

課題名 (タイトル) :

生体分子粗視化シミュレーションプログラム CafeMol の動作確認と大規模並列化テスト

利用者氏名 : 木寺 詔紀*

○ 検崎 博生**

所属 : * 和光研究所 次世代計算科学研究開発プログラム
次世代生命体統合シミュレーション研究推進グループ
分子スケール研究開発チーム
** 京都大学理学研究科生物科学専攻生物物理教室

1. 本課題の研究の背景、目的、関係するプロジェクトとの関係

理研を中心に行っている文部科学省プロジェクト「次世代生命体統合シミュレーションソフトウェアの研究開発」において、分子スケールチームに属し、「生体分子系の粗視化シミュレーション技術の開発」（研究代表者、高田彰二（京大理））を行ってきた。そこで、タンパク質、DNA、生体膜の複合系の粗視化分子モデルの分子動力学 (MD) シミュレーションを行うソフトウェア CafeMol を開発中である。

CafeMol は、これまでに、タンパク質、DNA、生体膜の複合系でシミュレーションできるようになっており、分子動力学の時間発展の MPI 並列化と 100 コア規模での並列性能は実証済みである。平成 21 年度では、分子動力学計算の時間発展の openMP によるスレッド並列化とレプリカ交換法の MPI 並列化を実装したので、数千コア規模での並列性能が期待できる状況が整いつつある。

本課題では、CafeMol の大規模並列計算のテストを行い、次世代スパコンにおけるさらに大規模な並列計算のための基礎データとする。具体的には、クロマチン（ヒストン DNA）系のレプリカ交換 MD を例に、8000 コアまでの並列計算のテストを行う。8000 コア規模の並列計算で並列性能が実証できれば、来る次世代スパコン実機での真の大規模計算への準備が進むことになる。

2. 具体的な利用内容、計算方法

テストに使う系としては、ヌクレオソームが 20 個フィラメント状につながった核酸/タンパク質複合体を用いる。ヌクレオソームは DNA がヒストンタンパク質におよそ 2 回転程巻きついたもので、細胞内での DNA の最小構造単位である。そして、ヌクレオソームは集まってフィラメントを形成するが、その構造やダイナミクスは生物的に大変興味を持たれている。

測定は RICC の超並列 PC クラスタを用いて行った。コンパイラは Intel コンパイラを用いた。

3. 結果

OpenMP による並列数は 8 に、レプリカ数は 64 で固定して、分子動力学時間発展の MPI 並列数を 1、2、4、8、15 の 5 つのケースで測定を行った。これは、それぞれ、512、1024、2048、4096、7680 コアの並列計算に相当する。計算のタイムステップは全ての場合で同じで、簡単のために全てのレプリカは同じ物理的条件でのシミュレーションになっている。

測定結果を表 1、2 にまとめる。測定区間については、力場計算の force とネイバリングリスト計算の neighbor、座標の更新の update、エネルギー計算の energy、レプリカ交換の replica、計算結果の出力 output に分かれる。また、force と neighbor、energy、replica については MPI 通信時間により時間がかかるので、演算部分 (ope) と MPI 通信部分 (mpi) に分けている。force の演算部分については、さらに、force(local)、force(nlocal)、force(pnl)

に分けている。このforce演算が最も計算時間が掛かる部分であり、特にforce(pnl)は静電相互作用の計算を含むために、もっとも計算量が多い部分である

加速率は、512コアの実行時間との比で求めている。main_loopの計算時間を見てみると、コア数2倍の1024コアで1.75倍、コア数4倍の2048コアで3.19倍、コア数8倍の4096コアで5.26倍、コア数15倍の7680コアで5.64倍となっている。4096コアまでは並列化により計算速度が上がっているが、7680コアでは頭打ちになっていることが分かる。レプリカ数を64に固定した今回の測定では、分子動力学計算の並列数は512、1024、2048、4096、7680コアの計算でそれぞれ、8、16、32、64、120であるので、64コア程度が分子動力学計算の並列化の限界であると考えられる。

なお、7680 コアでも演算自体の加速度は 10.81 倍と、高速化されているが、MPI 通信に掛かる時間が 3.471 秒と全体の計算時間である 7.159 秒の半分くらいを占めていて、MPI 通信が、分子動力学計算の並列化の限界を決めているといえる。また、この際、レプリカ交換に関する MPI 通信は 0.021 秒と短い。よって、レプリカ数を増やすことによって、並列数を増加させたならば、優れた並列化効率を出すことが期待できる。

表 1：コア数と実行時間

コア数	512	1024	2048	4096	7680
測定区間	実行時間 [sec]	実行時間 [sec]	実行時間 [sec]	実行時間 [sec]	実行時間 [sec]
force	31.086	15.741	8.206	4.932	3.327
force(ope)	30.924	15.468	7.841	3.926	2.196
force(mpi)	0.162	0.274	0.365	1.006	1.13
force(local)	0.153	0.091	0.057	0.048	0.037
force(nlocal)	0.036	0.017	0.008	0.004	0.002
force(pnl)	30.735	15.357	7.727	3.858	2.069
neighbor	1.722	1.145	1.311	0.883	2.341
neighbor(ope)	1.42	0.707	0.349	0.18	0.103
neighbor(mpi)	0.302	0.438	0.962	0.703	2.238
update	0.468	0.462	0.462	0.456	0.462
energy	4.65	2.336	1.201	0.632	0.38

energy(ope)	4.644	2.313	1.172	0.538	0.298
energy(mpi)	0.006	0.023	0.029	0.095	0.081
replica	0.036	0.004	0.032	0.002	0.022
replica(mpi)	0.036	0.004	0.031	0.002	0.021
output	2.253	3.226	1.296	0.632	0.505
ope	39.85	22.312	11.257	5.872	3.688
mpi	0.506	0.739	1.388	1.806	3.471
main_loop	40.356	23.05	12.645	7.678	7.159

表 2：コア数と加速率

コア数	512	1024	2048	4096	7680
測定区間	加速率	加速率	加速率	加速率	加速率
Force	1.00	1.97	3.79	6.30	9.34
force(ope)	1.00	2.00	3.94	7.88	14.08
force(mpi)	1.00	0.59	0.44	0.16	0.14
force(local)	1.00	1.68	2.68	3.19	4.14
force(nlocal)	1.00	2.12	4.50	9.00	18.00
force(pnl)	1.00	2.00	3.98	7.97	14.86
neighbor	1.00	1.50	1.31	1.95	0.74
neighbor(ope)	1.00	2.01	4.07	7.89	13.79
neighbor(mpi)	1.00	0.69	0.31	0.43	0.13
update	1.00	1.01	1.01	1.03	1.01
energy	1.00	1.99	3.87	7.36	12.24
energy(ope)	1.00	2.01	3.96	8.63	15.58
energy(mpi)	1.00	0.26	0.21	0.06	0.07
replica	1.00	9.00	1.13	18.00	1.64
replica(mpi)	1.00	9.00	1.16	18.00	1.71
output	1.00	0.70	1.74	3.56	4.46
ope	1.00	1.79	3.54	6.79	10.81
mpi	1.00	0.68	0.36	0.28	0.15
main_loop	1.00	1.75	3.19	5.26	5.64

4. まとめ

4096 コアまでは、分子動力学の時間発展の並列化とレプリカ交換による並列化によって、速度の向上がみられた。7680 コアに関しては、64 レプリカの場合は速度の頭打ちが見られたが、レプリカ数を増やすことにより、8000~数万コアでも速度の向上が見込まれることが期待できる。ただし、実際の計算では、レプリカ毎

に温度などのパラメータを変化させることによる、レプリカ間の速度のばらつきが問題になってくると思われる。このばらつきの問題をどのように解決するかは今後の課題である。

5. 今後の計画・展望

平成 21 年度では、分子動力学計算のスレッド並列とレプリカ方向の並列化を実装しテストした。今後、数千数万規模の効率的な並列化を実現するためには、レプリカ数を増やす方向での並列化が必須である。この際に問題となってくる、レプリカ間の速度のばらつきを解決することが課題である。また、現状あるソースコードのさらなる効率化も行う予定である。

6. 利用研究成果が無かった場合の理由

ソフトウェアCafeMolの大規模並列化テストが目的であったので、現状では成果発表の段階までいっていないため。

