

スポットスキャン照射法による線量分布計算の並列化とその評価

佐藤裕幸^I, 蒲越虎^{II}, 山本雄一^{III}

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) 電力・社会システム事業所

がん治療法の一つである粒子線治療のスポットスキャン照射法による線量分布計算の PC クラスタによる並列計算の高速化を行った。線量分布計算においては、スポット群を分割して各 PC に割り付けることにより並列化を行った。その後のスポット毎の重み付けの最適化処理も、線量分布計算で割り付けられた各 PC で処理することにより並列化を行った。計測の結果、8,000 スポットの線量分布計算が 16 台 PC で約 31 秒、最適化処理が約 16 秒で行えることが分かった。これは従来、WS で数時間かけてバッチ処理で行っていたのに比較して、非常に高速であると言える。また並列化の効果に関しては、16 台で 1 台の時に比較して 15 倍以上高速化されていることを確認した。

Parallel Dose Calculation for Spot Scanning

Hiroyuki Sato^I, Yuefu Pu^{II}, Yuichi Yamamoto^{III}

I Mitsubishi Electric Corp. Information Technology R&D Center,
II Mitsubishi Electric Corp. Advanced Technology R&D Center,
III Mitsubishi Electric Corp. Energy & Industrial Systems Center

To provide a more highly efficient radiation treatment planning server at the heavy ion treatment facility for cancer therapy, we make the dose calculation for Spot Scanning more high speed using a network parallel processing system including 16 Pentium4 CPUs. Spots are divided and distributed into each processor. By experimentation, 8,000 spots dose calculation takes for 31 seconds and optimization takes for 16 seconds with 16 processors. More than 15 times speedup is performed with 16 processors.

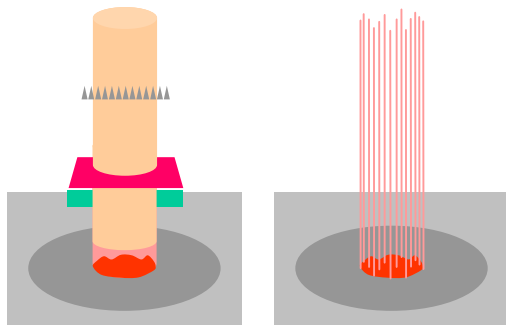
1 はじめに

がん治療法の一つである粒子線治療は、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]。この粒子線を用いたがん照射治療においては、患者毎に最適な照射方法を得る治療計画を立案する。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。そこで我々は、並列計算による線量分布計算の高速化を図った[2]。

粒子(陽子)線によるがん治療はこれまで、ブロードビーム法と呼ばれる照射方法が一般的であった。これは患者の形状にビームを整形するために、まず散乱体を用いてビームを広げ、そのあと患者形状に合わせたボーラス(ビームの進行方

向の深さを調整する機器)やコリメータ(ビームの広がり調整する絞り)と呼ばれる機器を使って整形し照射する方法である。しかしこの方法では、患部以外への照射があり、更に、機器を患者毎に製作する手間が必要であった。これに対して3次元スポットスキャン照射は、非常に細いパルス状ビームで患部を端から塗りつぶすように照射する方法であり(図1)、患部の輪郭に沿うように照射できるため、患部の外への不必要な照射がなく、患者ごとに機器を用意する必要もない。更に、患部中心に集中的に照射量を多くしたり、患部内部の照射線量分布を変化させることが可能である[3]。

このスポットスキャン照射法による線量分布計算は、より精密な計算が必要とされ、処理時間も圧倒的に要する。そこで、これまでの研究成果[2]を活かして、スポットスキャン照射法の線量分布計算の並列計算による高速化及びその評価を行ったので報告する。



ブロードビーム法 スポットスキャンニング法

図 1 ブロードビーム法と
スポットスキャンニング法

2 並列処理システム

近年のパソコン (PC) 用プロセッサや汎用ネットワークの性能向上はすさまじいものがあり、汎用並列計算機をはるかにしのぐコストパフォーマンスを有している。特にスポットスキャン照射法による線量分布計算では、非常に細いパルス状ビームである個々のスポット毎の線量分布計算は、それぞれ独立して行えるので、スポット群を各 PC (プロセッサ) に分割することにより並列化を行えば、汎用 PC を汎用ネットワークで接続した PC クラスタを用いても、充分性能を引き出せると考えられる。従って、安価に販売されている Windows PC をギガビットイーサネット で接続した PC クラスタを構築することにした。ただし、Windows 系 OS は、ネットワークを介したリモートアクセスの機能が弱いので、Linux を上書きインストールすることにした。

これまでの経験から、線量分布計算は演算性能そのものよりもメモリアクセス性能が全体の性能に影響することが分っている。特に計算過程において、電子密度データを計算点を中心として 3 次元にアクセスするため、キャッシュ・メモリがヒットし難い傾向にある。そこで、現在のメモリの中で最も高速な 400MHz デュアルチャネル DDR SDRAM を採用している PC を選択する事にした。各 PC (クライアント) のメモリ容量は、1 GB とし、サーバー PC については、現在の標準的な Linux がサポートしている 32 ビットアドレッシングで最大容量の 4GB とした。

サーバー PC、クライアント PC、及び全ての周辺機器は、幅 1,200mm、高さ 1,800mm、奥行き 450mm のラック内に収められている。クライアント PC は、スモールタワー型として、省スペース、廃熱性を高めた。サーバー PC はメモリ搭載量の要件からミニタワー型とした。各 PC は、現場でのコンソール操作のため、CPU 切り替え器によって、一台のディスプレイ、キーボードで全ての PC の



図 2 並列クラスタ外観

操作を行うように構成している (図 2)。

3 並列処理方式

並列化に関しては、前述のように、スポット群を各 PC に分割することにより行ったが、全体の処理の流れを図 3 に示す。ここで、サーバ、クライアントとはソフト的なプロセスを意味するものであり、サーバ PC 上ではサーバとクライアントの両方が動作している。従って、16 台の並列クラスタでは、1 つのサーバと 16 個のクライアントから構成されることになる。

- (1) サーバで本プログラムが起動されたら、使用する電子密度データやスポット情報等をコマンドライン又は設定ファイルから読み込む。そして、使用するクライアントの数によりスポット群を分割し、そのスポット情報と電子密度データ・ファイル名を各クライアントに送信する。
- (2) 各クライアントでは、送信された情報を基に各種初期設定、電子密度データの読み込みを行う。
- (3) 各クライアントでは、担当する各スポット毎に線量分布計算を行い計算結果を圧縮する。これを担当スポット全ての計算が終るまで繰り返す。
- (4) 各クライアントで担当スポット全てが終了したら、それらの線量値を合成して、各スポットの最大線量値とその位置と共にサーバへ送信する。
- (5) サーバは、各クライアントからの合成線量を合成する。
- (6) サーバは、各スポットの重みを算出するための基となる評価点座標を各スポットの最大線量位置から算出し、その情報を各クライアントへ送信する。
- (7) 各クライアントでは、受け取った各スポットの評価点の線量値を取り出し、サーバへ送信す

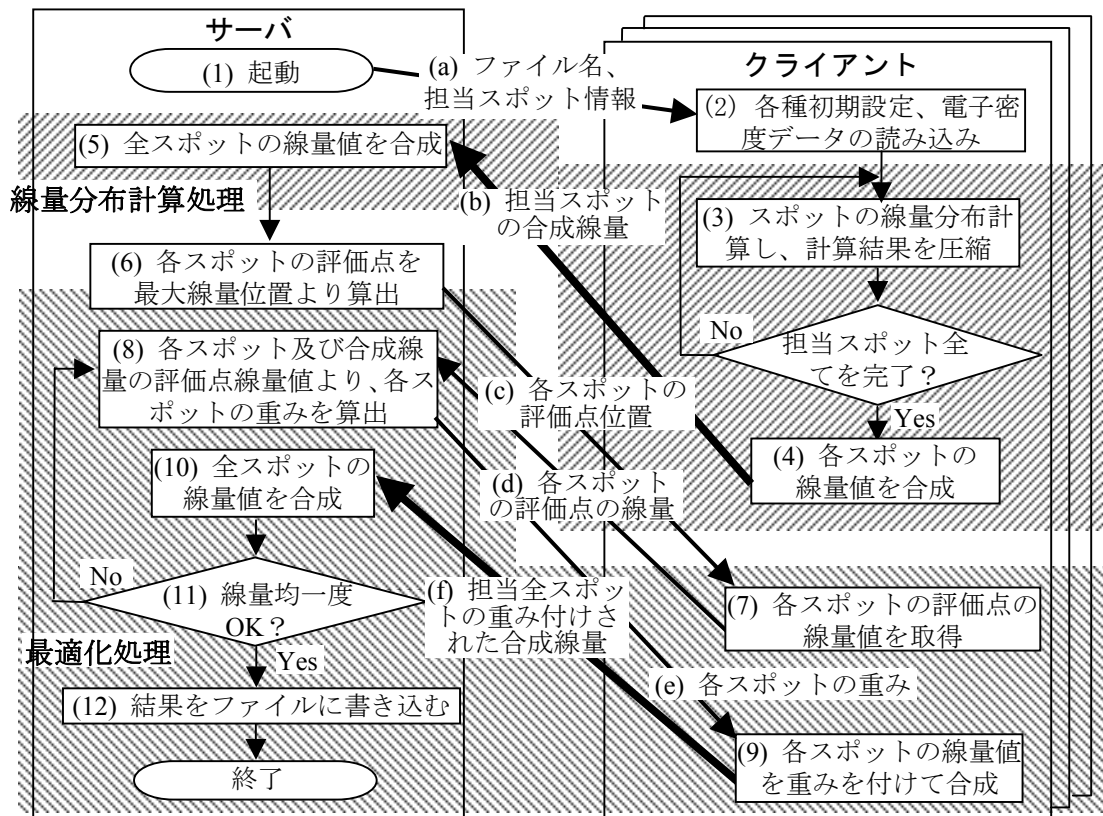


図 3 線量分布計算の並列処理の流れ

る。

(8) サーバは、評価点の各スポットの線量値と同位置の合成線量値から各スポットの重みを算出し、算出した重みを各クライアントへ送信する。

(9) 各クライアントは、送信された重み情報を基に、担当スポットを重みを付けて合成し、サーバへ送信する。

(10) サーバは、各クライアントからの重み付けされた合成線量を合成する。

(11) サーバは、照射領域内の合成線量の最大値と最小値から均一度を評価し、NG なら(8)へ戻り、OK なら次へ進む。

(12) サーバは、合成線量と各スポットの重みをファイルへ書き込み終了する。

この処理の流れにおいて、(3)~(5)が線量分布計算の処理で、(6)~(11)が各スポットの重みを算出する最適化処理になる。また、サーバとクライアントで通信が発生するのは図中の(a)~(f)である。

(a) ファイル名 (4B) と担当スポット情報 (8,000 スポットを 16 台で処理する場合は、 $2B \times 3 \text{ 次元} \times 500 \text{ スポット} = 3\text{KB} \times 15 \text{ 台} = 4.5\text{KB}$) である。

(b) 各クライアントの担当スポットの合成線量

である。通信の際に圧縮して送信するので、合成された線量が一塊になっていれば圧縮率が高く通信量も小さくなる。また通信の際は、各クライアントからサーバへ単純に送信するのではなく、各クライアント間で二分木状にマージしながら合成していくので、通信回数は PC 台数の \log となり、16 台で処理する場合はサーバから見れば 4 回の受信で済む。

(c) 各スポットの評価点の位置 (8,000 スポットを 16 台で処理する場合は、 $2B \times 3 \text{ 次元} \times 100 \text{ ケ所} \times 500 \text{ スポット} = 300\text{KB} \times 15 \text{ 台} = 4.5\text{MB}$) である。

(d) 各スポットの評価点の線量値 (8,000 スポットを 16 台で処理する場合は、 $4B \times 100 \text{ ケ所} \times 500 \text{ スポット} = 200\text{KB} \times 15 \text{ 台} = 3\text{MB}$) である。

(e) 各スポットの重み (8,000 スポットを 16 台で処理する場合は、 $4B \times 500 \text{ スポット} = 2\text{KB} \times 15 \text{ 台} = 30\text{KB}$) である。

(f) 各クライアントの担当スポットの重み付けされた合成線量である。通信量、通信方法等は、(b)と同様。

これらの処理の中で、線量分布計算の中心である(3)と最適化処理の中心である(8)~(11)が最も時間を要する部分である。(3)に関しては、各クライアントが担当するスポットの計算時間が均等になるようにするのが最も良い。また(8)~

(11)に関しては、それだけでなく(f)の通信時間も大きな割合を占めるため、その通信時間の短縮も考慮しなければならない。

各スポット毎の計算時間(線量分布計算時間及び線量の合成時間)は、その深さにより大きく異なっている。従って負荷分散方式として、各PCが担当するスポットの深さが均等になるようにスポット群を分割して、事前にスポット群のクライアントへの割り当てを決める静的負荷分散方式が考えられる。ただし、スポットが通過する位置の電子密度値によっても若干計算時間が異なっているので、それだけでは完全には負荷は均等になるとは限らず、各クライアント(PC)の負荷状況に基づいてスポット群を動的に割り付ける動的負荷分散方式がより均等化される。この動的負荷分散方式においては、可能な限り負荷分散オーバーヘッドを抑えるために、スポット群の9割を静的負荷分散と同様の方式で各クライアントに割り付け、残りの1割のスポット群を割り付けられたスポット群の計算が終了したクライアントがサーバへ次に計算すべきスポットを要求するという形(いわゆる要求駆動型動的負荷分散方式)で処理を進めていく。

一方、PC間の通信量(時間)を考えると、静的負荷分散によりスポット群を集中させて各クライアントに割り付けた方が圧縮効果が高くなり、通信量が低減する。従って、計算負荷の均等化という面では動的負荷分散方式、通信時間の低減という面では静的負荷分散方式が適していると言える。

4 計測評価

今回のスポットスキャン照射法による線量分布計算の並列化手法を評価するために種々の計測を行った。まず、1台と動的負荷分散方式による16台実行時の実行時間の内訳を計測し、それぞれの処理の内訳を解析する。次に、負荷分散方式を評価するために、動的負荷分散方式と静的負荷分散方式による実行時間の比較を行う。そして、各クライアントで計算した線量分布計算結果を合成する際に、二分木状にマージしながら線量を収集しているが、その効果を見るために、単純に収集した場合との比較を行う。

4.1 実行時間の内訳

表1に8,000スポットを1台のPCで実行した時と16台のPCで動的負荷分散方式により実行した時のサーバ上の実行時間の内訳を示す。また、表2に16台実行の線量分布計算における各PCの実行時間を示す。ここで「計算時間」とは実際に線量分布を計算している時間で、「待ち時間」とはサーバに新たな担当スポットを要求してから担当スポット情報が返されるまでの時間(複数回の総計)である。表3に重み最適化処理の実行時間内訳を示す。本来、最適化処理は、線量が均一になるまで繰り返されるが、今回は繰り返し数を100回に固定して計測した。また、線量分布計算の結果を図4に示す。図中で、黒枠で囲まれた部分がターゲット領域である(実際にはCT画像と重ね合わせて表示される)。

まず表1より、各処理の中では、線量分布計算と最適化処理が最も大きな割合を占めていることが分かる。16台での線量分布計算は表2から分かるようにPC0(サーバPC)は529本のスポットを担当しているので529回のループ処理であり、最適化では100回のループ処理となっていることから、多くの時間を要していることは予測できる。

次に表2より、スポットの深さ及び通過する電子密度値によって各スポットの計算時間が異なっているので、各PCが担当したスポット数は466~529とバラツキがある。動的負荷分散方式では各PCの負荷状況によってスポットを割り付けているので、各PCの全実行時間は100ミリ秒程度の差とほぼ均等になっている。待ち時間に関しても、最大で700ミリ秒程度であり、全実行時間の31秒に対して2%強とかなり小さく、動的負

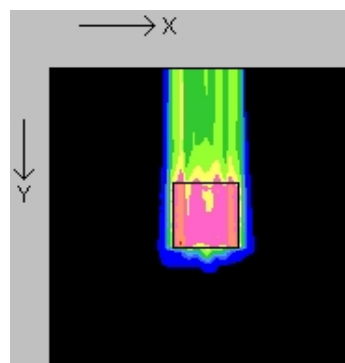


図4 8,000スポットの計算結果

表1 1台と16台での実行時間の内訳(秒)

| | 電子密度データ読込 | スポット生成 | クライアント生成 | 担当スポット情報送信 | 線量分布計算 | サーバ内線量合成 | 線量受信&合成 | 最適化 | 計算結果書出 | 合計 |
|-----|-----------|--------|----------|------------|---------|----------|---------|---------|--------|---------|
| 1台 | 0.231 | 0.245 | 0.000 | 0.009 | 481.677 | 0.899 | 0.007 | 100.367 | 0.295 | 583.730 |
| 16台 | 0.229 | 0.243 | 0.018 | 0.476 | 31.346 | 0.081 | 0.058 | 16.446 | 0.297 | 49.193 |

表 2 線量分布計算の各 PC の実行時間内訳 (秒)

| PC 番号 | 担当スポット数 | 全実行時間 | 計算時間 | 待ち時間 | PC 番号 | 担当スポット数 | 全実行時間 | 計算時間 | 待ち時間 |
|-------|---------|--------|--------|-------|-------|---------|--------|--------|-------|
| 0 | 529 | 31.346 | 31.227 | | 8 | 506 | 31.345 | 30.452 | 0.699 |
| 1 | 514 | 31.366 | 30.524 | 0.613 | 9 | 498 | 31.346 | 30.550 | 0.715 |
| 2 | 500 | 31.359 | 30.908 | 0.375 | 10 | 490 | 31.239 | 30.959 | 0.207 |
| 3 | 521 | 31.311 | 30.314 | 0.684 | 11 | 488 | 31.342 | 30.740 | 0.522 |
| 4 | 496 | 31.255 | 30.541 | 0.502 | 12 | 492 | 31.346 | 30.914 | 0.355 |
| 5 | 518 | 31.349 | 30.276 | 0.662 | 13 | 492 | 31.249 | 30.664 | 0.330 |
| 6 | 512 | 31.248 | 30.515 | 0.474 | 14 | 466 | 31.249 | 30.622 | 0.223 |
| 7 | 512 | 31.250 | 30.468 | 0.523 | 15 | 466 | 31.241 | 30.643 | 0.195 |

表 3 最適化処理の実行時間内訳 (秒)

| | 評価点線量値取得 | 重み計算 | 重み送信 | サーバ内線量合成 | 全体線量合成 | 線量受信 | 合計 |
|-----|----------|-------|-------|----------|--------|-------|---------|
| 1台 | 0.032 | 3.352 | 0.085 | 96.691 | 0.207 | 0.000 | 100.367 |
| 16台 | 0.038 | 3.427 | 0.641 | 7.200 | 1.295 | 3.844 | 16.446 |

表 4 各台数の実行時間 (秒) 及び台数効果 (速度向上率)

| 負荷分散方式 | PC台数 | 1 | | 2 | | 4 | | 8 | | 16 | |
|----------|--------|---------|------|---------|------|---------|------|--------|------|--------|-------|
| | 項目 | | | | | | | | | | |
| 動的負荷分散 | 線量分布計算 | 481.672 | 1.00 | 242.355 | 1.99 | 124.110 | 3.88 | 61.819 | 7.79 | 31.346 | 15.37 |
| | 最適化 | 100.367 | 1.00 | 58.421 | 1.72 | 31.907 | 3.15 | 20.555 | 4.88 | 16.446 | 6.10 |
| | 全体 | 583.729 | 1.00 | 302.522 | 1.93 | 157.558 | 3.70 | 83.828 | 6.96 | 49.237 | 11.86 |
| 静的負荷分散 | 線量分布計算 | 481.142 | 1.00 | 249.628 | 1.93 | 126.433 | 3.81 | 63.254 | 7.61 | 33.019 | 14.57 |
| | 最適化 | 99.982 | 1.00 | 53.864 | 1.86 | 30.232 | 3.31 | 20.100 | 4.97 | 14.117 | 7.08 |
| | 全体 | 582.825 | 1.00 | 305.234 | 1.91 | 158.181 | 3.68 | 84.796 | 6.87 | 48.568 | 12.00 |
| 単純静的負荷分散 | 線量分布計算 | 482.517 | 1.00 | 262.348 | 1.84 | 135.595 | 3.56 | 68.327 | 7.06 | 34.114 | 14.14 |
| | 最適化 | 99.551 | 1.00 | 61.885 | 1.61 | 42.002 | 2.37 | 25.671 | 3.88 | 22.772 | 4.37 |
| | 全体 | 583.762 | 1.00 | 326.049 | 1.79 | 179.198 | 3.26 | 95.544 | 6.11 | 58.331 | 10.01 |

荷分散のオーバーヘッドは小さいと言える。

表 3より、最適化処理においては、自PC内の各スポットの計算結果を合成する処理(サーバ内線量合成、全体線量合成、線量受信)が最も割合が大きい。最適化処理もスポット群を分割して並列処理していることになっているが、並列化の効果が出るのはサーバ内線量合成のみなので、最適化処理全体の並列化効果は線量分布計算に比べて大きくない。ただし今回の計測では、最適化処理の繰り返し数を100回にしているが、実際に最適化処理が収束するのがより短ければ、線量分布計算に比べて短時間で済むので、並列化効果をそれ程考慮する必要がないと思われる。

4.2 並列化効果

負荷分散方式を評価するために、動的負荷分散方式、静的負荷分散方式、単純静的負荷分散方式のそれぞれを用いた計測を行った。静的負荷分散方式では、各PCのスポットの深さの平均が等しくなるよう事前に割り付け、単純静的負荷分散方式では、そのような考慮をせずに単純に割り付けている。表 4に1台~16台の各PC台数毎の線量

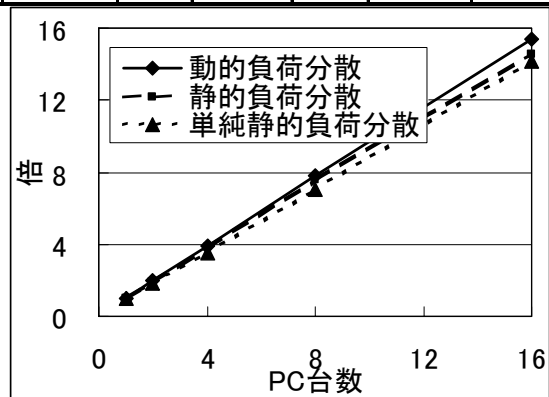


図 5 線量分布計算の並列化効果

分布計算、最適化処理及び全体の実行時間及び1台で実行した時に比べて各台数で何倍高速化されたかを示し、図 5に線量分布計算に関する速度向上率のグラフを示す。

まず、線量分布計算が31秒程度で処理できているのは、従来、WSで数時間かけてバッチ処理で行っていたのに比較して、非常に高速であると言える。これは、並列処理の効果だけでなく、Pentium 4 3.2GHzの(購入当時としては)最新のプロセッサを利用したことも大きい。また、16台で15.37倍の並列化効果に関しても、十分な高速化である。これは、スポット毎の計算の独立性

¹ 最適化処理は、現在も改良途中で、最終的には10回程度で収束することを目指している。

表 5 線量受信 & 合成時間

| 合成受信方式 | 項目 | PC台数 | 1 | 2 | 4 | 8 | 16 |
|---------|---------|------|---------|--------|--------|--------|--------|
| 二分木線量収集 | 線量受信&合成 | | 0.209 | 1.252 | 1.941 | 2.466 | 2.863 |
| | 最適化 | | 99.982 | 53.864 | 30.232 | 20.100 | 14.117 |
| 単純線量収集 | 線量受信&合成 | | 0.208 | 1.251 | 2.171 | 3.851 | 6.951 |
| | 最適化 | | 100.267 | 54.007 | 30.740 | 21.713 | 18.056 |

と計算量の大きさが、大きな要因であると思われる。一方、最適化処理に関しては、並列処理による効果は線量分布計算に比べてかなり低い。これは内訳でも示したように、通信時間の割合が高いためである。

それぞれの負荷分散方式による並列化効果を比較すると、線量分布計算ではやはり動的負荷分散方式が最も高く、最適化処理では静的負荷分散方式が最も高い。線量分布計算においては、計算量が十分に大きいため動的負荷分散のオーバーヘッドが相対的に小さくなっているからだと思う。一方、最適化処理においては、静的負荷分散方式の方が計算結果の圧縮率が高く通信量が小さくなっている効果が出ているのだと考えられる。このように 16 台までの計測では、どちらの負荷分散方式が良いかの判断は困難である。負荷バランス及び通信量とも全く考慮していない単純静的負荷分散方式は、予想通り並列化効果が低いという結果となっている。

また図 5 より、線量分布計算の並列化効果の伸びは直線になっているので、更に使用する PC の台数を増やすことにより、高速化できると予測できる。

4.3 線量合成

今回の並列処理方式では、各クライアント (PC) で計算した線量分布計算結果を合成する際に、二分木状にマージしながら線量を収集しているが、その効果を見るために、単純に収集した場合との比較を行う。単純な収集方式では、サーバ (PC0) から見れば、各 PC から 15 回の受信となり、二分木方式では、収集処理も並列化されるため、4 回の受信で済む。表 5 に最適化処理における各 PC からの合成線量の受信時間+サーバでの合成時間+線量の最大値と最小値の検索時間 (全て 100 回の合計時間) について、二分木状で収集する方式と単純に各 PC から逐次に収集する方式を示す。

2 つの収集方式の線量受信 & 合成時間を比較すると、4 台まではほとんど変わらず (理論的には 2 台までは変わらないはず)、8 台で二分木収集方式が若干速くなり、16 台で 2.4 倍程度速くなっている。16 台では二分木状で線量分布結果を収集することにより、15 回のデータ収集・合成が 4 回に減るが、時間的にはそこまでは効果が出ていないし、最適化処理全体の時間としても、

大きな効果とはなっていない。しかし、8 台から 16 台への高速化の度合いを見ると、更に PC 台数が多くなった場合には、大きな効果となると予測できる。

5 おわりに

以上、スポットスキュン照射法による線量分布計算の並列化方式及びその測定評価結果について報告した。

線量分布計算においては、スポット群を分割して各 PC に割り付けることにより並列化を行った。その後のスポット毎の重み付けの最適化処理も、線量分布計算で割り付けられた各 PC で処理することにより並列化を行った。計測の結果、8,000 スポットの線量分布計算が 16 台 PC で約 31 秒、最適化処理が約 16 秒で行えることが分かった。これは従来、WS で数時間かけてバッチ処理で行っていたのに比較して、非常に高速であると言える。負荷分散方式に関しては、各 PC の負荷状況により負荷を分散する動的負荷分散方式が 16 台で 1 台の時に比較して 15 倍以上高速化されている。また、予め各 PC の負荷が均等になるように割り付ける静的負荷分散方式でも 14 倍以上の高速化を確認した。最適化処理においては、PC 間の通信の割合が高いため、6~7 倍の高速化に留まっており、通信量の少ない静的負荷分散方式の方が並列化効果は高かった。ただしこれは、最適化処理の収束までの繰り返し数が 100 回の場合で、より少ない繰り返し数で収束すれば、最適化処理に関しては、並列化効果を考慮する必要はないかも知れない。

本研究開発は、独立行政法人 新エネルギー・産業技術総合開発機構の「産業技術実用化開発費助成金」により行われた。

参考文献

- [1] 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, Mar. 1997.
- [2] 佐藤裕幸, 中川隆文, 依田潔, 中島克人, 坂本豪信, 遠藤真広, "ワークステーション・クラスタを用いた放射線治療計画の高速化," 電子情報通信学会論文誌 D-I 第 J85-D-I 巻第 2 号, pp.184-192, 2002..
- [3] http://hadron.kek.jp/FFAG/150M_report/ffag_ir_you_pamph.pdf.