

# 各種計算機アプリケーション性能比較

---

平成24年度第1四半期

## 目次

1. 多倍長演算アプリケーション
  - 1.1 ループ積分
  - 1.2 6倍、8倍精度演算
2. 倍精度演算アプリケーション
  - 2.1 ループ積分
  - 2.2 重力多体問題(N体問題)
3. 単精度演算アプリケーション
  - 3.1 姫野ベンチ

## 各種計算機

アーキテクチャの相違は性能のみならず,精度,コンパイラの最適化機能,互換性にも影響が出てきます。主に使用した計算機は以下の3つです。

(ア)SR16000/M1

プロセッサ:power7

周波数:3.83GHz

1ノード当たり

CPUコア数 32(物理的),64(論理的)

理論最大性能 980.48 GFLOPs

メモリ容量 256GB

メモリアーキテクチャー NUMA,(16論理コア単位でflat)

SIMD(Single Instruction Multiple Data)を

サポートするVSX機構付き

L3キャッシュ On-Chip 32MB/8コア

演算器/物理コア 乗加算器4つ

(イ)BG/Q

周波数 1.6GHz

1ノード 16core 論理性能 204.8GFLOPs

L1 キャッシュ 16/16KB (Core)

L2 32MB (node)

Main storage 16GB (Core)

Smt=1,2,4

## (ウ)GPGPU

GPU カード型番:ATI RadeonHD5880

メモリ: GDDR5, 1 GB, 153.6 GB/s

ホストインタフェース: PCI Express 2.1 x16stream

processing unit: 3200個(演算プロセッサ)

動作周波数: 850 MHzピーク性能(単精度): 5440 Gflops

(=3200x2x850MHz)ピーク性能(倍精度): 1088 Gflops

# 1. 多倍長演算アプリケーション

## 1. 1 ループ積分

測定した数値積分の積分式,解析近似解は以下の様なものです。

$$I = \int_0^1 \int_0^{1-x} \int_0^{1-x-y} \frac{1}{D^2} dz dy dx \quad \text{式}$$

$$D = -sxy - tz(1-x-y-z) + (x+y)\lambda^2 + (1-x-y-z)(1-x-y)m_e^2 \\ + z(1-x-y)m_f^2$$

$$s = -500^2, t = -150^2, m_f = 150,$$

$$m_e = 0.0005, \lambda = 10^{-30} \quad \text{データ}$$

### 解析近似解

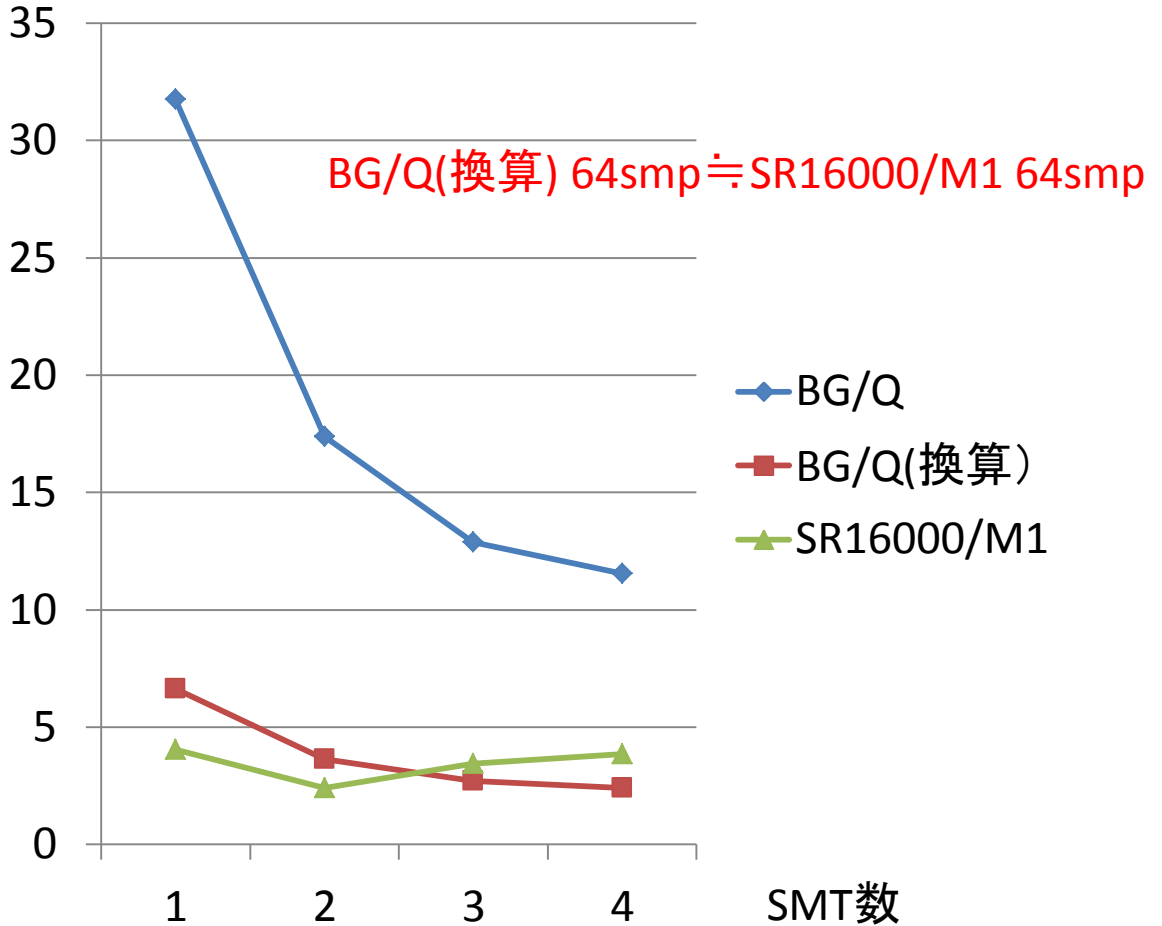
$$I = \frac{1}{-s(-t+m_f^2)} \ln\left(\frac{-s}{\lambda^2}\right) \ln \frac{(-t+m_f^2)^2}{m_e^2 m_f^2}$$

相対誤差  $7 \times 10^{-26}$  以下

# 1 ノード性能比較

サイズN=1024

実行時間(秒)

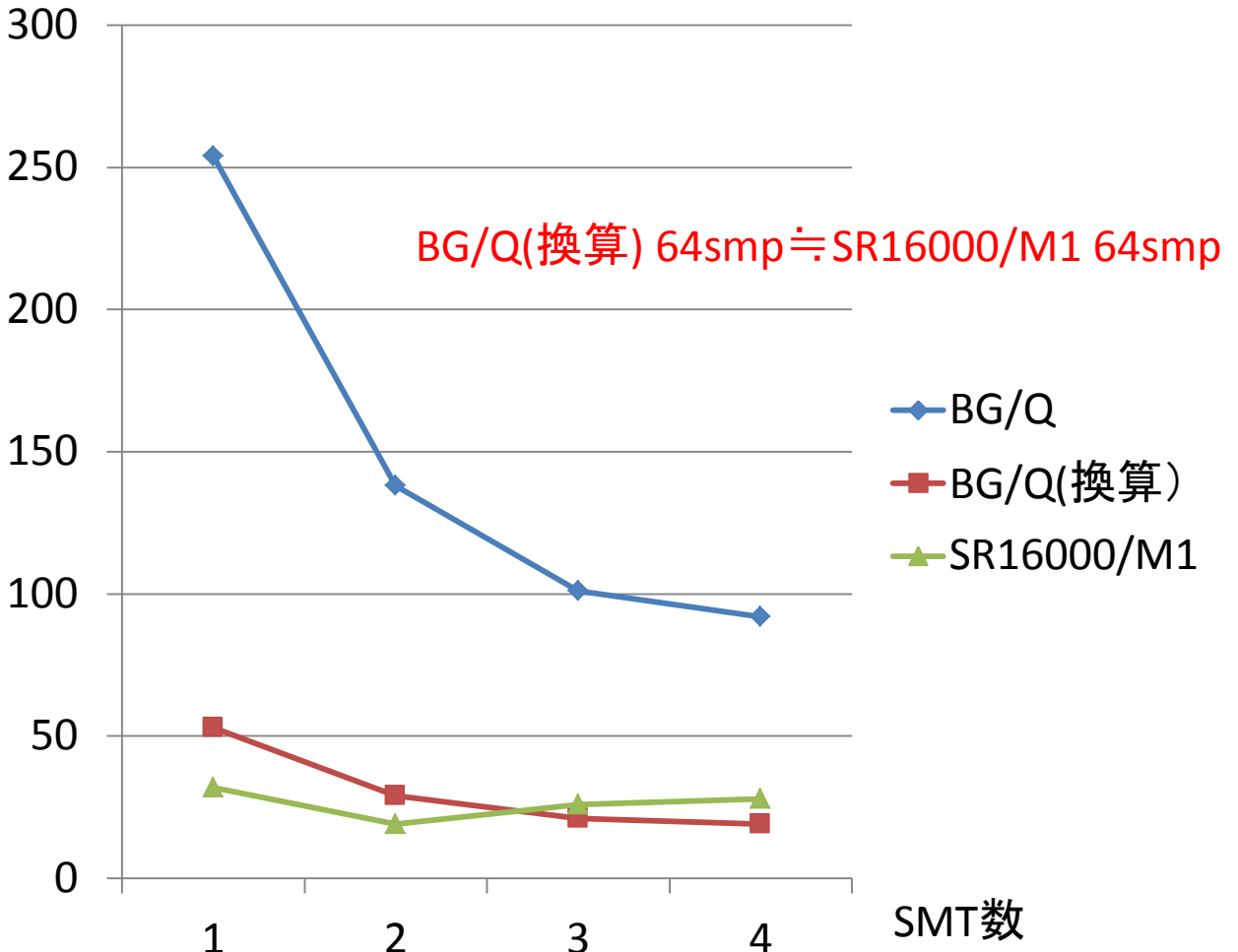


	BG/Q	SR16000/M1
SMT=1	16 smp	32 smp
=2	32 smp	64 smp
=3	48 smp	96 smp
=4	64 smp	128 smp

BG/Q(換算) 204.8GFLOPs => 980.48GFLOPs にしたときの値

# サイズN=2048

実行時間(秒)

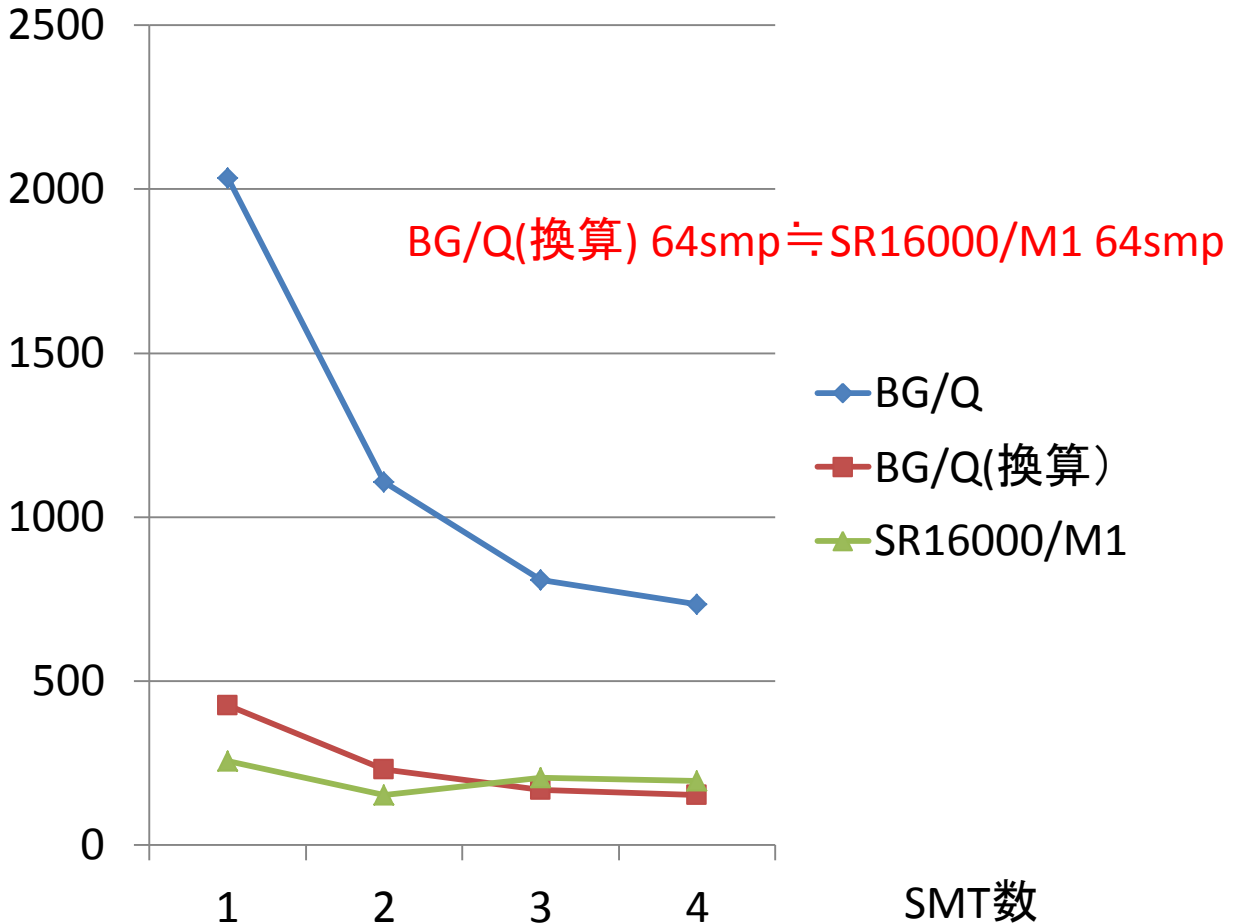


	BG/Q	SR16000/M1
SMT=1	16 smp	32 smp
=2	32 smp	64 smp
=3	48 smp	96 smp
=4	64 smp	128 smp

BG/Q(換算) 204.8GFLOPs => 980.48GFLOPs にしたときの値

サイズN=4096

実行時間(秒)

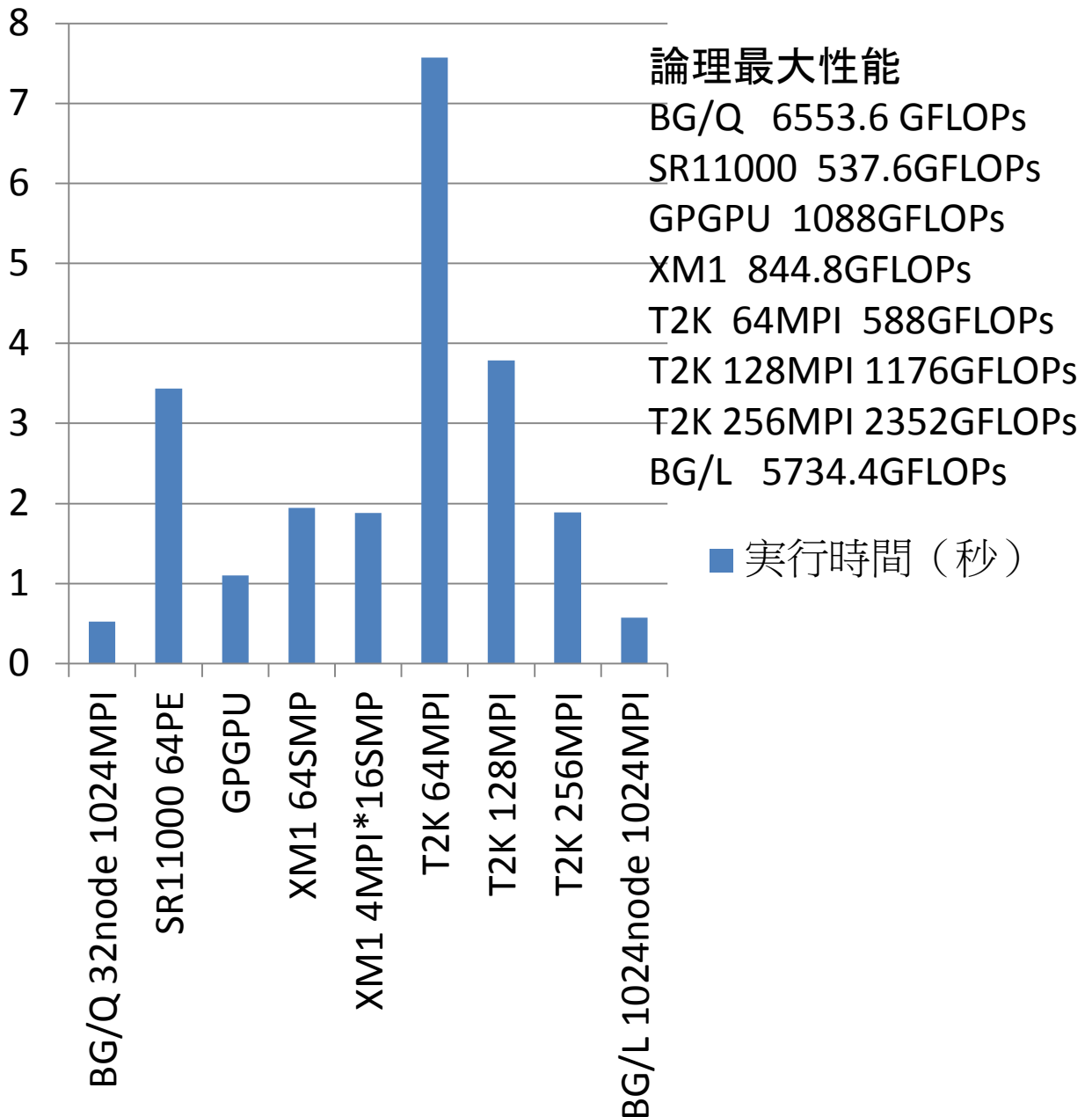


	BG/Q	SR16000/M1
SMT=1	16 smp	32 smp
=2	32 smp	64 smp
=3	48 smp	96 smp
=4	64 smp	128 smp

BG/Q(換算) 204.8GFLOPs => 980.48GFLOPs にしたときの値

# 4倍精度INFRA BOX N=1024 各種計算機実行時間(1)

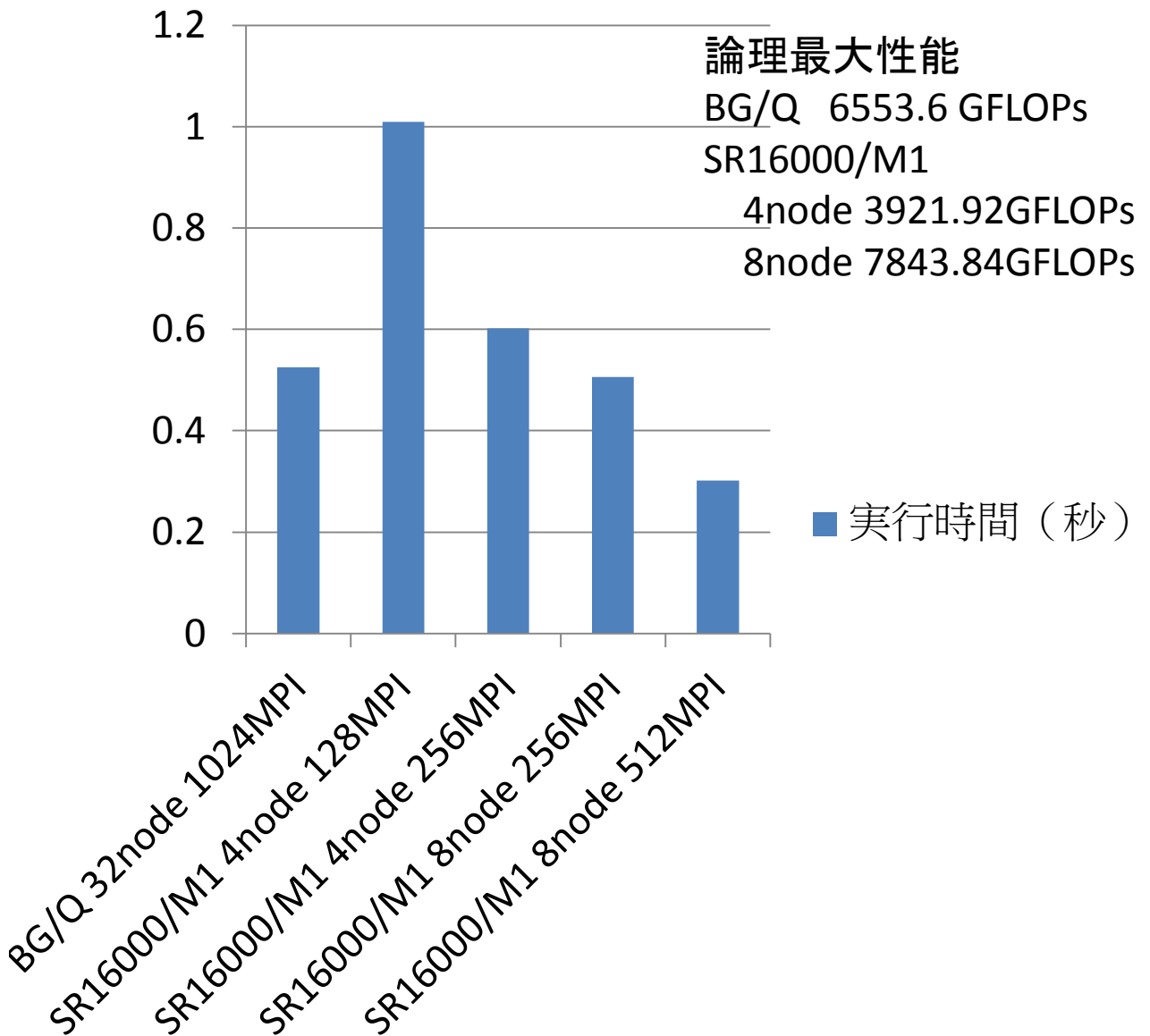
## 実行時間 (秒)





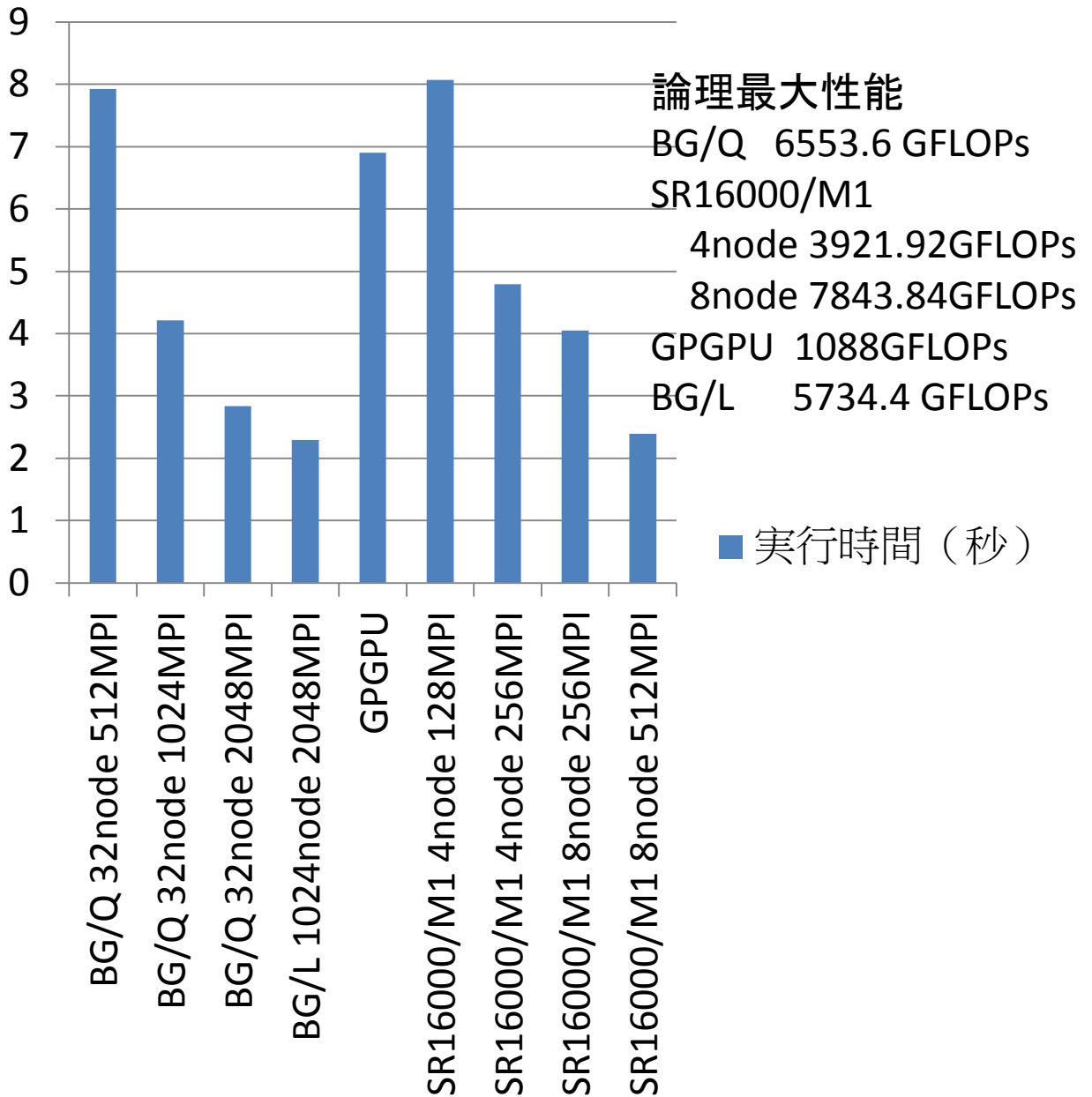
## 4倍精度INFRA BOX N=1024 各種計算機実行時間(2)

### 実行時間 (秒)



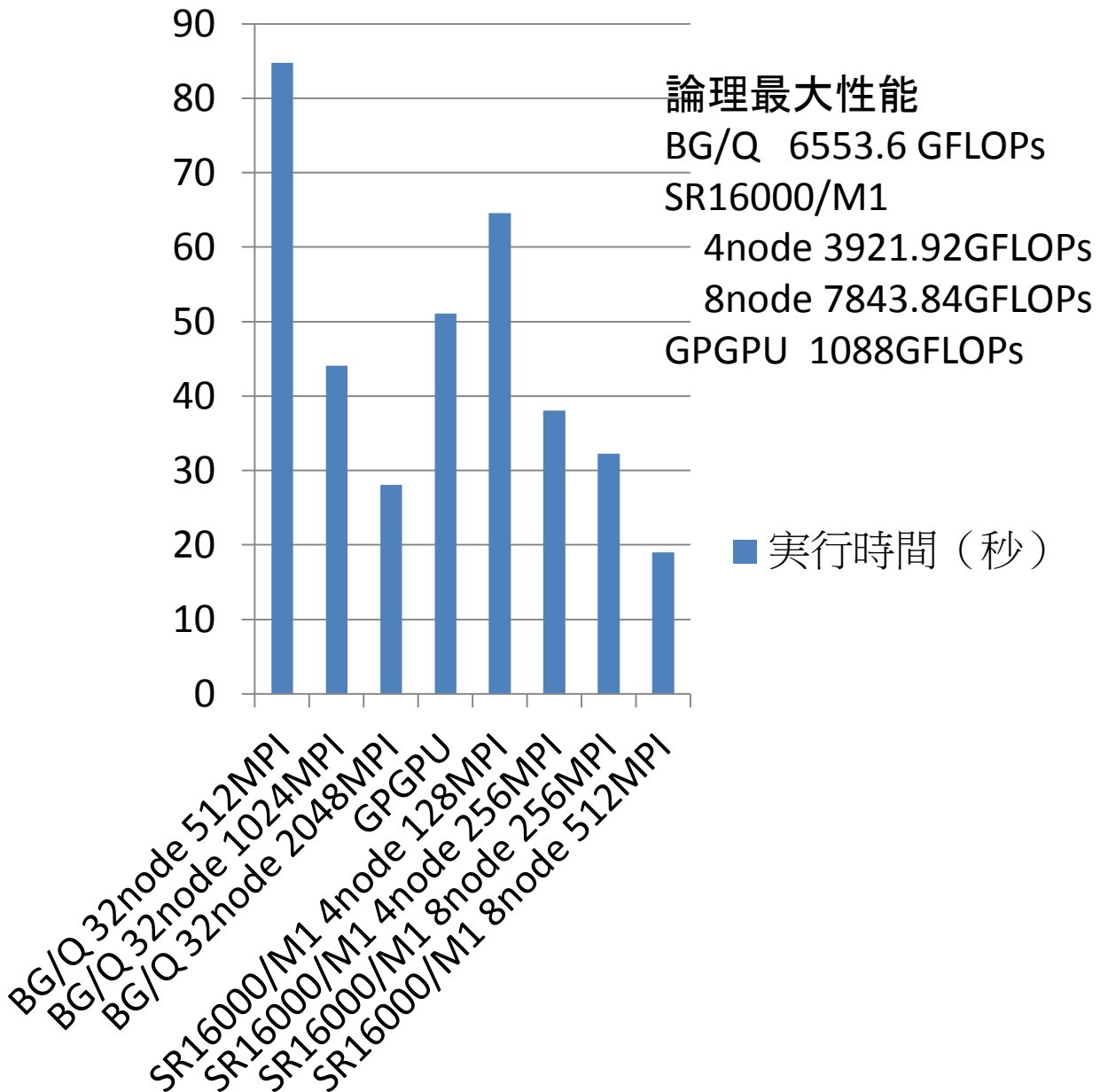
# 4倍精度INFRA BOX N=2048 各種計算機実行時間

## 実行時間（秒）



# 4倍精度INFRA BOX N=4096 各種計算機実行時間

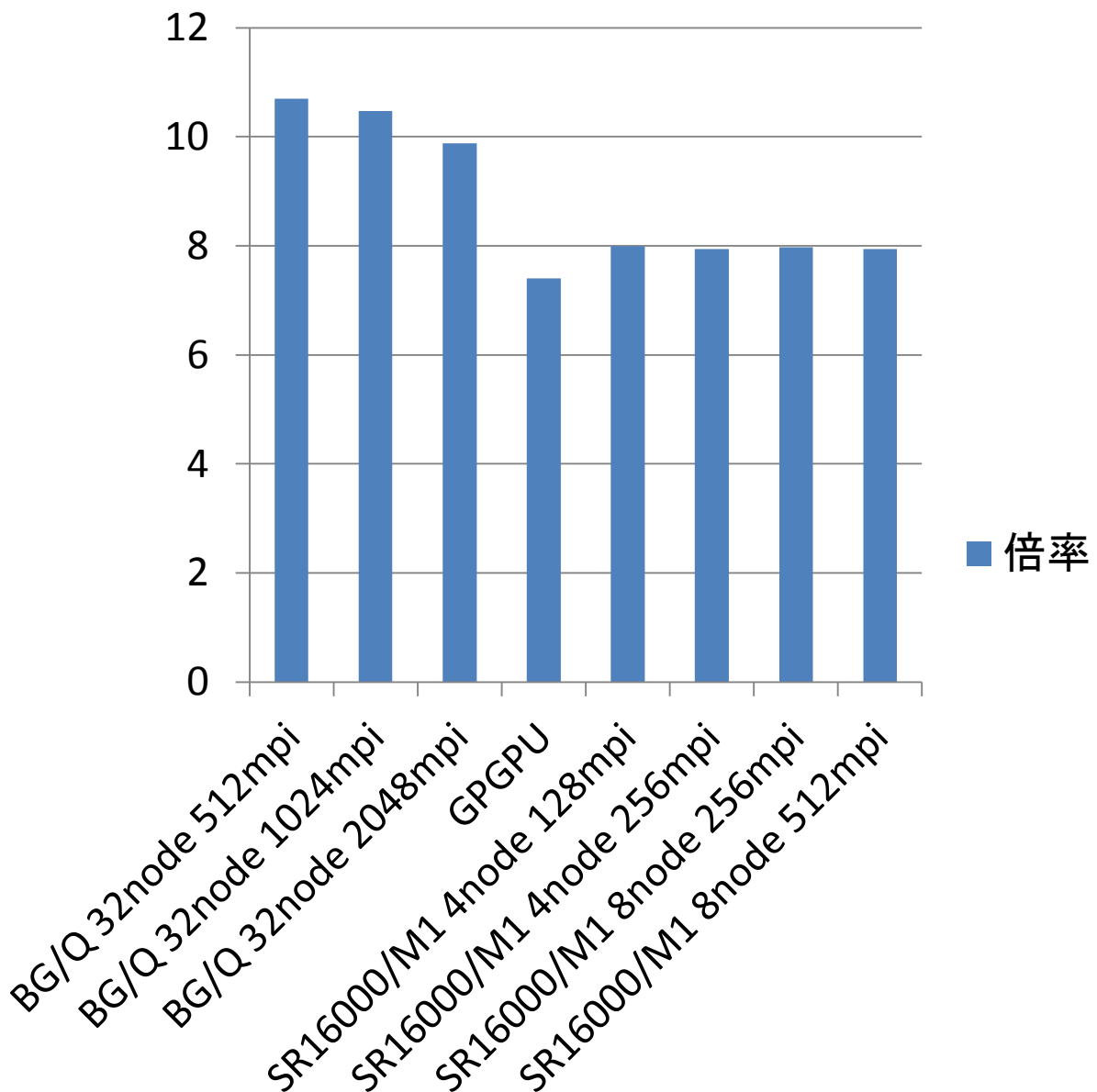
## 実行時間（秒）



# 4倍精度INFRA BOX N=2048とN=4096 各種計算機実行時間比率

演算量比率(N=4096/N=2048)=8

## 実行時間(N=4096/N=2048)比率



## 1.2 6倍、8倍精度演算

ここでは、泊&平木(2011) SWOPP'11 で  
80MHzのFPGAで作成した8倍精度演算器の性能  
537MFLOPs(OMFLOPs)と比較を行っています。

GPGPU 3200Core

N=1,000,000 の内積演算 3200回実行 演算量 6400MFLOP

精度	実行時間(秒)	性能(MFLOPs)
6倍精度	1.65	3879
8倍精度	1.88	3401

SR16000/M1 1ノード

N=2\*\*24 の内積演算 100回実行 演算量 3355.44MFLOP

		f90 実行結果	
精度	MPI数	実行時間(秒)	性能(MFLOPs)
6倍精度	16	2.415	1389
	32	1.745	1923
	64	0.878	3822
8倍精度	16	5.869	572
	32	3.277	1024
	64	1.854	1810

SR16000/M1 1ノード

サイズN=2048 の行列積 演算量 17179.8 MFLOP

コンパイラ		f90		xlf_r	
精度	スレッド数	実行時間(秒)	性能(MFLOPs)	実行時間(秒)	性能(MFLOPs)
6倍精度	1	172.13	100	122.83	140
	8	32.13	535	13.96	1231
	16	16.08	1068	6.98	2461
	32	8.04	2137	6.128	2780
	64	4.02	4274	4.41	3896
	128	6.31	2723	3.94	4360
8倍精度	1	414.51	41	261.94	66
	8	72.79	236	30.97	555
	16	36.43	472	25.65	670
	32	18.22	943	13.02	1320
	64	9.11	1886	8.57	2005
	128	12.89	1333	8.91	1928
			赤字 最高速		

## 2 倍精度演算アプリケーション

### 2.1 ループ積分

倍精度演算で十分な精度がでて並列化効果の大きい5つの数値積分プログラムで性能を比較しています。

(a)s221

$$S^{221}(s; m_1^2, m_2^2, m_3^2, m_4^2, m_5^2) = \int_0^1 \int_0^{1-x} \int_0^{1-x-y} \int_0^{1-x-y-z} \frac{1}{DC} dudzdydx$$

$$C = (x + y + z + u)(1 - x - y - z - u) + (x + y)(z + u)$$

$$E = (1 - x - y - z - u)(x + z)(y + u) + (x + y)zu + (z + u)xy$$

$$M^2 = xm_1^2 + ym_2^2 + zm_3^2 + um_4^2 + (1 - x - y - z - u)m_5^2$$

$$D = -sE + M^2C$$

テストケース

$$s^{221}(-1:100, 100, 0, 0, 100)$$

解析近似解 : 0.0380004438127

(b)laportad

$$I = \int_0^1 \int_0^{1-x_1} \int_0^{1-x_1-x_2} \int_0^{1-x_1-x_2-x_3} \int_0^{1-x_1-x_2-x_3-x_4} \frac{1}{D^2} dx_5 dx_4 dx_3 dx_2 dx_1$$

$$x_6 = 1 - x_1 - x_2 - x_3 - x_4 - x_5$$

D =

$$\begin{aligned} & \&-x1^{**2}*x2-x1^{**2}*x3-x1^{**2}*x4-x1^{**2}*x6-x1*x2^{**2}-x1*x2*x3 \\ & \&-2.d0*x1*x2*x4 \\ & \&-x1*x2*x5-x1*x2*x6-x1*x3^{**2}-2.d0*x1*x3*x4-x1*x3*x5- \\ & x1*x3*x6 \\ & \&-x1*x4^{**2} \\ & \&-x1*x4*x5-2.d0*x1*x4*x6-x1*x5*x6-x1*x6^{**2}-x2^{**2}*x4- \\ & x2^{**2}*x5 \\ & \&-x2*x3*x4 \\ & \&-x2*x3*x5-x2*x4^{**2}-2.d0*x2*x4*x5-x2*x4*x6-x2*x5^{**2}- \\ & x2*x5*x6 \\ & \&-x3^{**2}*x4 \\ & \&-x3^{**2}*x5-x3*x4^{**2}-2.d0*x3*x4*x5-x3*x4*x6-x3*x5^{**2}- \\ & x3*x5*x6 \\ & \&-x4^{**2}*x5 \\ & \&-x4^{**2}*x6-x4*x5^{**2}-3.d0*x4*x5*x6-x4*x6^{**2}-x5^{**2}*x6- \\ & x5*x6^{**2} \end{aligned}$$

解析近似解=0.2762092253588



(c) laportag

$$I = \int_0^1 \int_0^{1-x_1} \int_0^{1-x_1-x_2} \int_0^{1-x_1-x_2-x_3} \int_0^{1-x_1-x_2-x_3-x_5} \int_0^{1-x_1-x_2-x_3-x_5-x_6} \frac{C}{D^3} dx_7 dx_6 dx_5 dx_3 dx_2 dx_1$$

$$x_4 = 1 - x_1 - x_2 - x_3 - x_5 - x_6 - x_7$$

$$C = x_1 * x_4 + x_1 * x_5 + x_1 * x_6 + x_2 * x_4 + x_2 * x_5 + x_2 * x_6 + x_3 * x_4 + x_3 * x_5 + x_3 * x_6 + x_4 * x_5$$

$$\& + x_4 * x_6 + x_4 * x_7 + x_5 * x_7 + x_6 * x_7$$

$$D =$$

&-

$$(x_1^{**2} + x_2^{**2} + x_3^{**2} + x_7^{**2} + x_1 * x_2 + x_1 * x_3 + x_1 * x_7 + x_2 * x_3 + x_2 * x_7 + x_3 * x_7)$$

$$\& * (x_4 + x_5 + x_6)$$

$$\& - x_4^{**2} * (x_1 + x_2 + x_3 + x_5 + x_6 + x_7)$$

$$\& - (x_5^{**2} + x_6^{**2} + x_5 * x_6) * (x_1 + x_2 + x_3 + x_4 + x_7)$$

$$\& - 3.0 * x_4 * (x_1 * x_5 + x_6 * x_7)$$

$$\& - 2.0 * ((x_1 + x_2 + x_3) * x_4 * x_6 + (x_2 + x_3 + x_7) * x_4 * x_5)$$

解析近似解=0.1723367907503

(d)laportah

$$I = \int_0^1 \int_0^{1-x_1} \int_0^{1-x_1-x_3} \int_0^{1-x_1-x_3-x_2} \int_0^{1-x_1-x_3-x_2-x_7} \int_0^{1-x_1-x_3-x_2-x_7-x_6} \frac{C}{D^3} dx_4 dx_6 dx_7 dx_2 dx_3 dx_1$$

$$x_5 = 1 - x_1 - x_3 - x_2 - x_7 - x_6 - x_4$$

$$C = (x_1 + x_2 + x_3 + x_4) * (x_4 + x_5 + x_6 + x_7) - x_4 * x_4$$

$$cc = x_1 * m_{12} + x_2 * m_{22} + x_3 * m_{32} + x_4 * m_{42} + x_5 * m_{52} + x_6 * m_{62} + x_7 * m_{72}$$

$$D = -c * cc$$

$$. + s * (x_1 * x_2 * (x_4 + x_5 + x_6 + x_7) + x_5 * x_6 * (x_1 + x_2 + x_3 + x_4) + x_1 * x_4 * x_6 + x_2 * x_4 * x_5)$$

$$. + t * x_3 * x_4 * x_7$$

$$. + p_{12} * (x_1 * x_3 * (x_4 + x_5 + x_6 + x_7) + x_3 * x_4 * x_5)$$

$$. + p_{22} * (x_2 * x_3 * (x_4 + x_5 + x_6 + x_7) + x_3 * x_4 * x_6)$$

$$. + p_{32} * (x_5 * x_7 * (x_1 + x_2 + x_3 + x_4) + x_1 * x_4 * x_7)$$

$$. + p_{42} * (x_6 * x_7 * (x_1 + x_2 + x_3 + x_4) + x_2 * x_4 * x_7)$$

## テストデータ

m12=1.0d0

m22=1.0d0

m32=1.0d0

m42=1.0d0

m52=1.0d0

m62=1.0d0

m72=1.0d0

p12=1.0d0

p22=1.0d0

p32=1.0d0

p42=1.0d0

s =1.0d0

t =1.0d0

解析近似解=0.1036407209893

(e)laportai

$$I = \int_0^1 \int_0^{1-x_1} \int_0^{1-x_1-x_2} \int_0^{1-x_1-x_2-x_3} \int_0^{1-x_1-x_2-x_3-x_4} \int_0^{1-x_1-x_2-x_3-x_4-x_5} \frac{C}{D^3} dx_6 dx_5 dx_4 dx_3 dx_2 dx_1$$

$$x_7 = 1 - x_1 - x_2 - x_3 - x_4 - x_5 - x_6$$

$$C = (x_1 + x_2 + x_3 + x_4 + x_5) * (x_1 + x_2 + x_3 + x_6 + x_7) - (x_1 + x_2 + x_3) ** 2$$

$$cc = x_1 * m_{12} + x_2 * m_{22} + x_3 * m_{32} + x_4 * m_{42} + x_5 * m_{52} + x_6 * m_{62} + x_7 * m_{72}$$

$$D = -c * cc$$

$$. + s * (x_1 * x_2 * (x_4 + x_5 + x_6 + x_7) + x_1 * x_5 * x_6 + x_2 * x_4 * x_7 - x_3 * x_4 * x_6)$$

$$. + t * x_3 * (-x_4 * x_6 + x_5 * x_7)$$

$$. + p_{12} * (x_1 * x_3 * (x_4 + x_5 + x_6 + x_7) + x_3 * x_4 * (x_6 + x_7))$$

$$. + p_{22} * (x_2 * x_3 * (x_4 + x_5 + x_6 + x_7) + x_3 * x_6 * (x_4 + x_5))$$

$$. + p_{32} * (x_4 * x_5 * (x_1 + x_2 + x_3 + x_6 + x_7) + x_4 * x_6 * (x_2 + x_3) + x_1 * x_5 * x_7)$$

$$. + p_{42} * (x_6 * x_7 * (x_1 + x_2 + x_3 + x_4 + x_5) + x_4 * x_6 * (x_1 + x_3) + x_2 * x_5 * x_7)$$

## テストデータ

m12=1.0d0

m22=1.0d0

m32=1.0d0

m42=1.0d0

m52=1.0d0

m62=1.0d0

m72=1.0d0

p12=1.0d0

p22=1.0d0

p32=1.0d0

p42=1.0d0

s =1.0d0

t =1.0d0

解析近似解=0.0853513981538

SR16000/M1 複数ノード

LAPORTA (D)

MPI数	ノード数	実行時間 (秒)
128	4	3.231716
256	4	2.165075
256	8	1.617797
512	8	1.107155

LAPORTA(G)

MPI数	ノード数	実行時間 (秒)
128	4	684.465022
256	4	544.483556
256	8	345.817823
512	8	277.552501

LAPORTA (H)

MPI数	ノード数	実行時間 (秒)
128	4	541.664086
256	4	387.666776
256	8	270.937051
512	8	198.333612

LAPORTA (I)

MPI数	ノード数	実行時間 (秒)
128	4	728.709051
256	4	508.013126
256	8	368.304897
512	8	258.708141

S221 N=2048

MPI数	ノード数	実行時間 (秒)
128	4	467.774509
256	4	412.210191
256	8	233.916246
512	8	206.360301

GPGPU

LAPORTA

プログラム	実行時間(秒)
D	5.85
G	589.553
H	569.934
I	611.884
S221	2166.834527

BG/Q

LAPORTA(D)

MPI数	ノード数	実行時間 (秒)
512	32	12.844877
1024	32	6.633127
2048	32	3.967958

LAPORTA(G)

MPI数	ノード数	実行時間 (秒)
512	32	977.217064
1024	32	541.747376
2048	32	439.703755

LAPORTA(H)

MPI数	ノード数	実行時間 (秒)
512	32	946.719043
1024	32	514.122564
2048	32	391.466375

LAPORTA(I)

MPI数	ノード数	実行時間 (秒)
512	32	954.046258
1024	32	532.563084
2048	32	430.696899

S221 N=2048

MPI数	ノード数	実行時間 (秒)
512	32	2144.31316
1024	32	1161.5669
2048	32	801.675789

理論最大性能(GFLOPs)

BG/Q 6553.6

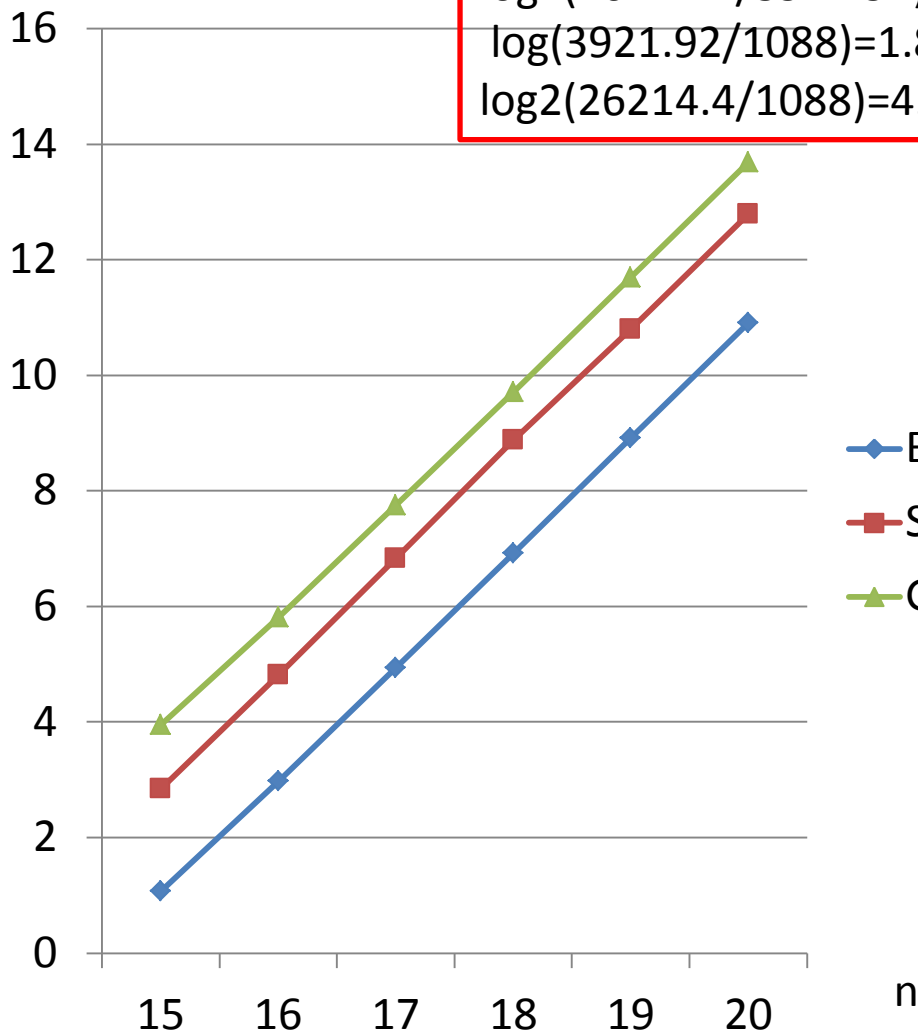
SR16000 4node 3921.92

8node 7843.84

GPGPU 1088

## 2.2 重力多体問題(N体問題)

Log2(実行時間(秒))



比較用換算値

$$\log_2(26214.4/3921.92)=2.74$$

$$\log(3921.92/1088)=1.85$$

$$\log_2(26214.4/1088)=4.59$$

◆ BG/Q  
■ SR16000/M1  
▲ GPGPU

粒子数 $N=2^{**}n$

BG/Q 128node (SMP\*MPI)ハイブリッド

理論最大性能 26214.4 GFLOPs

SR16000/M1 4node 128MPI フラット

理論最大性能 3921.92GFLOPs

GPGPU 3200 core

理論最大性能 1088GFLOPs

### 3. 単精度演算アプリケーション

#### 3.1 姫野ベンチ

サイズ2049\*2049\*1025 フラットMPI

ソース	CPU	MPI数	ノード数	論理最大性能 (GFLOPs)	実行性能 (GFLOPs)
ori	BG/Q	2048	32	6553.6	276
	BG/Q	8192	128	26214.4	975
	BG/Q	32768	512	104857.6	4163
	SR16000/M1	256	4	3921.92	135
	SR16000/M1	512	8	7843.84	269
tune	BG/Q	2048	32	6553.6	314
	BG/Q	8192	128	26214.4	1065
	BG/Q	32768	512	104857.6	5038
	SR16000/M1	256	4	3921.92	488
	SR16000/M1	512	8	7843.84	967

実行効率      オリジナル      BG/Qが良い  
                 チューニング      SR16000/M1      が良い