

分散メモリ型並列計算機上でのラジオシティ法の並列処理 ——並列フォームファクタ計算法と静的負荷分散の評価

上 嶋 明^{†,☆} 山 崎 勝 弘[†]

ラジオシティ法では光の相互反射を考慮した非常にリアルな画像を生成できるが、フォームファクタを求めるための計算量が莫大であり、その高速化が必要不可欠である。本論文では、フォームファクタ計算に用いる半球を細分化し、2次元のインタリーブ方式で各プロセッサに静的に処理を割り当てることでラジオシティ法を並列化する手法について述べる。各プロセッサで分散して描画された部分半球データを通信して、全プロセッサが完全な描画データを得た後、フォームファクタを同時に計算する処理方法を提案する。この方法では、通信量はエレメント数と無関係となり、シーンが小規模でプロセッサ数が少ない場合を除いて、部分フォームファクタデータの通信よりも性能が良くなる。また、プロセッサ間の負荷均衡度を平均値、最大値、および最小値の観点から評価する。プロセッサ64台を備えた分散メモリ型並列計算機AP1000+上で実験した結果、速度向上はベンチマークシーンで部分フォームファクタデータの通信の約2倍の20.4~21.8、現実的なシーンで35.0~40.2である。また、プロセッサ64台での負荷均衡度は0.93~0.96であり、負荷均衡がよく図られていることが分かった。

Parallel Radiosity on a Distributed Memory Parallel Computer: Evaluation of Parallel Form Factor Calculations and a Static Load Balancing Algorithm

AKIRA UEJIMA^{†,☆} and KATSUHIRO YAMAZAKI[†]

Although the radiosity algorithm can generate photo-realistic images due to the global illumination effects, a large amount of form factor calculations are required. This paper describes how to parallelize the radiosity algorithm by subdividing hemispheres into multiple elements and allocating them statically to multiple processors. An enhanced communication procedure, in which partial hemisphere data on each processor are communicated and complete hemisphere data are prepared on all processors, is proposed. In this procedure, sizes of communication data are independent of the number of elements. In addition, the load balance efficiency of our static load balancing algorithm is evaluated. On the distributed memory parallel computer AP1000+ with 64 processors, the speedup is 20.4 ~ 21.8 (almost twice compared to our previous method) for benchmark scenes, and 35.0 ~ 40.2 for classroom scenes. The load balance efficiency is 0.93 ~ 0.96.

1. はじめに

ラジオシティ法¹⁾では光の相互反射を考慮した非常にリアルな画像を生成できるが、フォームファクタ(ある部分から他の部分へ光が到達する割合)を求めるために莫大な計算量が必要となる。この計算量はシーンを構成するプリミティブ(基本形状)の数が多くなる

につれて増加し、逐次処理では非現実的な計算時間を要するため、並列処理による高速化の研究が進められている²⁾。

我々はマルチトランスピュータシステム上でのラジオシティ法の並列化で、ヘミキューブを細分化し、それを動的負荷分散と静的負荷分散により各プロセッサに割り当てる方法を用いた^{3),4)}。静的負荷分散はヘミキューブの解像度がプロセッサ数の倍数でない場合に、動的負荷分散と比べて並列化効率が最大で12%悪くなったが、他の場合では両者による性能の差はほとんどなかった。そこで、分散メモリ型並列計算機AP1000+システム上での並列化では、半球を細分化し、静的負荷分散により各プロセッサに割り当てる方法を用い

[†] 立命館大学理工学部

Faculty of Science and Engineering, Ritsumeikan University

[☆] 現在、神戸大学大学院自然科学研究科

Presently with Graduate School of Science and Technology, Kobe University

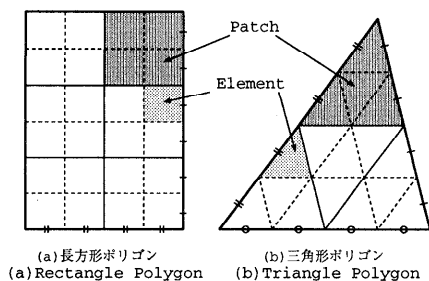


図 1 ポリゴンの分割

Fig. 1 Subdivision of polygons.

た⁵⁾。一般的に静的負荷分散では各プロセッサ間の処理時間の不均衡が問題となるので、負荷均衡を極力達成できる分散方法を用いることが重要である。本研究では、細分化した半球を2次元のインタリーブ方式で各プロセッサに割り当てるため、静的負荷分散法でも負荷均衡を図ることができ、かつ半球の解像度を比較的自由に設定することができる。

並列化効率は、プロセッサ1台での処理時間を $T(1)$ 、 n 台での処理時間を $T(n)$ とした場合に $T(1)/nT(n)$ で表される値として定義される。我々の従来の研究^{3)~5)}では、シーンが複雑になるにつれて並列化効率が低下するという問題があった。これは、並列フォームファクタ計算の際に、パッチ数またはエレメント数に比例した量の部分フォームファクタデータの通信が発生するためである。本研究では、並列フォームファクタ計算の際の通信方法を改良してこの問題を解決する。すなわち、各プロセッサに分散されている部分半球データを通信して、すべての半球データを集め、その後、全プロセッサで同時にフォームファクタを計算する。そして、最大のポリゴン数が約10万個の大規模なベンチマークシーンの処理を、部分フォームファクタデータ通信と部分半球データ通信の2通りの方法で行い、通信方法と負荷分散の観点から評価する。

本論文では、分散メモリ型並列計算機 AP1000+システム上で、ベンチマークシーンと現実的な教室のシーン各々3種類の処理を行い、通信の改良による並列フォームファクタ計算の性能改善と半球の細分化による静的負荷分散の評価を行い、考察を加える。

2. ラジオシティ法とその並列化

2.1 計算手順

シーン中の物体の表面を長方形と三角形のポリゴンで定義する。図1に示すように、各々のポリゴンを等間隔に分割して、光の放射の計算単位であるパッチ(patch)とし、さらに各々のパッチを等間隔に分割し

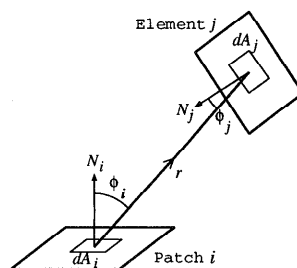


図 2 フォームファクタ

Fig. 2 Form factor.

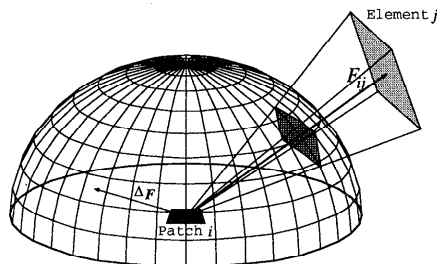


図 3 半球法

Fig. 3 Hemisphere method.

て、光の入射の計算単位であるエレメント(element)とする。ラジオシティの処理には計算効率の良い漸進法⁶⁾を用いる。漸進法では、ラジオシティの大きいパッチから順にエレメントへの光の放射を収束するまで繰り返す。任意の時点で計算途中の解が得られ、かつ所要メモリ量を節約できるのが特長である。放射を計算する際、フォームファクタと呼ばれる、パッチからエレメントへ光のエネルギーが到達する割合の計算が必要になる。

2.2 半球法によるフォームファクタ計算

パッチとエレメントが図2に示すような位置関係にあるとき、フォームファクタ F_{ij} は、次のように表される¹⁾。

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos \phi_i \cos \phi_j}{\pi r^2} dA_j dA_i \quad (1)$$

パッチとエレメントの間に障害物が存在するような複雑な位置関係のときに、この式によりフォームファクタを正確に計算するのは困難であるため、本研究では半球法⁷⁾を用いてフォームファクタを計算する。

半球法では、図3に示すように、放射パッチを中心にした半球を考え、その表面を細かいメッシュに分割し、各々のメッシュへのデルタ・フォームファクタ(ΔF)をあらかじめ計算しておく。全エレメントを半球に投影して隠面消去し、エレメントが対応づけられたメッシュの ΔF の合計を求めれば、パッチから全エレメントへのフォームファクタが計算できる。ただし、

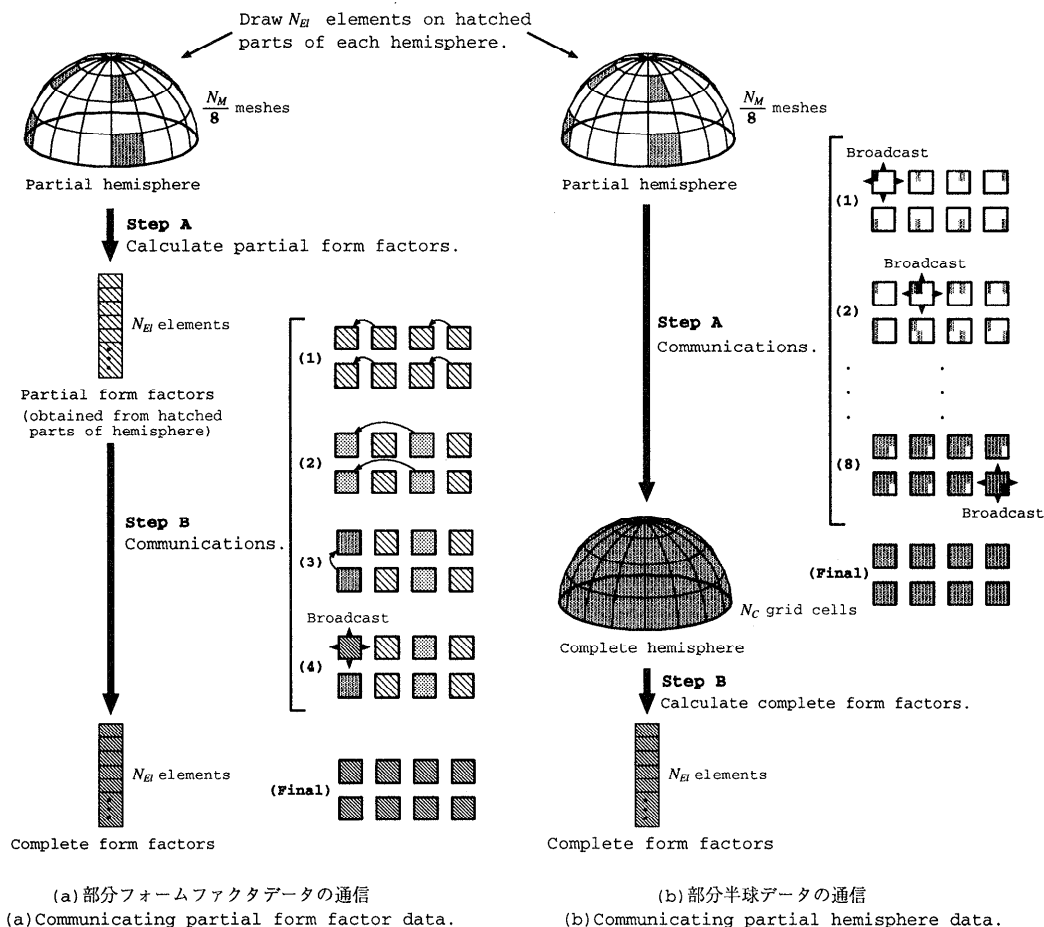


図4 2種類の通信方法による並列フォームファクタ計算
Fig. 4 Parallel form factor calculations using two communication procedures.

本研究での半球法の処理においては、エレメントを直接投影するのではなく、ポリゴンを半球に投影して隠面消去を行った後、各メッシュが対応するエレメントを決定している⁵⁾。半球には、ヘミキューブに存在する面の継ぎ目がなく、また、放射パッチの法線方向に近く放射ラジオシティが大きい部分のメッシュが細くなるという特長がある。

2.3 データ配置

シーンデータの配置では、ポリゴンデータのコピーを全プロセッサに持たせ、パッチデータを各プロセッサにインタリーブ方式に分散させている。エレメントデータは、それを含むパッチデータが置かれるプロセッサが保持する。この方法では、全プロセッサが半球への描画に必要なポリゴンデータを保持するため、並列フォームファクタ計算の際に座標のデータをプロセッサ間で転送する必要はない。さらに、パッチデータとエレメントデータを各プロセッサに分散して配置

し、最大ラジオシティ値を持つパッチの検索と各エレメントのラジオシティ値の更新を、保持するプロセッサに分散して行わせることでメモリの節約を行う⁵⁾。

2.4 並列フォームファクタ計算法

並列フォームファクタ計算の際、図4に示すような、部分フォームファクタデータの通信と部分半球データの通信の2通りの方法を用いる。ここでは、8台のプロセッサによる並列処理を示している。いずれの場合も、まず各プロセッサがシーンに存在する N_{EI} 個のエレメントを担当部分に描画する。次に、前者では各プロセッサが部分フォームファクタデータを並列に計算した後、それらをプロセッサ間で通信して全プロセッサが完全なフォームファクタを得る。後者では各プロセッサが担当した描画データをプロセッサ間で通信して全プロセッサが完全な描画データを得た後、各プロセッサが同時にフォームファクタを計算する。

2.4.1 部分フォームファクタデータの通信による方法

図4(a)に示すように、各プロセッサがそれぞれの描画担当部分から得られる部分フォームファクタ (partial form factors) を計算する (ステップ A)。そして、全プロセッサが部分フォームファクタデータの通信により完全なフォームファクタを得る (ステップ B)。ステップ B の通信では、各プロセッサが保持する部分フォームファクタデータを、プロセッサ 2 台分ずつ合計しながら 1 カ所に回収することで完全なフォームファクタを計算し、それを全プロセッサにブロードキャストする。プロセッサ数を N_{Pr} とすると、回収のための $\lceil \log_2 N_{Pr} \rceil$ ステージの通信と 1 ステージのブロードキャスト通信が必要である。たとえば、プロセッサ 8 台の場合、図 4(a) のような 4 ステージの通信となる。エレメント 1 個分の部分フォームファクタデータは 4 バイトであるため、シーン中に N_{El} 個のエレメントがある場合、各通信ステージでの通信量は $4N_{El}$ バイトとなる。

この方法では、ステップ A のデルタ・フォームファクタの加算による部分フォームファクタの計算を並列化できるが、ステップ B でエレメント数に比例した通信量が必要となることが欠点である^{5),8)}。この欠点は、エレメント数の多い大規模シーンで並列化効率が大きく低下する原因となっていた。

2.4.2 部分半球データの通信による方法

図 4(b) に示すように、各プロセッサ上の部分半球 (partial hemispheres) データの通信により全プロセッサが完全な描画データを得る (ステップ A)。そして、各プロセッサがデルタ・フォームファクタの加算により完全なフォームファクタの計算を行う (ステップ B)。ステップ A の通信では、各プロセッサが保持する部分半球データを、順にブロードキャストしている。プロセッサ数と等しいステージのブロードキャスト通信が必要であり、たとえば、プロセッサ 8 台の場合、図 4(b) のような 8 ステージのブロードキャスト通信となる。半球の各メッシュに対してエレメント ID を格納する領域が 4 バイトあるため、半球のメッシュ数を N_M 、プロセッサ数を N_{Pr} とすると、各通信ステージでの通信量は $(4N_M/N_{Pr})$ バイトとなる。

この方法には、エレメント数ではなく半球の 2 方向の解像度の積 ($R_\phi \times R_\theta$) に依存した通信量で処理を行えるという特長がある。

3. 半球の細分化による負荷分散

3.1 動的負荷分散と静的負荷分散

フォームファクタ計算の際、シーンを構成するポリゴンの配置に依存して、ヘミキューブや半球の各メッシュごとの描画時間には大きな差が生じる可能性がある。そこで、各プロセッサ間の処理時間を可能な限り均等にする工夫が必要であり、細分化したヘミキューブや半球を各プロセッサに割り当てる方法として、動的負荷分散と静的負荷分散が考えられる。前者では、各プロセッサの負荷に応じて、各プロセッサへの割当てを処理中に動的に決定する。そのため、各メッシュごとの処理時間の差が大きい場合でも負荷均衡を図れるが、動的な割当てを行うためのオーバーヘッドが発生する。一方、静的負荷分散では、各プロセッサへ固定的に割り当てるため、負荷均衡の点では動的負荷分散より劣るが、割当てのためのオーバーヘッドは発生しない。

我々はマルチトランスピュータシステム上での実験で、動的負荷分散と静的負荷分散の比較を行った^{3),4)}。静的負荷分散を用いた場合、ヘミキューブの解像度がプロセッサ数の倍数でない場合に並列化効率が最大で約 12% 悪くなるという問題があるものの、他の場合では負荷分散方法による大きな性能の差はなかった。すなわち、静的負荷分散であっても、各プロセッサに等しい数の処理をインタリーブ方式で割り当てれば、大きな負荷不均衡が生じないと考えられ、本研究では静的負荷分散を用いる。

3.2 半球の細分化による静的負荷分散

ヘミキューブ法と半球法は、各メッシュにエレメントを描画してフォームファクタ計算を行う点で処理が類似しているため、半球法に対してもインタリーブ方式による負荷分散を適用できる。半球表面を細分化したメッシュを、2 次元のインタリーブ方式で各プロセッサに静的に処理を割り当てる。プロセッサが $(0,0) \sim (3,1)$ までの 8 台の場合、半球上の座標 $(\phi, \theta) = (p, t)$ のメッシュは、プロセッサ $(t \bmod 4, p \bmod 2)$ に割り当てられる。図 5 に解像度 $(R_\phi, R_\theta) = (8, 32)$ の半球を示す。たとえば、プロセッサ $(0,1)$ は図の網掛部分の描画を担当する。半球へのポリゴンの描画の際には、まず、全プロセッサが担当する部分の描画を並列に行う。その後、2.4 節で述べた 2 通りの方法で全プロセッサが完全なフォームファクタを得る。

表 1 ベンチマークシーン

Table 1 Benchmark scenes.

シーン	立方体数	ポリゴン数	パッチ数	エレメント数	放射回数	データ量 (MB)
B1	4096	24588	49776	199032	100	5.4
B2	8000	48012	96624	386424	100	10.5
B3	17576	105468	211536	846072	100	23.0

表 2 教室のシーン

Table 2 Classroom scenes.

シーン	ポリゴン数	パッチ数	エレメント数	放射回数	データ量 (kB)
C1	3159	6344	17402	758	642.3
C2	4629	9236	24342	1202	931.2
C3	7592	14941	37127	1751	1502.2

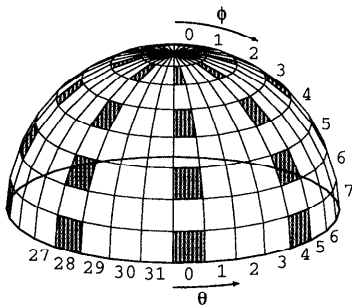


図 5 半球の細分化

Fig. 5 Subdivision of hemispheres.

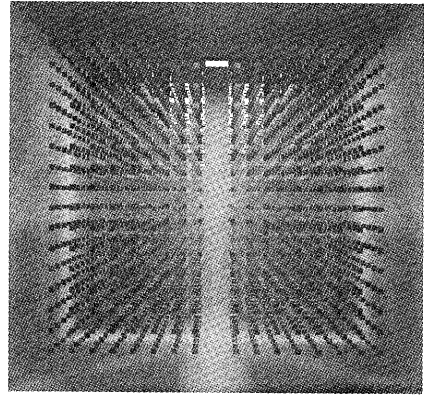


図 6 ベンチマークシーン B1

Fig. 6 Benchmark scene B1.

4. 実験と考察

4.1 分散メモリ型並列計算機 AP1000+

AP1000+は富士通研究所で開発された分散メモリ型の並列計算機である⁹⁾。本研究ではセルが64台接続されたシステムを使用する。各セルは動作クロック周波数 50 MHz の SuperSPARC プロセッサ、64 M バイトのローカルメモリ、メッセージコントローラ、およびネットワークデバイスなどで構成される。各セルは、セル間 1 対 1 通信用 T-net、ホスト-セル間通信とセル間のブロードキャスト通信用 B-net、およびバリア同期とステータス通信用 S-net の 3 種類のネットワークにより結合される。任意のセル間の通信やブロードキャスト通信などを指定でき、セル間の通信に要する時間は距離にはほぼ無関係で、通信量に依存するという特徴がある。メッセージ通信は基本的に非同期型で、送信側では、送信の完了を待つことなく後続の処理を実行できる。ファイル入出力などは、セル群と接続されたホスト計算機（ワークステーション）上で処理される。

4.2 実験方法

4.2.1 ベンチマークシーンの処理

表 1 に示す 3 通りのベンチマークシーンを用いて、2 通りの並列フォームファクタ計算法の評価を行う。両計算方法とも、計算量は半球の解像度に依存して増加する。また、部分半球データの通信を用いた場合は、通信量も半球の解像度に依存するため、3 通りの半球の解像度による評価も行う。処理はいずれも漸進法の放射を 100 回行う。全シーンに共通して、天井、床、4 枚の壁で囲まれた空間内に光源と複数の立方体がある。立方体は 6 個のポリゴンで構成され、各ポリゴンは 2 個のパッチに、さらに各パッチは 2×2 個のエレメントに分割されている。天井、床、4 枚の壁のポリゴンは各々 5×5 個のパッチに、各パッチは 2×2 個のエレメントに、そして光源の 6 個のポリゴンは、各々 2×2 個のパッチに分割されている。シーン B1 のラジオシティの計算結果をレンダリングしたものを図 6 に示す。

4.2.2 現実的な教室のシーンの処理

表 2 に示す 3 種類の現実的な教室のシーンを用い

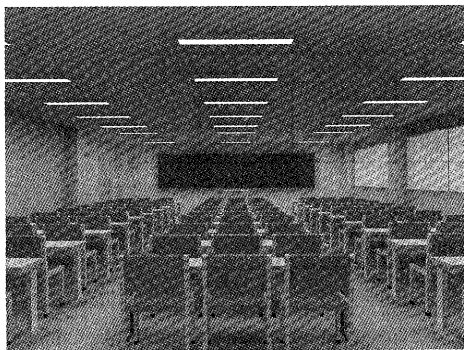


図 7 教室のシーン C3

Fig. 7 Classroom scene C3.

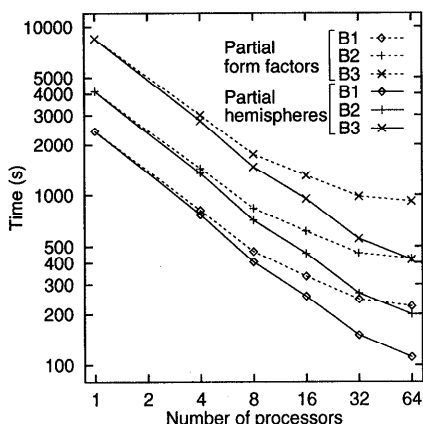


図 8 通信方法による比較 (ベンチマークシーン)

Fig. 8 Comparison of the two communication procedures (benchmark scenes).

て、2通りの並列フォームファクタ計算法、および負荷分散の評価を行う。処理は漸進法の計算が収束するまでとする。これらの各シーンは部屋の大きさと光源、机、椅子などの数が異なり、C1, C2, C3の順に複雑になる。C1, C2, C3は各々36人, 54人, 90人が着席できる教室のシーンである。シーンC3のラジオシティの計算結果をレンダリングしたものを図7に示す。

4.3 結果と考察

4.3.1 並列フォームファクタ計算法の評価

(1) ベンチマークシーンの処理

結果を図8に示す。半球の解像度はすべて(128, 512)として、ベンチマークシーンB1, B2, B3の処理を行っている。Partial form factorsが部分フォームファクタデータの通信, Partial hemispheresが部分半球データの通信を示している。部分フォームファクタデータの通信を用いてプロセッサ数を増加させたとき、処理時間の減少の割合が悪くなっている。これは、通信

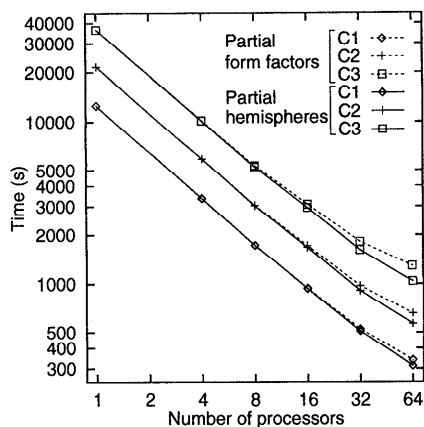


図 9 通信方法による比較 (教室のシーン)

Fig. 9 Comparison of the two communication procedures (classroom scenes).

する部分フォームファクタデータの量がエレメント数に比例して増加し、特にプロセッサ数が多い場合には、この通信によるオーバーヘッドの割合が大きくなるためである。すべてのシーンにおいて、明らかに部分半球データの通信の方が優れている。そして、通信方法による差はプロセッサ数が多くなるほど大きく現れている。プロセッサ64台の場合の速度向上は、部分フォームファクタデータの通信を用いたとき9.5~11.1, 部分半球データの通信を用いたとき20.4~21.8となり、部分半球データの通信を用いることで処理速度が約2倍に向上している。本実験で用いたベンチマークシーンでは均等に立方体を配置しているが、不均等な配置のとき、処理時間の増加はプロセッサ64台の場合で約2%程度となることを確認した。

(2) 現実的な教室のシーンの処理

結果を図9に示す。半球表面の解像度はすべて(128, 512)としている。C1からC3へとシーンが複雑になるにつれ、両通信方法による処理時間の差が大きくなり、部分半球データの通信がより有効となる。ただし、C1, C2, C3ともベンチマークシーンに比べて小規模であるため、通信方法による差は小さくなっている。シーンC1でのプロセッサ数が8台以下の場合には部分フォームファクタデータの通信を用いた方がわずかながら性能が良いが、他の場合は部分半球データの通信を用いた方が性能が良くなっている。つまり、シーンが小規模でプロセッサ数が少ない場合を除いて、部分半球データの通信の方が有効である。プロセッサ64台の場合の速度向上は、部分フォームファクタデータの通信を用いたとき28.2~37.0, 部分半球データの通信を用いたとき35.1~40.2となる。ベンチマー

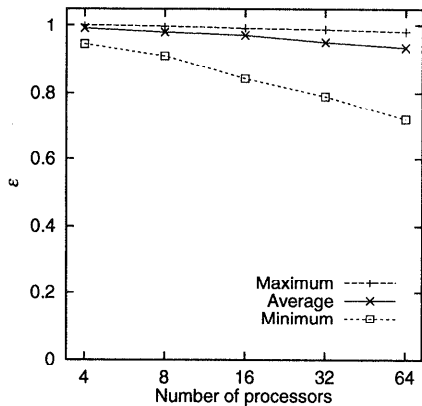


図 10 プロセッサ数による負荷均衡の評価 (シーン C2)

Fig. 10 Evaluation of load balancing based on the number of processors (scene C2).

クシーンに比べて小規模であるため、両通信方法による速度向上の差も小さくなっている。本実験で用いた教室のシーンでは部屋の内部に机と椅子を均等に配置しているが、部屋の一部分に集中して配置したとき、処理時間の増加はプロセッサ 64 台の場合で約 3% 程度となることを確認した。

4.3.2 負荷分散の評価

全プロセッサの中の最短の処理時間を T_{min} 、最長の処理時間を T_{max} としたときの負荷均衡度を $\varepsilon = T_{min}/T_{max}$ とする。負荷均衡が完全にとれているとき $\varepsilon = 1$ となり、不均衡になるにつれて値が小さくなる。漸進法の 1 回の放射ごとの ε の平均値、最大値、最小値を測定した。シーン C2 の処理におけるプロセッサ数による負荷均衡の評価を図 10 に示す。半球の解像度はすべて (128, 512) としている。プロセッサ数が増えるほど ε は低下し、負荷均衡を図ることが難しくなる。しかし、処理時間に大きく影響する平均値は 4 台で 0.99、64 台でも 0.94 であり、負荷均衡がうまく図られていることが分かる。

シーンと解像度による負荷均衡の評価を図 11 に示す。プロセッサ数は 64 台である。各シーンにおいて解像度と負荷均衡の関係では、解像度が高くなると ε が大きくなり、負荷均衡がよく図れることが分かる。これは、解像度が高いほど各メッシュの描画時間の差が小さくなること、および割り当てる処理数が増加することが理由と考えられる。シーンと負荷均衡の関係では、平均値と最大値は解像度が同じ場合にシーンにかかわらずほぼ同等で、それぞれ 0.93 ~ 0.96、0.97 ~ 0.99 となっている。ただし、C3 の最小値がほかにくらべて低く、0.55 ~ 0.71 になっている。これは、シーン C3

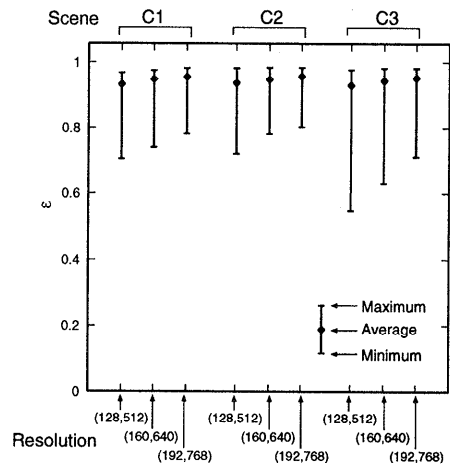


図 11 シーンと解像度による負荷均衡の評価 (プロセッサ 64 台)

Fig. 11 Evaluation of load balancing based on scenes and resolutions (64 processors).

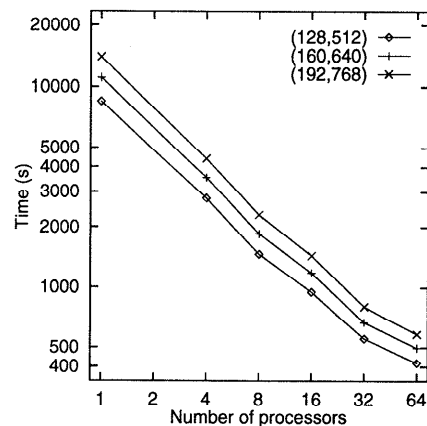


図 12 半球の解像度による比較

Fig. 12 Comparison based on the resolution of the hemispheres.

のポリゴン数が多いため、半球の各メッシュの描画時間の差が比較的大きくなること、および漸進法の放射回数が多いため、不均衡となる放射が現れる確率が高くなることが原因と考えられる。

4.3.3 半球の解像度による比較

結果を図 12 に示す。3 通りの解像度 (128, 512)、(160, 640)、(192, 768) を用いて、部分半球データの通信によりベンチマークシーン B3 の処理を行っている。一般的に、半球法によるフォームファクタ計算では、エレメントの大きさが相対的に小さくなるほど解像度を高くする必要がある。プロセッサ 1 台のとき、解像度 (192, 768) の処理時間は解像度 (128, 512) の 1.65 倍となっているが、プロセッサ 64 台の場合は 1.40

倍となっている。プロセッサ 64 台の場合の速度向上では、解像度 (128, 512) のとき 20.27 倍, (192, 768) のとき 23.91 倍となっている。すなわち、半球の解像度が高いほど並列化効率が良くなっている。これは、解像度を高くすることによる通信量の増加の割合よりも描画時間の増加の割合の方が大きいためであると考えられる。

5. 関連する研究

ヘミキューブの細分化による並列化を行った Baumら¹⁰⁾の研究では、共有メモリ型マルチプロセッサ・グラフィックワークステーション SGI 4D/280 GTX を用い、1つのプロセッサをプロデューサ、残りのプロセッサをコンシューマとしている。プロデューサはヘミキューブへのポリゴンの描画をハードウェアで処理した後、ヘミキューブを分割して各コンシューマに割り当てる。各コンシューマは並列にデルタ・フォームファクタの加算、放射、次の放射パッチの選択を行う。シーンデータはプロデューサのみが使用するため、1カ所に集中して配置される。性能を向上させるため、プロデューサとコンシューマの処理はオーバラップして行われる。

大規模シーンの並列処理を行った Sturzlingerら¹¹⁾の研究では、プロセッサ 256 台の分散メモリ型並列計算機 nCube2S 上で、パッチ数 6 万個、エレメント数 13.1 万個のシーンの処理を行っている。フォームファクタ計算にはレイキャスティングを用い、パッチを各プロセッサに分散させて交差判定を並列化している。プロセッサのグループ間で動的負荷分散法を用いて負荷均衡を図っている。速度向上は、机上計算でプロセッサ 64 台のとき約 20 倍、256 台のとき約 50 倍となっている。

また、Renambotら¹²⁾の研究では、プロセッサ 32 台の分散共有メモリ型並列計算機 SGI Origin 2000 上で、ポリゴン数 108 万個、パッチ数 128.8 万個のシーンの処理を行っている。環境を細かく分割して各プロセッサに割り当て、各プロセッサは、放射パッチからのレイと環境内の物体との交差判定の結果であるマスクを隣接環境担当のプロセッサに順次送信する。速度向上は 21 倍となっている。

本研究では半球への描画も並列化している点、および部分半球データの通信によるフォームファクタ計算法を新たに考案し、並列化効率を大幅に改善している点に特長がある。その上で、パッチ数 21.1 万個、エレメント数 84.6 万個の大規模なシーンの処理を行った。

6. おわりに

半球を細分化して 2 次元のインタリーブ方式で各プロセッサに割り当てるラジオシティ法の並列化方法において、部分半球データの通信による並列フォームファクタ計算法を新たに考案し、部分フォームファクタデータの通信による方法よりも優れていることを示した。また、負荷均衡度を平均値、最大値、および最小値の観点から評価した。

並列フォームファクタ計算法の評価では、ベンチマークシーンを処理したとき、部分半球データの通信の方がすべての場合において優れており、プロセッサ 64 台での速度向上は、従来の方法の約 2 倍の 20.4 ~ 21.8 である。現実的な教室のシーンの処理でも、シーンが小規模でプロセッサ数が少ない場合を除いて、部分半球データの通信の方が優れており、速度向上は 35.0 ~ 40.2 である。

負荷均衡の評価では、プロセッサ数の増加とともに負荷均衡度が下がるが、平均値は 64 台の場合でも 0.93 ~ 0.96 となり、負荷均衡がよく図られていること、および半球の解像度が高いほど負荷均衡度が高くなることが分かった。

今後の課題として、より多くのシーンを用いた実験、処理時間のモデル化、およびマルチパスレンダリング法の並列化¹³⁾などがあげられる。

謝辞 本研究全般を通じて、有益な議論とコメントをいただいた立命館大学理工学部得丸英勝教授、渡部透教授に深謝します。また、並列計算機を利用させていただいた(株)富士通研究所 HPC 研究センターならびに京都大学工学部情報工学教室富田研究室に感謝します。さらに、有益なコメントをいただいた査読者に感謝します。

参 考 文 献

- 1) Goral, C., Torrance, K., Greenberg, D. and Battaile, B.: Modeling the Interaction of Light Between Diffuse Surfaces, *Computer Graphics*, Vol.18, No.3, pp.213-222 (1984).
- 2) 安部 毅, 大野義夫, 西田友是, 近藤邦雄, 中嶋正之: 並列グラフィクスアルゴリズムのサーベイ, 情報処理グラフィクスと CAD 研究報告, 68-2, pp.9-16 (1994).
- 3) 上嶋 明, 山崎勝弘, 渡部 透, 得丸英勝: マルチトランスピュータシステム上でのラジオシティ法の並列化, 情報処理学会論文誌, Vol.37, No.7, pp.1477-1484 (1996).
- 4) Yamazaki, K., Uejima, A., Watanabe, T. and Tokumaru, H.: Parallel Radiosity Image Gen-

- eration on a Distributed Memory Machine, *Proc. 7th Transputer/occam International Conference*, pp.211-225 (1996).
- 5) 上嶋 明, 山崎勝弘: 分散メモリ型並列計算機 AP1000+上でのラジオシティ法の並列化, 電子情報通信学会論文誌 (D-II), Vol.J80-D-II, No.7, pp.1852-1859 (1997).
 - 6) Cohen, M., Chen, S., Wallace, J. and Greenberg, D.: A Progressive Refinement Approach to Fast Radiosity Image Generation, *Computer Graphics*, Vol.22, No.4, pp.75-84 (1988).
 - 7) Spencer, S.: The Hemisphere Radiosity Method: A Tale of Two Algorithms, *Eurographics Workshop on Photosimulation, Realism and Physics in Computer Graphics*, pp.127-135 (1990).
 - 8) 上嶋 明, 山崎勝弘: AP1000+上でのラジオシティ法の並列化—静的負荷分散の評価と通信の改良による性能改善, 並列処理シンポジウム (JSPP'98) 論文集, Vol.98, No.7, pp.167-174 (1998).
 - 9) 石畑宏明, 堀江健志, 清水俊幸, 林 憲一, 小柳洋一, 今村信貴, 白木長武: AP1000+: デザインコンセプト, 情報処理学会研究会報告, 94-ARC-107, Vol.94, No.66, pp.153-160 (1994).
 - 10) Baum, D. and Winget, J.: Real Time Radiosity Through Parallel Processing and Hardware Acceleration, *Proc. 1990 Symposium on Interactive 3D Graphics*, pp.67-75 (1990).
 - 11) Sturzlinger, W., Schaufler, G. and Volkert, J.: Load Balancing for a Parallel Radiosity Algorithm, *IEEE/ACM 1995 Parallel Rendering Symposium (PRS'95)*, pp.39-45 (1995).
 - 12) Renambot, L., Arnaldi, B., Priol, T. and Pueyo, X.: Towards Efficient Parallel Radiosity for DSM-Based Parallel Computers Using Virtual Interfaces, *Proc. IEEE Symposium on Parallel Rendering (PRS'97)*, pp.79-86 (1997).
 - 13) Yamauchi, H., Maeda, T., Kobayashi, H. and Nakamura, T.: Performance Evaluation of a Parallel Multi-Pass Rendering Algorithm Based on the Object-Space Parallel Processing Model, 並列処理シンポジウム (JSPP'98) 論文集, Vol.98, No.7, pp.175-182 (1998).

(平成 10 年 8 月 31 日受付)

(平成 11 年 3 月 5 日採録)



上嶋 明 (正会員)

昭和 43 年生. 平成 3 年立命館大学理工学部情報工学科卒業. 平成 5 年同大学院博士前期課程修了. 平成 8 年同大学院博士後期課程単位取得退学. 平成 10 年 4 月立命館大学理工学部助手. 平成 10 年 10 月神戸大学大学院自然科学研究科助手. 工学博士. コンピュータ・グラフィックス, 並列処理の研究に従事. 電子情報通信学会, ACM, IEEE 各会員.



山崎 勝弘 (正会員)

昭和 28 年生. 昭和 51 年京都大学工学部情報工学科卒業. 昭和 53 年同大学院修士課程修了. 昭和 55 年同大学院博士課程退学. 同年より宇都宮大学工学部助手. 昭和 63 年立命館大学理工学部助教授. 平成 6 年同教授. 工学博士. 平成 4 年から 1 年間マンチェスター大客員研究員. 平成 6 年山内業績賞. 並列処理, 並列プログラミング, 事例ベース推論の研究に従事. 電子情報通信学会, ACM, IEEE 各会員.