

各種計算機基本性能調査

平成25年度第1四半期

目次

1.はじめに

2.SR16000/M1 システム

2.1 QCD 計算

2.2 量子モンテカルロ法による物性スペクトル計算

2.3 Rump's例題計算

2.4 6倍精度,8倍精度,10倍精度行列積計算

2.5 3次元反復計算

3.BG/Q システム

3.1 QCD 計算

1. はじめに

使用しました計算機の論理最大性能は以下の様なものです。

SR16000/M1	1ノード	980.48GFLOPs
BG/Q	1ノード	204.8GFLOPs

扱ったプログラムは

1.QCD MPI版

**2.量子モンテカルロ法による物性スペクトル計算
に現れるGREEN関数計算**

**3.3次元反復解法
です。**

**また,Parallel Program Generator に関しては、
サーバー系に適応した例を各種計算機性能比較
で記述しました。**

2.SR16000/M1 システム

2.1 QCD計算

チューニングはコンパイラオプションの選択のみ。

テストデータは $32*32*32*64$

実行時間一覧表 (秒)

	1ノード			
	1smp	16smp	32 smp	64 smp
total	3760.4	508.39	339.15	396.52
solver	3754.9	503.22	336.02	393.14
cg-step	3714.7	497.45	332.08	388.36
mult	2867.7	388.25	266.41	299.02

**Smp並列では32smpが最高性能で、
実行時間は339秒**

mpi,smp並列での性能測定結果

コンパイルオプション、jcfのチューニングのみ

QCD 32*32*32*64 並列実行性能測定結果

実行時間一覧表(秒)

node	mpi	smp	total	solver	cg-step	mult	com
1	16	2	237.28	236.42	233.85	187.84	13.09
1	16	4	236.31	236.03	233.47	187.54	13.44
1	8	4	217.4	216.63	214.25	168.87	17.29
1	8	8	220.3	219.26	216.84	170.61	17.85
1	4	8	222.64	221.08	218.57	172.21	21.02
1	4	16	223.91	222.42	219.9	172.76	21.34
4	16	8	98.58	96.63	95.49	88.27	14.28
4	16	16	101.56	99.56	98.4	91.02	14.69
4	8	16	98.56	96.54	95.38	80.32	21.44
4	8	32	103.23	100.98	99.76	84.76	25.62
4	4	32	97.38	96.1	94.8	79.44	21.22
4	4	64	122.8	121.32	119.75	103.6	23.07
4	256	1	46.37	44.45	43.94	36.87	5.7
4	128	1	69.46	67.7	66.92	59.77	5.35
4	128	2	62.67	60.83	60.16	48.65	5.79
4	64	2	81.13	79.41	78.48	71.16	6.97
4	64	4	73.19	73.05	72.24	60.58	7.59
4	32	4	68.67	66.94	66.17	59.08	9.73
4	32	8	68.73	68.3	67.51	60.35	10.11
8	16	16	95.21	93.23	92.09	84.88	14.87
8	16	32	90.62	89.83	88.75	82.17	12.88
8	8	32	61.55	55.85	55.05	46.91	16.33
8	8	64	74.52	69.3	68.36	58.98	21.44
8	512	1	18.8	17.22	16.96	13.23	4.03
8	256	1	20.2	18.22	18.01	14.38	3.44
8	128	2	25.21	23.42	23.15	19.58	4.8
8	128	4	24.68	22.92	22.66	19.02	5.36
8	64	4	36.25	34.52	34.12	30.43	6.97
8	64	8	34.64	34.53	34.13	30.29	7.38
8	32	8	54.18	52.36	51.75	47.79	9.19
8	32	16	53.74	53.58	52.9	48.96	9.9

各ノード毎の最高性能は以下の条件の場合です。

QCD 32*32*32*64 並列実行性能測定結果							
実行時間一覧表(秒)							
node	mpi	smp	total	solver	cg-step	mult	com
1	8	4	217.4	216.63	214.25	168.87	17.29
4	256	1	46.37	44.45	43.94	36.87	5.7
8	512	1	18.8	17.22	16.96	13.23	4.03

- (1) 1ノードではsmp並列よりハイブリッド並列の方が性能がよい。
339秒 vs 217秒
- (2) 4,8ノードではフラットMPIの性能がもっとも良い。
実行には `poe mpibind ./実行ファイル` で
`mpibind` を忘れず入れる必要があります。
- (3) 4,8ノードは1ノードに対しノード数より
高い性能を出している。

2.2 量子モンテカルロ法による物性スペクトル計算

プログラムは高エネルギー加速器研究機構、物構研岩野氏より提供いただいたのをベースにしています。

数値計算からみた場合の計算内容

$G_i, A_i, B, C_i : n \times n$ 行列

$$A_m = C_m C_{m-1} \dots C_1$$

$$A_0 = I \quad (\text{単位行列})$$

$$B = C_L C_{L-1} \dots C_1$$

$$G_m = -A_m (I + B)^{-1}$$

パラメータ: n, L, β, u

$$0 \leq m \leq L, dt = \frac{\beta}{L}$$

C_i の絶対値の最大値, 最小値 $e^{\sqrt{dt \times u}}, e^{-\sqrt{dt \times u}}$

$\text{Tr}(G_m)$ を計算し, 結果のチェックは $\text{Tr}(G_L) = -1$ で行います。

最初の実行結果 (4倍精度)

$n=40, \beta=10, L=112, u=5$ 結果は正確

$n=100, \beta=10, L=448, u=5$ 結果は不正確

目標 $n=100, L=448$ を固定

$\beta=10, 20$ $u=5, 6, 7, 8, 9, 10$ での結果を
10進10桁一致させる。

使用されている計算手法

行列のQDR分解。

行列Vを $V=QR$ (Q:ユニタリ行列,R:上三角行列)に分解後,Rの対角要素をDに入れ,Rの対角要素を1にする。

扱ったケース:設定したパラメータより結果の精度が見積もりやすい。

$$P = e^{\sqrt{dt \times u \times L}}, E(m) = |\text{Tr}(G_m)|$$

プログラム実行中に現れる数値の最大値 P^2 ,0以外の絶対値最小値 $1/P^2$

行列やベクトル要素に現れる数値の最大値 P ,0以外の絶対値最小値 $1/P$

$$E(0) = 1, E(L) = 1, m = \frac{L}{2} \text{で} E\left(\frac{L}{2}\right) = \frac{1}{\sqrt{P}} \text{で最小になるので,}$$

$$\text{演算に必要な最小ビット数} \log_2 \sqrt{P} = \frac{1}{2} \log_2 P。$$

精度検証結果 (10進10桁一致したもの)

6倍精度 $\beta = 10, u = 5, 6, 7, 8, 9$

8倍精度 $\beta = 10, u = 5, 6, 7, 8, 9, 10$

$\beta = 20, u = 5, 6, 7, 8$

10倍精度 $\beta = 20, u = 9, u = 10$

性能測定条件

4倍精度 $\beta = 10, u = 5$

**結果は10進3桁までしか一致しないが
他との比較のため.**

6倍精度 $\beta = 10, u = 6$

8倍精度 $\beta = 20, u = 8$

10倍精度 $\beta = 20, u = 10$

チューニング手順

- (1) 6倍精度, 8倍精度での加減算, 乗算, 除算, 平方根計算をサブルーチン化して, `uinline`オプションを適用
- (2) 演算量の多い行列積計算, QDR積計算部分をサブルーチン化し、並列実行する。
- (3) 加減算, 乗算において, `if`文削除のアルゴリズムに変更。
- (4) (1) ~ (3) の方式を取り入れて10倍精度演算ルーチンを作成。

性能測定結果

量子モンテカルロ実行時間(秒)					
SR16000/M1 1ノード					
4倍精度			10倍精度		
smt on,off	実行時間		QDR積計算	64smp	32core
32core	9.327		並列化不可	733	783
64smp	11.04		分割並列化	426	456
			数値制限解除	451	435
inlineオプション					
精度	サブルーチン化	64smp	32core		
6倍精度	612	263	243		
8倍精度	1190	463	386		
行列積,QDR積計算					
精度	64smp	32core	IF文削除		
6倍精度	112	105	64smp	32core	
8倍精度	192	174	23.743	28.837	
			42.074	51.739	

10倍精度演算の問題

加減算,乗算での演算数の増加

- (1) 最適化,inlineの効果が増少
- (2) 行列積計算での並列化効果の減少
- (3) QDR積計算が並列化不可となり,2つのサブルーチンに分割した事による並列化効果の減少

表現できる数値範囲の制限

大きな数の平方根計算では、アルゴリズムの変更が必要。

2.3 Rump's 例題計算

8倍精度演算方式

QD方式:倍精度変数を4つつなげた方式
DQ方式:4倍精度変数を2つつなげた方式

(注)

- (1) DQ方式はQD方式より精度が悪くなる。
最悪の場合TD (倍精度変数を3つつなげたもの) と同じ精度になる。
- (2) ieee754-2008でのDQ方式の有効ビット数は226ビットで,DD方式のQD方式より精度が良い。

Rump's例題8倍精度演算		
100,000,000回実行の時間(秒)		
SR16000/M1 1ノード実行		
smp数	QD方式	DQ方式
1	99.22	334.01
2	52.95	168.78
4	31.49	86.83
8	36.55	47.55
16	45.68	42.06
32	54.52	47.67
64	49.09	43.3

2.4 6倍精度,8倍精度,10倍精度行列積計算

SR16000/M1 1ノード実行

コンパイルオプションは

-W0,' opt (o (s),disbracket (0)) '

opt (o (ss)) にしたり、disbracket (0) を外すと結果不正となります。

行列積テスト結果一覧表

N=100,1000回実行時間(秒)

精度	single	32core	64smp
6倍精度	46.315	1.877	1.304
8倍精度	80.729	3.285	2.14
10倍精度	188.761	7.58	5.275

- (1) 並列化効果はよくでている。
- (2) 8倍精度はSIMDがよく効いているので他の2つの精度に比べ演算数比率より性能が良い。(8倍精度は4つの倍精度変数を扱うため。)

2.5 3次元反復計算

A : 7点差分に現れる係数.

b : $x^T = (1,1,\dots,1)$ になる様に $Ax = b$ より設定

3次元反復法		SR16000/M1		
		129*129*129 収束判定値 1.0e-12		倍精度
		初期値は $x(i)=0.0$ ($i=1,\dots,n$)		
		実行時間(秒)		
解法	反復回数	single	32core	64smp
bcg	618	24.274	3.571	2.86
cgs	414	16.649	2.721	2.232
scg	618	16.255	1.975	1.799
bicgs		16.339	2.393	2.382
	反復回数	402	395	420
cgsmil		26.125	18.985	19.739
	反復回数	277	292	308
gpbicg		24.897	3.094	2.707
	反復回数	434	381	414
cgsiluq		307.614	159.012	158.055
	反復回数	275	275	275

(注) 10000 は収束しないのでこの回数で打ち切ったもの

257*257*257 収束判定値 1.0e-12				倍精度
実行時間(秒)				
解法	反復回数	single	32core	64smp
bcg		370.112	25.078	26.285
	反復回数	1104	1104	1104
cgs		288.527	17.385	20.083
	反復回数	819	819	820
scg		249.21	16.395	18.881
	反復回数	1104	1104	1104
bicgs		274.517	17.513	19.615
	反復回数	792	776	806
gpbicg		402.435	24.112	26.03
	反復回数	822	754	752
513*513*513 収束判定値 1.0e-12				倍精度
実行時間(秒)				
解法	反復回数	single	32core	64smp
bcg		7779.489	365.732	361.249
	反復回数	2119	2104	2104
cgs		6271.015	346.917	331.985
	反復回数	1644	1900	1900
scg		4947.553	236.644	232.252
	反復回数	2114	2104	2104
bicgs		5473.903	300.964	269.416
	反復回数	1448	1650	1543
gpbicg		6640.65	393.268	406.77
	反復回数	1413	1584	1590

サイズが大きくなると、 x の初期値 $x^T = (0,0,\dots,0)$ では、4倍精度にしても、不完全LU分解を使用するcgsmil,cgsilu法では収束しない。

初期値を[0,1]の一様乱数を取った場合

プログラム	サイズ	反復回数	実行時間(秒)		
			single	32core	64smp
cgsilu	129		22.083	15.124	14.805
		反復回数	234	234	234
	257		341.781	207.416	207.187
		反復回数	446	447	447
	513		5365.806	2967.391	2939.24
		反復回数	806	788	806
cgsmil	257		249.238	150.409	150.486
		反復回数	325	325	325
	513		4195.07	2288.959	2298.282
		反復回数	630	630	630

**xの初期値を変更すると,不完全LU分解を行う
cgsmil,cgsilu法ではサイズを大きくしても
倍精度で収束しています。**

3.BG/Q システム

3.1 QCD計算

チューニングはコンパイラオプションの選択のみ。

テストデータは $32 \times 32 \times 32 \times 64$

並列化はParallel Program Generetor で
作成したソースを使用

実行時間一覧表(秒)						
1ノード						
	1smp	8 smp	16 smp	32 smp	64smp	
total	28954	3756	2347.4	2221.9	2247.6	
solver	28936	3740.1	2332.3	2207	2232.5	
cg-step	28624	3699.5	2307.6	2183.6	2208.9	
mult	24922	3145.1	1794.7	1673.4	1694.3	

**Smp並列では32smpが最高性能で、
実行時間は2222秒**

コンパイルオプション、jcfのチューニングのみ 並列化はParallel Program Generetor で 作成したソースを使用

QCD 32*32*32*64 並列実行性能測定結果						
(BG/Q 32ノード)						
実行時間一覧表(秒)						
mpi	smp	total	solver	cg-step	mult	com
32	16	169.24	168.02	166.07	125.72	35.85
32	32	149.54	148.34	146.57	115.63	28.09
32	64	210.62	208.12	205.59	170.6	31.39
64	8	147.99	145.86	144.2	108.04	41.29
64	16	97.27	96.22	95.08	74.37	23.69
64	32	89.16	88	86.91	70.92	15.6
128	8	85.29	82.98	82.03	59.24	20.82
128	16	61.16	60.02	59.31	45.78	10.89
256	8	60.1	57.12	56.34	41.46	11.43
512	1	109.49	105.1	103.72	64.86	30.37
512	2	77.96	75.92	74.97	48.21	17.96
512	4	56.34	52.79	52.08	35.58	10.44
1024	1	66.93	61.17	60.29	40.96	20.76
1024	2	59.04	53.57	52.88	38.78	11.6
2048	1	66.29	56.16	55.56	39.07	15.71

**最高性能は512mpi,4smpでの56.34秒で
1ノードの39.4倍の性能となっています。**