

Linux Alpha クラスタを用いた放射線治療計画の高速化

佐藤裕幸^I, 中川隆文^{II}, 依田潔^{III}, 中島克人^I,
坂本豪信^{III}, 土谷昌晴^{III}, 遠藤真広^{IV}

I 三菱電機 (株) 情報技術総合研究所, II 三菱電機 (株) 先端技術総合研究所,
III 三菱電機 (株) 通信機製作所, IV 科学技術庁 放射線医学総合研究所

遠隔地の複数の医療機関から重粒子線がん照射施設の計算サーバを利用して患者の治療計画の作成を可能にするために、高性能な Linux Alpha を汎用のネットワークで接続したクラスタを用いて、線量分布計算等を並列化して放射線治療計画を高速に処理するシステムの研究開発を行っている。この並列化は計算領域を分割することにより行い、データ圧縮等によりプロセッサ間の通信時間を抑え、また、各プロセッサに複数領域を担当させることで負荷を均等化している。実験の結果、5 台で 4.18 倍の高速化を確認することができた。

Speed-up of Radiation Treatment Planning using a Linux Alpha Cluster

Hiroyuki Sato^I, Takafumi Nakagawa^{II}, Kiyoshi Yoda^{III}, Katsuto Nakajima^I,
Hidenobu Sakamoto^{III}, Masaharu Tsuchiya^{III}, Masahiro Endo^{IV}

I Mitsubishi Electric Corp. Information Technology R&D Center,

II Mitsubishi Electric Corp. Advanced Technology R&D Center,

III Mitsubishi Electric Corp. Communication Systems Center,

IV National Institute of Radiological Sciences

To provide a more highly efficient radiation treatment planning server at the heavy ion treatment facility for cancer therapy, we make the dose calculation and the digitally reconstructed radiograph more high speed using a network parallel processing system including 10 alpha21164 CPUs. Calculation area is divided and each processor takes charge of more than one part to balance a load. Communication data is compressed to save communication time. By experimentation, about 4.18 times speedup is performed with 5 CPUs and effect of data compression at data communication is found.

1 はじめに

粒子線治療は、ヘリウム、炭素、ネオンなどの重粒子線を患部に集中して照射し、周辺の正常細胞への影響を最小限に抑え、患者の早期社会復帰を可能とする非常に有効ながん治療方法である [1] [2]。この粒子線を用いたがん照射治療を普及させるためには、遠隔地の複数の医療機関から重粒子線がん照射施設の計算サーバにより、照射施設のノウハウを利用して患者毎に最適な照射方法を得る治療計画を立案できることが望ましい。この治療計画では、効率的な治療を行うために、患者体内の線量分布を正確に計算しなければならない。しかし、従来の高精度な三次元の線量分布計算は処理時間がかかるため、治療計画の効率が悪いとい

う問題点があった。

近年、パーソナルコンピュータ (PC) の低価格化とギガビット・イーサネットを代表とする汎用ネットワークの高速化により、PC やワークステーション (WS) を構成要素とする並列処理クラスタが、安価で拡張性の高い計算サーバとして注目されている [3]。そこで我々は、計算サーバの処理能力及び信頼性等を向上させるために、特に浮動小数点演算性能に優れた Alpha WS を高速な汎用ネットワークで接続したクラスタを用いて、並列処理により、治療計画処理の高速化を行っている。

この並列化においては、CT 画像等の大量データを扱うので、分散メモリ環境でデータ通信時間をいかにして抑えるかが大きな課題となる。

また、領域分割により並列化を行うが、各領域の計算負荷の予測が困難なため、プロセッサ間の負荷の均等化も課題の一つとなる。

本稿では、この並列処理システムの概要と初期実験結果について報告する。

2 線量分布計算の並列化

線量分布は、患部周辺の平行横断面である CT 画像を複数枚（～100 枚）取得することで得られる三次元情報（図 1）を基に計算する。1 枚の CT 画像は 512×512 ピクセルであり、1 ピクセル当たり 4 バイトで表現しているため、その三次元データ量は 0.5K ピクセル×0.5K ピクセル×100 枚×4 バイト＝100M バイトと大量となる。線量分布計算では、この三次元化された CT 画像データや各種照射に関するデータを基にして、各点の吸収線量を計算するので、計算結果も 100M バイトのデータ量となる。このように、線量分布計算では、大量データを扱うので、並列化の際に、データの通信時間をいかにして抑えるかが課題となる。

線量分布の計算は、以下のような手順で行う。

- 照射される面を格子状に分割し、ビームを各格子への線 (Ray) に分割する。図 2 では、3×3 の 9 つの Ray に分割している。
- 各 Ray 毎に Ray 上に沿って一定間隔に計算点を決める（図 2 の星印）。
- 各計算点毎に、前計算点の計算結果、計算間隔（距離）、計算点上の CT 値等を基に線量を算出する。

なお放射線としては、陽子及び炭素を対象としている。この方式による線量分布計算は、通常の WS を用いて数十秒程度かかり、更にペンシルビーム法[4]と呼ばれるより正確な結果が得られる方式では更に時間がかかる。

計算点毎の処理は前計算点の結果を利用してするため並列化が困難であるが、Ray 毎の計算は独立しているため Ray 束の単位で並列に処理する。つまり、計算領域を複数の部分領域に分割し、それぞれの分割された領域を各プロセッサで独立に計算し、計算結果を最後にマージするという並列化方法である。各部分領域はその CT 値により計算時間が異なるため、部分領域によって負荷が異なる。従って、並列化においては、各プロセッサ間で負荷をいかにして均等化するかということも課題となる。

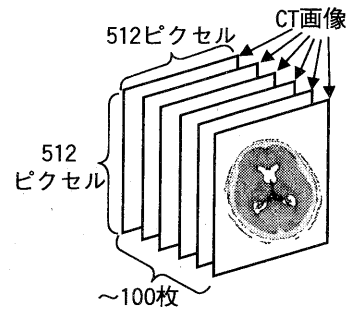


図 1 三次元 CT 画像

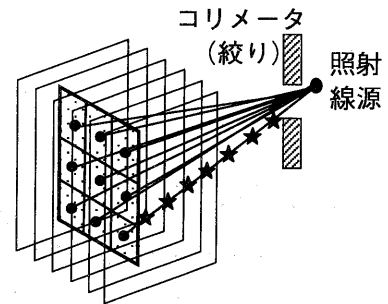


図 2 線量分布計算

また、治療計画においては、CT 画像を基に照射線源から照射体を見て撮影したような画像を再構成する DRR (Digitally Reconstructed Radiograph: CT 再構成シミュレーション画像) の生成も処理時間がかかるので、線量分布計算と同様の方法で並列化する。

3 並列処理システム

3.1 システム構成

治療計画の計算処理システムは、10 台の WS を汎用のネットワークで接続した、いわゆる WS クラスタである（図 3）。各 WS の CPU は Alpha21164A 600MHz であり、10 台の内 1 台がサーバとして使用される。サーバには 1GB のメモリが搭載され、それ以外のマシンには 512MB 搭載されている。各マシンは、100Mbps のイーサネットワーク・スイッチで接続されている。将来は、1 Gbps のイーサネットワークが安価になり、VIA (Virtual Interface Architecture) [5] 等の高速通信技術が容易に利用可能になれば、積極的に採用していく予定である。このクラスタが、遠隔地の医療機関からギガビットレベルのネットワークを介して利用されることを想定している。

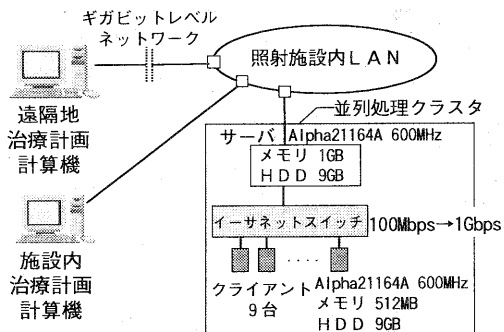


図3 システム構成

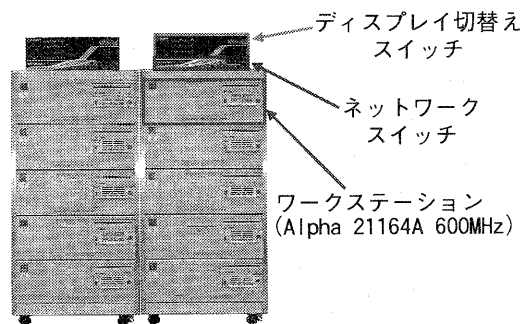


図4 並列処理クラスタ

図4に並列処理クラスタの実物写真を示す。このクラスタは、2セットのデスクトップ型のWSを5台積み重ねたものから成り、1台数十万円のWSとネットワーク機器等で、総計数百万円という安価かつ高性能で拡張性のある並列処理環境となっている。

3.2 ソフトウェア構成

図5に、並列処理クラスタのソフトウェア構成を示す。オペレーティングシステムは、Linuxを用い、プロセッサ (WS) 間の通信ライブラリは、多くの種類のマシン上で動作し広く利用されているPVM (Parallel Virtual Machine) [6] 及びMPI (Message Passing Interface) [7] を用いる。

並列処理を支援するツール (ミドルウェア) として、我々の開発した分散型並列処理支援ツール ParaJET [8] を使用する。ParaJET は、あらかじめ指定されたジョブ群を各プロセッサの負荷状況に応じて分配する負荷分散ツールである。特に並列動作するジョブ間で全く通信がなく独立して実行できる場合は、従来の逐次プログラムを変更することなく並列処理で

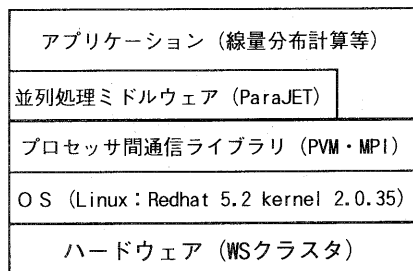


図5 ソフトウェア構成

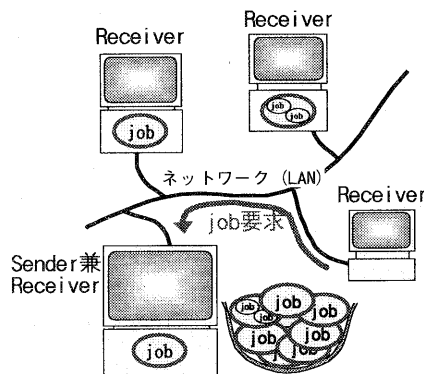


図6 ParaJETの負荷分散方式

きるという特長を持っている。また、各プロセッサの負荷状況と実行状態 (実行中及び実行を完了したジョブ群) を表示するモニター機能を持っており、粒度調整等の並列処理のチューニングが容易に行えるようになっている。

ParaJETの負荷分散方式は、いわゆる要求駆動型の動的負荷分散である (図6)。Senderマシンにジョブ群をプールしておき、実際にジョブを実行する各Receiverマシンは実行するジョブがなくなったら (又はなくなりそうになったら)、Senderマシンにジョブの要求を行う。なお、SenderマシンはReceiverマシンを兼ねても良い。

4 実験

ここでは、計算モデルを簡略化した主に並列処理システムの機能検証のための実験 (予備実験) と、本格版が動作し始めているのでその初期評価実験について報告する。

4.1 予備実験

4.1.1 計算モデルと並列処理方式

この実験で用いた計算モデルは、400mm×400mm×100mmの水で満たされた円柱の中

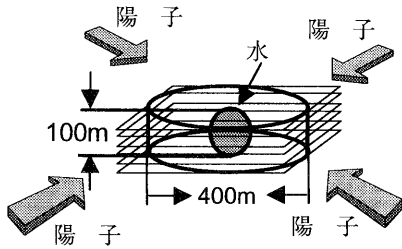


図7 計算モデル (4方向照射)

に腫瘍形状を模擬する直径 100mm の球があり (球の媒質も水)、複数方向から等角度間隔に球ターゲットに陽子線を照射した際の線量分布を計算する¹ (図7)。基となる CT データは媒質が均質な円柱なので各マシン内で計算により生成し (データ転送はしない)、線量分布結果は 400×400×25 ピクセル (断面のピクセルサイズが 1mm でスライス間隔が 4mm) として計算する。ここでは 1ピクセルを 4バイトで表現しているので、計算結果のデータ量は、0.4K ピクセル×0.4K ピクセル×25 枚×4バイト=16M バイトとなる。

この実験では実現容易性を考慮して、Ray を分割するのでなく、以下のように照射方向毎に従来の逐次プログラムを用いて線量分布計算を行い、最後に計算結果をマージする。

- 照射方向毎の線量分布計算プログラムを ParaJET の下で起動し、同時にデータ収集プログラムも起動する。
- 線量分布計算プログラムは ParaJET により最適な WS で実行され、計算結果を PVM によりデータ収集プログラムへ送信する。
- データ収集プログラムは、データが送られる毎に各ピクセルの総和を取り、全ての照射方向からのデータが揃ったら結果をディスクに書き込んで終了する。

4.1.2 実験結果とその考察

10 方向から照射した場合の線量分布計算の実行時間を図8に示す。各線量分布の計算過程では前述したように 4 バイトの floating で行っているが、最終的な精度としては 2バイト程度でも十分である。そこで、データ転送時間を削減するために、転送時にデータ圧縮する場合も計測した。計測の結果、データ圧縮による通信の高速化の効果が若干ではあるが確認で

¹ このように複数方向から照射する場合を多門照射と呼び、実際の治療でも行われている。

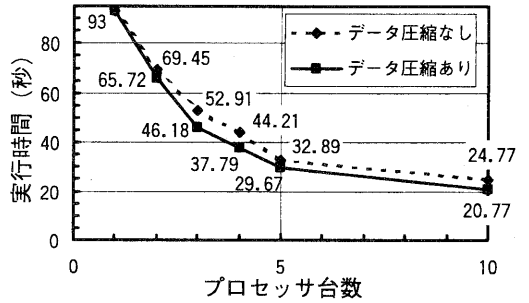


図8 並列実行時間 (10方向照射)

き、10台では1台の時に比べて約4.5倍の高速化 (データ圧縮時) となっている。

ここで、この4.5倍という結果がプロセッサの台数比例の10倍よりかなり低くなっていることについて考察する。

各処理の実行時間を個々に計測すると、図9のようになる。線量分布計算は照射方向によりその実行時間が異なっており、最も短いのが5秒、次に7.5秒、10.5秒が2方向あり、それ以外の方向は12.7秒であった。線量分布計算を最も長い12.7秒とし、データ転送がサーバ以外の9台で順番に逐次にしか行われないとすると、それらを合計した値 (全実行時間) は23.45秒となる。これは実測値の20.77秒より長い。これは、線量分布計算の実行時間が照射方向により異なっているため、一部の線量分布計算とデータ転送が同時に行われているからだと考えられる。つまり、図10に示すように、照射方向によって早く計算が終了するプロセッサがいるため、他のプロセッサの計算が終了する前にデータ転送が始まっている。このようにクライアント2の10.5秒の線量分布計算終了か

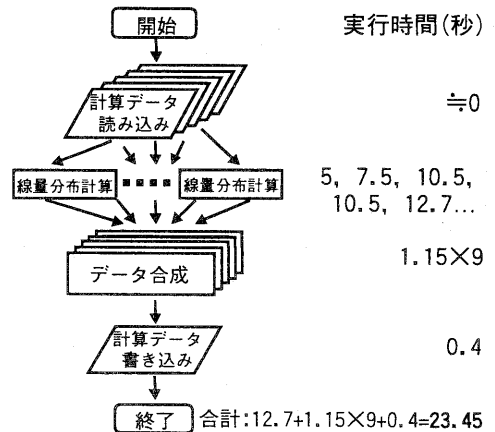


図9 並列処理の流れと実行時間

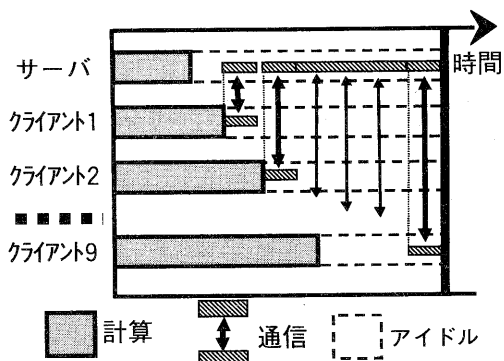


図 10 計算と通信の流れ

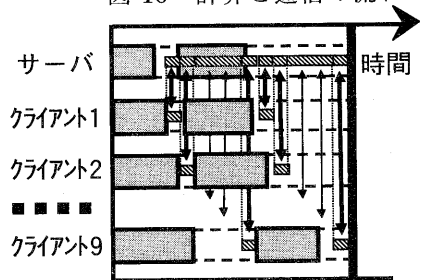


図 11 領域の多分割による計算と通信のオーバーラップ化

らデータ転送が逐次で行われているとすると、全体の実行時間は $10.5 + 1.15 \times 8 + 0.4 = 20.1$ 秒となり、実測値の 20.77 秒に近づく。

いずれにしても図 10 を見れば分かるように、サーバへのデータ転送がボトルネックとなり、アイドルの部分が大きくなっているため、10 台を使用しても 4.5 倍程度しか高速化されなかったと考えられる。このデータ転送のボトルネックを解消する方法として、データ転送を二分木状に行うことにより逐次に行われる通信の回数を対数オーダーにすることが考えられる。しかし、実験の結果、データ圧縮なし：20.56 秒、データ圧縮あり：18.27 秒となり、データ転送回数の半減という予想したほどの効果は見られなかった。

以上のことから、本格版においては、以下のことを考慮しなければならない。

- データ転送がネックとなることが明白なので、転送データ量を削減する。
- 領域毎の負荷のアンバランスの解消とデータ転送と計算実行をできるだけ同時に行えるように²、図 11 に示すように領域をプロ

² データ転送時にも CPU を使用するので計算とデータ転送は完全には同時に行えない。

セッサ台数以上に分割し、各プロセッサに複数の領域を担当させる。

4.2 初期評価実験

4.2.1 計算モデルと並列処理方式

この実験で用いた計算モデルは、ほぼ実際の放射線治療に近い形であり、256mm×256mm×200mm の水で満たされた立方体に直径 40mm の球（球の媒質も水）があり、線源側に直径 40mm のコリメータ（絞リ）と球の表面に均一線量を照射するために半径 20mm の半球状にくり貫かれた補償体（放射線吸収体）がある（図 12）。計算の基となる電子密度データ及び線量分布結果は、512×512×100 ピクセルであり、1 ピクセル 4 バイトで表現しているので、サイズは 100M バイトとなる。このモデルでの線量分布の計算結果は、図 13 のようになる。実際にはこのような等線量分布表示は、患部の CT 画像と重ね合わせて表示される。

予備実験の経験に基づき、今回は以下のように並列処理を行う。

- 計算領域（Ray 束）を渡されるとその領域の線量分布計算を行うプログラムを全プロセッサで起動する。ただし、サーバでは、このプログラムはデータ収集プログラムを兼ねている。つまり、クライアントから計算結果を受信していない時だけ、線量分布計算を行う。
- 計算領域を自マシン上の上記線量分布計算プログラムに渡す分割数分のプログラムを

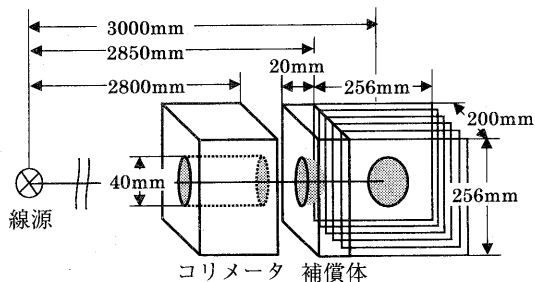


図 12 計算モデル

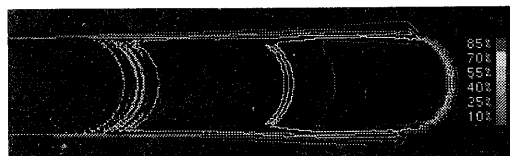


図 13 線量分布計算結果

ParaJET の下で起動する。なお、分割数はプロセッサ数の定数倍とする。

- 各マシン上の線量分布計算プログラムは、計算領域を受け取るとその領域の線量分布計算を行い、計算結果をサーバへ送信する。その際、計算した領域のみデータ転送し、線量値も 4 バイトから 2 バイトに圧縮して転送する。
- サーバ上のデータ収集プログラムは、分割数分の計算結果を受け取ると、全クライアントに終了を通知すると共に、結果をディスクに書き込んで終了する。

予備実験との違いは、並列処理方式が Ray 束単位であること他に、以下の通りである。

- (1) 計算結果を計算した領域のみサーバへ返す。全データ量は 100M バイトと非常に大きい。照射領域に相当するデータ量は、コリメータ直径 40mm とマージン 40mm から約 9.8M バイト ($80^2 \div 256^2 \times 100M$) となる。更に線量値を 2 バイトに圧縮して通信するので、通信データ量は 4.9M バイトとなる。
- (2) 計算領域をプロセッサ台数以上に細かく分割して、各プロセッサで複数領域を担当する。これにより、領域毎の計算時間のアンバランスを解消できる。
- (3) クライアントの線量分布計算プログラムは起動した後に計算領域を受け取る。複数領域を担当しても電子密度データの入力やコリメータや補償体等の照射機器情報の初期化が 1 回で済む。また、サーバの線量分布計算プログラムとデータ収集プログラムが 1 プロセスとなっているので、その間のデータのやり取りが不要となる。

4.2.2 実験結果

逐次及び 5 台並列³での実行時間を図 14 に示す。「並列」とは総数 25,600 本の Ray を 40 分割して各プロセッサで並列実行した実行時間であり、(a)~(c)は上記(1)~(3)の効果を確認するために計測したものである。

- 結果データの転送時に圧縮をしない。
- 領域をマシン台数と同じ 5 分割にする。
- サーバでデータ収集を別プロセスにし、クライアントでは計算領域を受け取るたび

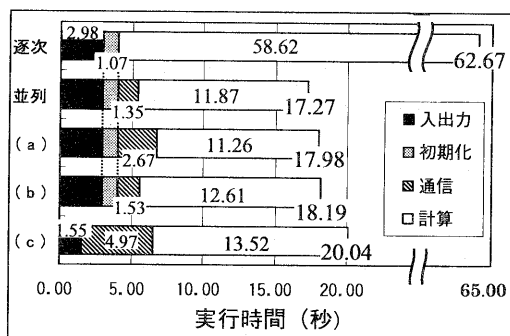


図 14 実行時間の内訳

に初期化を行う。領域は 10 分割。ここで「初期化」及び「入力」は、「計算」に含まれる。

これらの結果から、(1)~(3)の効果を確認することができた。本線量分布計算は、線源や照射機器情報を調整しながら何度か繰り返す利用形態が多い。つまり、初期化以降が繰り返されると考えて良く、その実行時間の並列台数効果は 5 台で 4.18 倍であり、かなり良い結果と言える。

5 おわりに

以上、放射線治療計画の並列化について、システム概要と評価実験結果について報告した。今後は、クラスタ内のギガビットイーサネットの評価や計算サーバとしての信頼性を確保するための故障対策等の機能を検討していく予定である。

参考文献

- [1] National Institute of Radiological Sciences, "Research for Cancer Treatment with Accelerated Heavy Ions," <http://www.nirs.go.jp/ENG/project1.htm>, 1993.
- [2] 辻井博彦, "重粒子放射線によるがん治療の現状," バイオメカニズム学会誌, Vol.21, No.3, pp.108-112, 1997.
- [3] 緑川博子編, "特集: 計算機クラスタ," 情報処理, Vol. 39, No. 11, pp.1072-1100, 1998.
- [4] L.Hong et al.,"A Pencil beam algorithm for proton dose calculations," Phys. Med. Biol. 41, pp.1305-1330, 1996.
- [5] <http://www.viarch.org/>
- [6] http://www.epm.ornl.gov/pvm/pvm_home.html.
- [7] <http://www.mpi-forum.org/index.html>.
- [8] 白石将, 佐藤裕幸, 中島克人, "分散型並列処理支援ツール ParaJET," 信学技報 CPSY96-60, pp.23-30, 1996.

³ 10 台のクラスタを二つに分けてデバッグに使用しているため、10 台での計測はまだできていない。