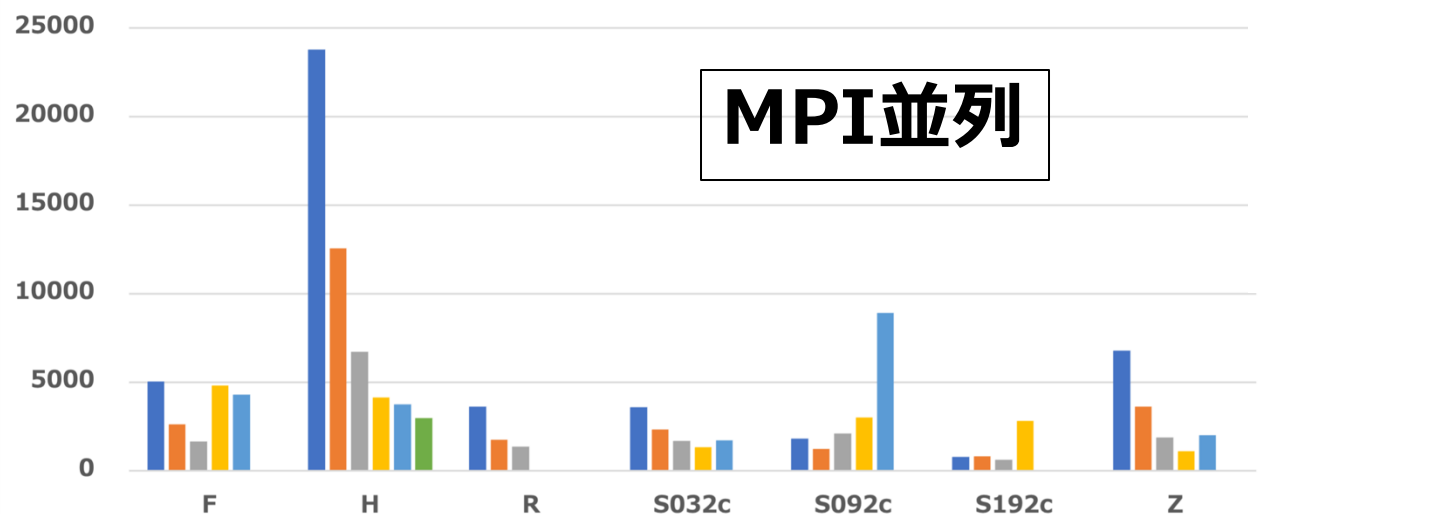
ハイブリッド並列



Better



MPI並列



Better



利用料金(ノード数依存)

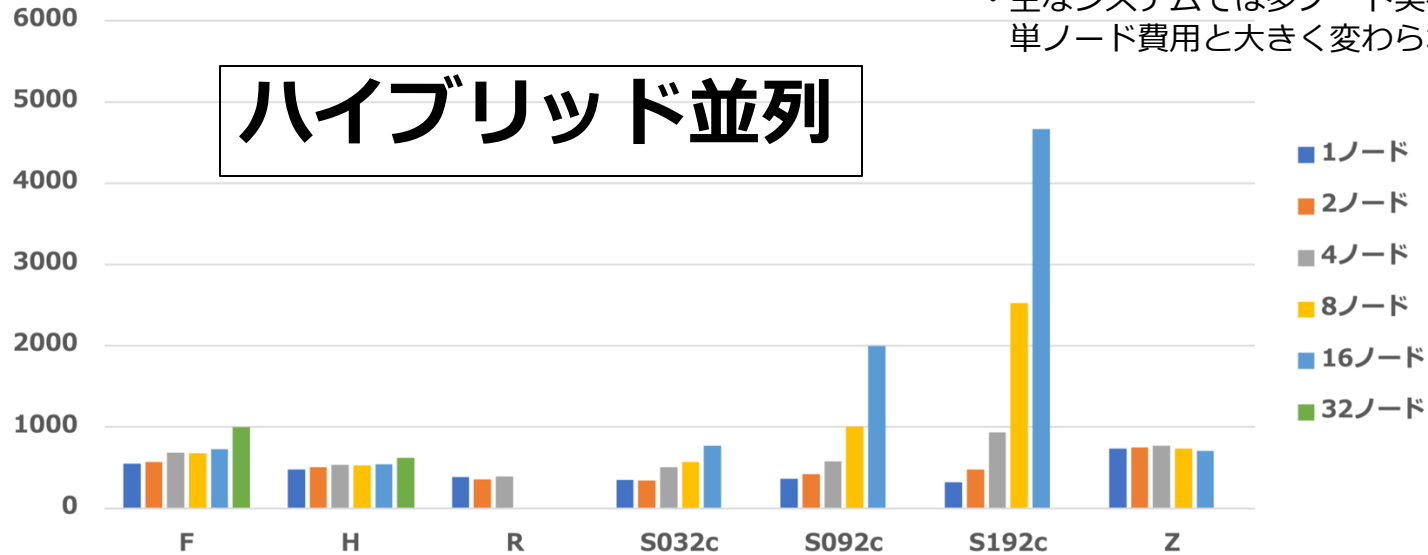
各システムの料金単価は「各システム概要」を参照してください。

このケースでは

- ・ハイブリッド並列では、並列効率のよさもあり、S192c 1ノードの計算が最も安価であった。
(高い単価にも関わらず、コストパフォーマンスが高い)
- ・主なシステムでは多ノード実行をしても費用は単ノード費用と大きく変わらない。(並列割引の効果)

料金 (円)

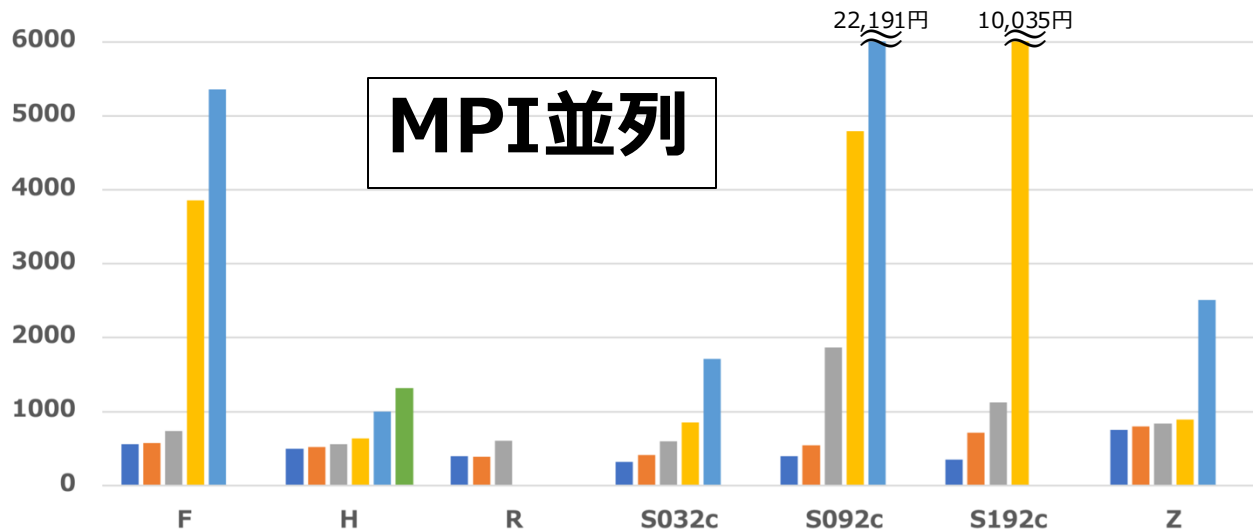
ハイブリッド並列



Better



MPI並列



Better

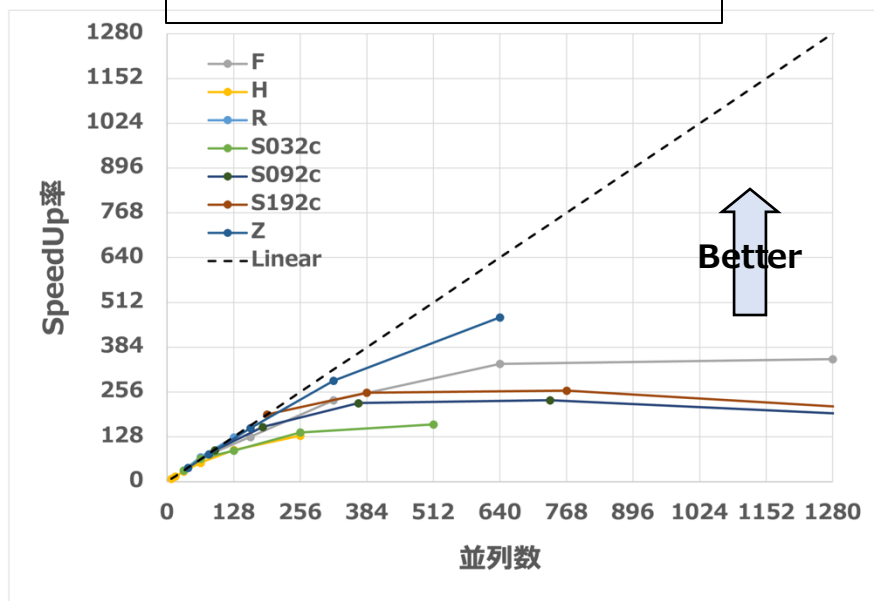


SpeedUp率(並列数依存)

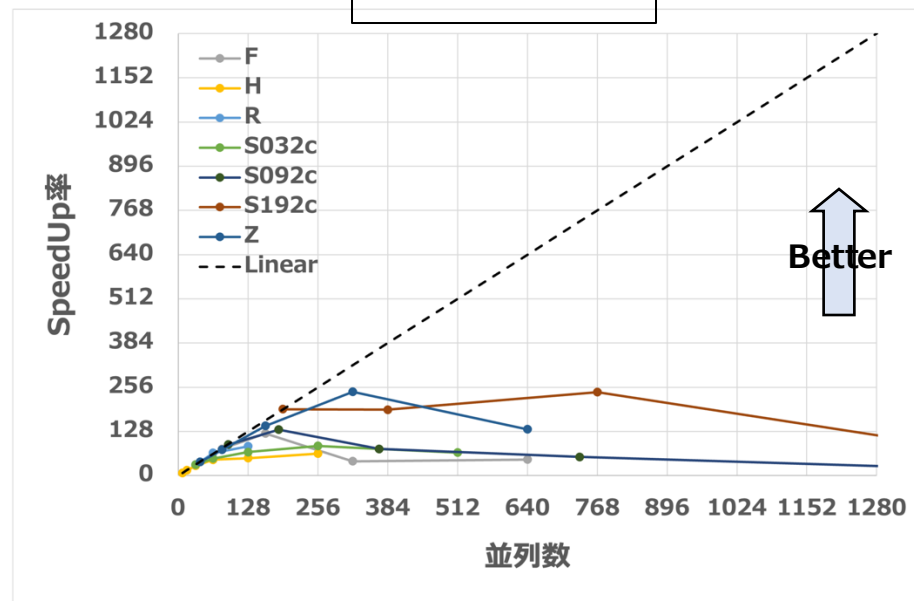
$$\text{SpeedUp率} = \text{実行時間}(1\text{並列}) / \text{実行時間}(N\text{並列})$$

実行時間(1並列) = 各システム1ノードでの実行時間 * ノード搭載コア数 とした。

ハイブリッド並列



MPI並列



このケースでは、

- ・ハイブリッド並列の方がMPI並列と比較して、効率のよい並列計算ができています。
- ・ハイブリッド並列の場合は約400並列で並列効率飽和しており、FOCUSスパコンとしてはS192c x 2ノードでの利用が最もコストパフォーマンスが高い計算と言える