

超並列計算機 CP-PACS における分子動力学法シミュレーション

服部 正樹† 松原 正純†
板倉 憲一† 朴 泰祐†

本研究では、超並列計算機 CP-PACS 上で分子動力学法シミュレーションを実行し、実測に基づいた評価を行なう。

プログラム作成に当たり、CP-PACS 上で HXB の強力なノード間通信機能を活かし、分子分割法に基づく並列化を行なった。その結果、内部処理を均等に負荷分散することが可能となり、広いノード間通信バンド幅を利用することにより、最大 512 台のノードを用いて十分な台数効果を得ることができた。

Molecular dynamics simulation on CP-PACS

MASAKI HATTORI,† MASAZUMI MATSUBARA,†
KEN'ICHI ITAKURA† and TAISUKE BOKU†

In this research, we implement a molecular dynamics simulator on the massively parallel processor CP-PACS, and evaluate its performance.

Utilizing a powerful interprocessor communication feature of CP-PACS, we parallelize the application based on particle distribution method, which is usually not suitable for mesh-type interconnection network. In this method, we can perfectly balance the load on all processing units and achieve high scalability up to 512 nodes, deriving very wide bandwidth of interconnection network of CP-PACS.

1. はじめに

分子動力学法は材料科学をはじめとする物理・化学の分野で盛んに用いられており、より多くの分子数でのシミュレーションを行なうために、並列化による高速化が望まれている。そこで本研究では、分子動力学法シミュレーションにおける並列化と高速化について、超並列計算機 CP-PACS (Computational Physics by Parallel Array Computer System) に適した手法を考察し、シミュレーションプログラムを作成して CP-PACS 上で実測による実験を行なう。そしてその評価結果から、超並列計算機における分子動力学法の可能性と、CP-PACS における実問題の有効性の 2 点について検討する。

2. CP-PACS

CP-PACS¹⁾ は MIMD 方式の分散メモリ型並列計算機で、2048 台の PU (計算専用プロセッサ) と 128 台の IOU (入出力プロセッサ)、計 2176 台のノードか

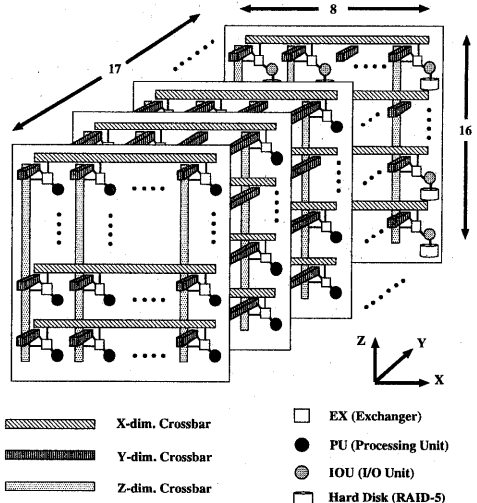


図1 CP-PACS の構成図

ら構成されている。

ノードプロセッサには、Hewlett Packard 社の PA-RISC1.1 アーキテクチャに基づく RISC プロセッサに

† 筑波大学 電子・情報工学系
Institute of Information Sciences and Electronics, University of Tsukuba

PVP-SW (Pseudo Vector Processor based on Slide-Windowed registers) 機構を付加したものを使用している。PVP-SW はキャッシュを用いずにメモリレイテンシを隠蔽する方法を提供し、疑似的なベクトル処理を可能にする。このプロセッサはクロック周波数は300MHzで動作し、また2命令並列実行可能な superscalar 方式を採用している。主記憶容量は64MBである。ライトスルー/ダイレクトマップ方式の2レベルキャッシュを用い、容量は1次が命令用・データ用共に16KB、2次が命令用・データ用共に512KBである。

ノード間の相互結合網には、3次元ハイパクロスバネットワーク (Hyper-Crossbar Network: 以下 HXB) を採用している。3次元 HXB は3次元直交座標上の各格子点にノードを配置し、各次元方向のノードをクロスバスイッチ (XB) で結合したネットワークである。ノード番号が2次元以上異なるノード間では、エクスチェンジャ (EX) と呼ばれる小規模なクロスバスイッチにより、複数の次元の XB を経由してデータを交換する。また、メッセージ転送は wormhole ルーティングを採用しており、ネットワーク中でのメッセージ間のデッドロックを回避するため、固定ルーティング方式を用いている。ネットワークの最大転送スループットは1リンク当たり300MB/secである。

CP-PACS のノード間通信は、リモート DMA 転送と呼ばれる。これは、送信側・受信側共にユーザの仮想アドレス空間の一部を物理メモリ上に固定的に割り当てておき、それらのメモリ間でデータ転送を行う高速通信方式である。異なったノード上の、ユーザ空間同士で直接データ転送を行うため、カーネルとユーザ空間との間でのデータコピーが発生せず、ネットワークのスループットを最大限に発揮することができる。また、リモート DMA 転送では OS を介さずユーザが直接 NIA (Network Interface Adaptor) を起動することにより、通信立ち上げオーバーヘッドを軽減するモードが用意されている。

3. 分子動力学法

分子動力学法 (Molecular Dynamics: 以下 MD) とは、物理法則に従って運動する粒子の集合体が位相空間中に描く軌跡を計算することにより、安定な分子構造や、熱伝導度、拡散係数などの物質の動的な性質を調べる手法である。近年における計算機の急速な性能向上によって、逐次処理では数千個から百万個程度のシミュレーションが行なえるようになったが、現実の物質はアボガドロ数 6.02×10^{23} 程度の分子の集まりであり、それにはまだ程遠い。そこで、並列処理によるさらなる高速化が期待されている。

以下では本研究における MD の方法を述べる。

分子間に働く力は、ポテンシャル関数で簡略化した分子間相互作用として与える。ここでは式 (1) で表

されるレナード・ジョーンズの2体間ポテンシャル²⁾を用いる。

$$\phi(r) = 4\epsilon \left\{ \left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right\} \quad (1)$$

r は分子間距離、 ϵ, σ は物質によって異なる定数である。

シミュレーションは単純化のために立方体の MD セルと呼ばれる箱の中で行なわれる。その際、セルの壁と分子との相互作用を無視できるように、周期境界条件を用いて疑似的に無限に大きい体積空間を作る。また、分子間の相互作用は、最近接鏡像法に従って計算する。

次に分子運動のアルゴリズムについて述べる。

本研究では、全エネルギーが保存する系、すなわちマイクロカノニカル・アンサンブルの MD を行なう。そこで verlet の速度形式²⁾と呼ばれるアルゴリズム (式 2, 3) を用いることにする。

$$r_i^{n+1} = r_i^n + v_i^n \Delta t + \frac{1}{2m} F_i^n (\Delta t)^2 \quad (2)$$

$$v_i^{n+1} = v_i^n + \frac{1}{2m} (F_i^{n+1} + F_i^n) \Delta t \quad (3)$$

r_i^n, v_i^n, F_i^n はそれぞれステップ n における分子 i の位置と速度と力を、 Δt は時間さざみ幅を表す。これにより初期値 r_i^1, v_i^1 を与えて、時間発展的に計算を繰り返せば、各ステップにおける位置と速度が次々に求められる。

初期値を求めるには、まずもっともらしい初期条件を課して、その後、エネルギーを足したり引いたりして設定値に持っていく。これは verlet の速度形式のアルゴリズムでは、全ての分子の速度を一斉に調整する、速度スケールリングによって行なうことができる。

4. シミュレーションの手法

本節では、本研究で行なう CP-PACS における並列 MD シミュレーションの最適化手法について説明する。

4.1 ポテンシャルのカットオフ

MD シミュレーションでは、分子数 N が大きくなると分子対の数が急激に増大し、何ら特別な工夫を行わなければ、1ステップにおける全計算時間のほとんどがポテンシャルの計算に使われてしまう。そこで、ポテンシャルエネルギーの計算コストを縮小するために、精度を確かめた上で特定の分子間距離以上の相互作用をカットする手法を用いる。

レナード・ジョーンズポテンシャルのように分子間距離が増大すると相互作用の大きさが急速に0に近づくポテンシャルでは、固定距離でのカットが有効となる。(1)式によると、 $r = 2.5\sigma$ の時に約 $-1.63 \times 10^{-2} \epsilon$ となり十分0に近づくので、通常 r_c の値には $2.5\sigma \sim 3.5\sigma$ が用いられる。

4.2 並列化

MD では、分子一つ一つは、ある時刻ステップ内に

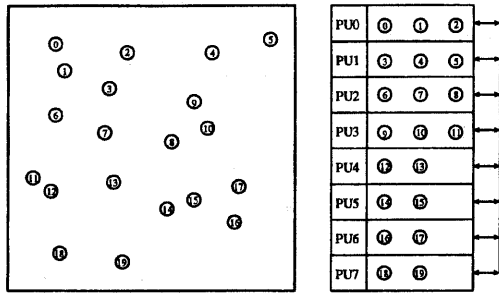


図2 分子分割法概念図

限っては独立に処理できるため、並列化に非常に向いていると言える。並列化の方法としては、処理の分割の方法から空間分割法と分子分割法の2つに大別できる。

以下にそれぞれの概念を説明する。

4.2.1 空間分割法

空間分割法は、MDセルの空間をメッシュ状に分割して各PUに割り当てる方法である。各PUは割り当てられた部分空間（部分セル）の中に存在する分子についてのみ、運動の計算を行なう。

空間分割法で必要となるPU間通信は、主に下の2種類に分けられる。

- ポテンシャル計算時の自分の部分セルの外にある分子の位置情報の参照
- 分子が部分セルを出入りすることによる分子の受渡し

ここで部分セルの一辺の長さ l に対してポテンシャルのカットオフ距離を $r_c < l$ とすれば、通信を大幅に限定することができる。特に、Mesh型のPU間ネットワークを持つ並列計算機や、HXBを持つCP-PACSでは、3次元空間をそのままマッピングすることができるので、どちらの通信も隣接PU間通信となり、非常に有効である。

ところが、空間分割法ではPU毎の計算負荷が変動する可能性が指摘されている。特に低温、低密度の物理状態のシミュレーションを行なった場合、分子が空間で局所的に集中する可能性がある。したがって、空間分割法では如何に動的負荷分散を行なうかが問題となってくる。

4.2.2 分子分割法

空間分割法では空間を基本に分割するのに対し、分子分割法では分子を基本に分割する。分子分割法概念図を図2に示す。図のように、分子の位置を考慮せず各PUに均等な数の分子を配置する。各PUは自分が持つ分子についての運動を計算する。PU間での分子の移動は基本的にはない。

分子分割法で必要となるPU間通信は、主に下の1種類だけである。

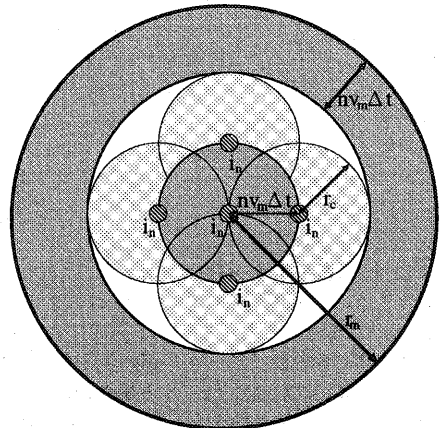


図3 時間間引き法概念図

- ポテンシャル計算時の自分以外のPUの持つ分子の位置情報の参照

ポテンシャルのカットオフを行なう場合、通信の相手を限定することもできるが、限定範囲が動的に変化するため好ましくなく、基本的には全対全転送を行なった方が通信パターンを固定できるので都合が良い。

分子分割法では、位置に関わらず分子をPUに分配することができ、さらにシミュレーション中にも各PUで処理を行なう分子数は固定されるので、計算負荷を均等に分割することができ、空間分割法のような問題はない。

分子分割法は大規模・大域的な転送パターンを必要とするため、これまでは敬遠されがちで、並列MDの研究では、空間分割法で如何に動的な負荷分散を行なうかが主題であった³⁾。しかし、CP-PACSでは大規模かつ大域的なデータ転送をその強力な通信性能でカバーできるため、計算負荷が均等に分散できる分子分割の方が向いていると考えられる。また、ポテンシャルのカットオフを行わないMDも、同様の手法で行なえるため有利である。そこで本研究では、分子分割法によって並列化を行なう。具体的には、分子分割法で必要となる全対全転送を、Butterfly Collection アルゴリズムを用いて効率良く行なう⁴⁾。

4.3 時間間引き法

衝突する分子同士をある程度予測する方法として、bookkeeping法⁵⁾という方法が提案されている。ここでは、bookkeeping法を発展させた方法として、時間間引き法を提案する。

図3に時間間引き法概念図を示す。

時間間引きのステップを n とする。まず、この n ステップ内に注目する分子 i がどの範囲に動く可能性があるかを考える。一般化するために、シミュレーション中の全分子における最大速度を v_m とすると、 n ス

ステップ間に分子 i と相互作用を及ぼす可能性のある分子が最初のステップにおいて存在する範囲は i_0 から、

$$nv_m \Delta t + r_c + nv_m \Delta t = 2nv_m \Delta t + r_c \quad (4)$$

の半径の球内にある、と限定することができる。 Δt は時間幅を、 r_c はカットオフを表す。これを r_m とする。

そこで、各分子について半径 r_m の球内にある分子番号を表に記録しておけば、その時点から n ステップの間は表中にある分子だけを対象にカットオフ判定を行なうようにすることにより、判定のための距離計算をある程度絞ることができる。そして、その表は n ステップ毎に書き換えられることになる。

以上が時間間引き法概念である。この方法は、テーブルのサイズがある程度小さく、テーブルのデータをキャッシュに載せることができれば、RISC プロセッサ上では有効に働くが、テーブルが大きくなるとキャッシュミスによって性能が低下する場合がある。また、テーブルを引いて分子情報をアクセスするため、リストベクトル処理となり、同様にキャッシュミスによる欠点を持っている。しかし CP-PACS では、PVP-SW が有効に機能すれば、この欠点を克服でき、効果的に性能向上を図ることができると考えられる⁴⁾。

5. 評価及び考察

本節では、MD シミュレーションを CP-PACS 上で実行し、各手法に対し評価及び考察を行なう。まずシミュレーションの各条件を示し、次に評価の方法を述べ、その後実験結果を示して考察を行なう。

5.1 シミュレーションの条件

単原子液体の液体アルゴンをエネルギー一定の分子動力学法でシミュレートする。各条件は無次元単位で、時間刻みが 0.064、系の平均温度が 2.53、そしてシミュレーション総ステップ数は 2000 とする。このうち、1000 ステップまで、50 ステップ周期で速度スケールングを行なう。

5.2 評価の方法

上で述べた条件に従い、CP-PACS 上で動作する MD シミュレーションプログラムを作成する。プログラムは Fortran90 言語で記述し、通信は全てリモート DMA 転送を使用する。

シミュレーションを実行し、その上で elapse 時間を測定する。測定範囲は、データを初期化した後から、シミュレーションを開始して結果を格納するまでとする。

5.3 ポテンシャルカットオフの評価

ここでは、ポテンシャルのカットオフがどの程度計算時間に影響を与えるのかについて考察する。

表1は、分子数 256 の場合でカットオフを変えた時の 1PU での逐次実行時の計算時間の変化を示している。

$r_c = 2.5$ の時には、ある 1 個の分子に相互作用を与える分子数は平均 42 個になり、 $r_c = 3.5$ の時には平均 114 個になる。全分子のポテンシャルの計算は相互

表1 カットオフ距離の違いによる計算時間の変化 (sec)

	$r_c = 2.5$	$r_c = 3.5$	no cutoff
全実行時間	31.473	40.636	42.012
力の計算時間	4.294	10.845	36.969
カットオフの計算時間	22.062	23.800	0

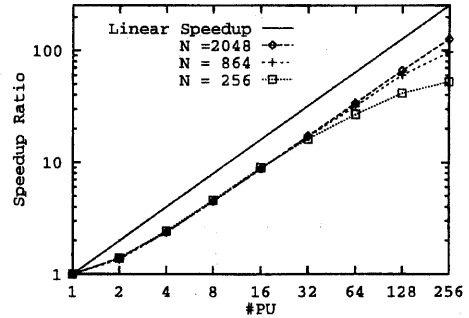


図4 分子数の違いによる速度向上率の変化

作用を与える分子の組合せの数に比例するので、ポテンシャル計算に要する時間はカットオフなしの場合と比べて、 $r_c = 2.5$ のときには約 1/6、 $r_c = 3.6$ の時には約 1/2 弱になるが、実際はカットオフ時に距離をすでに計算しているもので、それ以下の値になっている。

ところが、カットオフのオーバーヘッドが大きいため、全計算時間の削減にはあまり反映されておらず、特に $r_c = 3.5$ の時にはカットオフなしとそれほど変わらない。

5.4 並列化の評価

次に、分子分割法による並列化の評価を行なう。

まず、分子数 256・864・2048 で、PU 数を 1~256 に増やしていった時の実行時間の変化を表 2 に、その時の速度向上率のグラフを図 4 に示す。

表2 分子数と PU 数の違いによる実行時間の変化 (sec)

		$N = 256$	$N = 864$	$N = 2048$
#PU	1	31.473	275.599	1462.727
	2	22.579	202.532	1047.553
	4	12.897	115.982	615.708
	8	6.822	61.174	325.554
	16	3.508	31.453	167.032
	32	1.943	16.191	85.174
	64	1.171	8.642	43.226
	128	0.759	4.559	22.018
	256	0.601	2.852	11.524

速度向上率を見ると、 $N = 256$ の 256PU で頭打ちになっているもの以外は、2PU を基本に見るとかなり良い速度向上率が得られている。しかし、1PU から 2PU への速度向上が低く、全体としてあまり良いよう

に見えない。これはあるPU内にある分子とその他のPUにある分子との相互作用を計算する時、逐次実行ならば1回で済む計算を、並列実行では各PUで別々に計算するため、2度手間になっているためである。

次に、実行時間を内部処理時間と通信時間に分けて見てみる。分子数256・2048で、PU数を1～256に増やしていった時の内部処理時間と通信時間の変化を表3に示す。

表3 分子数とPU数の違いによる内部処理及び通信時間の変化 (sec)

		N = 256		N = 2048	
		内部	通信	内部	通信
#PU	1	31.473	0	1462.727	0
	4	12.832	0.065	615.429	0.279
	16	3.423	0.085	166.672	0.360
	64	1.070	0.101	42.841	0.385
	256	0.486	0.115	11.120	0.404

表3から、全実行時間に占める通信時間の割合はいずれもごく小さいことが分かる。分子数256の256PUでも約22%と、粒度を考えるとそれほど大きいものではない。したがって、通信時間は並列化効率にはそれほど影響を与えておらず、前述した力の計算の2度手間によるオーバーヘッドが大きいと考えられる。

また、ノード数の増加に対して通信時間の増加が緩やかであることが分かる。ここで行なう通信は全てButterfly Collectionで、ノード数Pに対して、転送回数は $O(\log(P))$ で増加し、また1回の転送量は $O(1/P)$ で減少する。HXBではこのButterfly Collectionを無衝突で行なうことができるため、非常に効率の良い通信となっている⁴⁾。

次に、計算負荷の分散を見るために、各PU毎に実行時間を見る。厳密な測定を行なうために、各PU毎の1シミュレーションステップの内部処理時間をCPUクロック単位で測定する。分子数256で測定し、結果からPU間での平均・最大・最小値、及び標準偏差を計算し、表4に示す。表中の \bar{T} ・ T_{max} ・ T_{min} ・ σ はそれぞれ平均値・最大値・最小値・標準偏差を表す。なお、標準偏差は平均値との相対値で表記する。

表4 計算負荷の分散状況 (CPU サイクル数)

		N = 256			
		\bar{T}	T_{max}	T_{min}	σ
#PU	2	1667827	1687038	1648616	0.0115
	4	930036	971309	904426	0.0294
	16	251939	258205	246731	0.0123
	64	74667	79596	70130	0.0284
	256	22888	25688	20539	0.0416

表4より、最大でも±10%のばらつきで済んでいるのが分かる。

5.5 時間間引き法の評価

次に、時間間引き法による最適化の評価を行なう。分子数256・2048、カットオフの距離 $r_c = 2.5$ 、1～256PUで、(4)式における間引きステップ数 n を $5 \cdot 10 \cdot 15$ に変えた時と、間引きを行なわないものとの実行時間の比較を行なう。

理論的な r_m の値、すなわち $2nv_m\Delta t + r_c$ の値と、実際の r_m の最小値とは異なった値となる。理論的な r_m の値は、分子が n ステップの間常に最高速度で直線的に運動した場合のもので、実際には最高速度以下の速度で非直線的な運動をするため、実際には r_m の値は理論値より小さくても良い。そこで、間引きなしの結果と変わらない範囲で r_m を0.1単位でどこまで小さくできるか実験し、それによって求められた r_m の最小値を使用する。

5.5.1 分子数256の結果

分子数256の実行結果を表5に示す。なお、表中で下線を引いたものがそのPU数における最も良い値である。

表5 分子数256の時の実行時間 (sec)

		間引きステップ数			
		なし	5	10	15
#PU	1	31.473	22.509	22.868	25.416
	2	22.579	14.645	14.901	16.822
	4	12.897	8.152	8.351	9.599
	8	6.822	4.261	4.347	4.923
	16	3.508	2.338	2.335	2.632
	32	1.943	1.356	1.338	1.482
	64	1.171	0.876	0.857	0.922
	128	0.759	0.625	0.615	0.658
	256	0.601	0.521	0.527	0.540

全体的な時間間引き法の効果としては、ここでは1.1～1.4倍程度の性能向上として現れている。

間引きステップ数が小さすぎると、 r_m を小さくできるのでカットオフのオーバーヘッドが小さくなるが、テーブルの更新が頻繁に起こって効率が悪くなる。一方、ステップ数が大きくなりすぎると、テーブル更新の頻度は下がるが、 r_m が大きくなりカットオフの効率が悪くなる。このように、間引きのステップ数と r_m はトレードオフの関係にあると言える。分子数256の結果では、1PUの実行時間を見ると $n=5$ の場合が最も効率良くトレードオフできているようである。256PUでは粒度が小さくなり過ぎて並列化効率が頭打ちになっており、間引きステップによる差はそれほど見られない。

5.5.2 分子数2048の結果

分子数2048の実行結果を表6に示す。

時間間引き法の効果は、3.0～5.7倍の性能向上と、かなり高い効果が得られている。また、粒度も十分のため256PUでも十分な効果が出ている。

表6 分子数 2048 の時の実行時間 (sec)

		間引きステップ数			
		なし	5	10	15
#PU	1	1462.727	474.034	362.773	345.556
	2	1047.553	296.805	210.395	197.639
	4	615.708	181.161	131.556	124.307
	8	325.554	91.762	64.705	60.205
	16	167.032	47.433	33.398	30.567
	32	85.174	24.025	16.596	15.634
	64	43.226	12.469	8.667	8.070
	128	22.018	6.681	4.741	4.386
	256	11.524	3.803	2.852	2.672

最適な間引きステップ数は 15 程度となった。

5.6 分子数 16384 の結果

最後に、格子のサイズ $S = 16$ 、分子数 $16 \times 16 \times 16 \times 4 = 16384$ 、ステップ数 2000 のシミュレーションを、PU 数 64 ~ 512 で実験を行ない、その結果を示す。なお、カットオフの距離 $r_c = 2.5$ 、時間間引きステップ数 $n = 10$ 、 $r_m = 3.5$ でシミュレーションを行なった。結果を表 7 に示す。

表7 分子数 16384 の結果 (sec)

#PU	64	128	256	512
実行時間	304.483	152.476	78.798	41.883

この結果を見ると、CP-PACS の 2048PU 全てを使用すれば、分子数 16384 のシミュレーションは 10sec 程度で行なうことができ、分子数 $20 \times 20 \times 20 \times 4 = 32000$ の 2000 ステップのシミュレーションでも数分で終了することが予想される。また、分子数を増やせば増やすほど並列度が上がるため、分子数が多いほど並列計算向きであると考えられる。2048PU 以上の超並列計算機で、より多くの分子数のシミュレーションを行なうことは、非常に有意義であると思われる。

6. 結 論

本研究では、超並列計算機 CP-PACS 上で分子動力学法シミュレーションを行なうプログラムを作成し、実行及び評価を行なった。

評価結果から、CP-PACS では分子分割法に必要な全対全通信を非常に効率的に行なって通信時間を最小限に押さえることができ、また計算負荷を均等に分散させることができることが分かった。ところが、並列化によって内部処理にオーバーヘッドが生じ、これが台数効果を妨げる原因となった。ただし、2PU 以上だけで見ると顕著な台数効果が見られた。

最適化に関しては、分子分割法においては、ポテンシャルのカットオフ単体だけではカットオフ判別のオーバーヘッドが大きいためそれほど効果を発揮せず、時間間引き法を併用すると著しく性能を向上でき

ることが分かった。

分子数 16384 の MD を 512PU で行ない、32000 といったさらに多くの分子数でも数分でシミュレーションできることが予想できた。

全体としては、分子動力学法シミュレーションが並列化に非常に向いており、分子数を増やせば増やすほど並列度が上がるという、典型的な超並列計算機向けの問題であることが分かった。また、CP-PACS のアーキテクチャは分子分割法による並列化が適しており、長距離相互作用などを含めたより深いシミュレーションを行なうことが容易に可能であると考えられる。

今後の課題として、

- 長距離相互作用、多原子分子間ポテンシャル等を採り入れ、さらに深いシミュレーションを行なう。
- より多くの分子数でシミュレーションを行なう。
- 空間分割法による並列化の評価を行なう。
- 時間間引き法に伴う誤差の解析を行なう。
- 時間間引き法の v_m を計算ステップ中で動的に求める。

などが挙げられる。

以上の課題に対してさらなる追求を行なうことによって、CP-PACS はもとより、超並列計算機の科学技術問題に対するさらなる可能性を見出すことができると思われる。

謝辞 本研究に関し貴重な御意見を頂いた、電気通信大学中澤喜三郎教授、東京大学中村宏助教授ならびに筑波大学電子・情報工学系アーキテクチャ研究室並列処理研究グループ諸氏に深く感謝します。なお、本研究の一部は文部省創成的基礎研究費 (08NP0401) の補助によるものである。

参 考 文 献

- 1) 岩崎洋一, 中澤喜三郎ほか: 計算物理学と超並列計算機 — CP-PACS 計画 —, 情報処理, Vol. 37, No. 1, pp. 10-42 (1996).
- 2) 小澤哲, 篠嶋安記: シミュレーション物理学, シュプリンガー・フェアラーク東京 (1990).
- 3) 林亮子, 堀口進: 並列分子動力学法シミュレーションにおける動的負荷分散法, JSPP'96 論文集, pp. 81-88 (1996).
- 4) 板倉憲一ほか: 超並列計算機 CP-PACS の基本性能評価, 情報処理学会研究報告, ARC123-4, pp.19-24(1997).
- 5) 田中実, 山本良一: 計算物理学と計算化学, 講談社 (1994).