

科学技術計算の可視化処理における並列画像生成方式とその評価†

大宅 伊久雄†† 吉田 隆 義†† 和 宇 慶 康††

画像生成技術は、様々な応用分野でマンマシンインタフェースの向上に役立っている。科学技術計算分野では計算結果の可視化が単なる後処理としての確認作業に止まらず、事象の観察や解析に特に有効であることが認識されてきた。この可視化を実現するには、動的な3次元空間の事象の変化をモデル化し、アニメーション画像として生成することが要求される。しかしこの画像生成には、形状モデル生成のための幾何処理と座標変換、隠面処理や滑らかな濃淡付け処理など多様な演算を含む膨大な繰返し計算を必要とする。従来様々な処理方式の適用が試みられたが、実用的に満足する時間内で処理することは困難であった。本研究では画像生成処理に内在する並列性に着目して可視化処理に有効な並列モデルを初めに構築し、それに基づく並列処理装置プロトタイプを試作した。装置本体は幾何計算処理部とピクセル処理部からなる多層並列処理方式を採用した。そして装置本体上に制御ソフトウェアを構築し、プラズマ流体シミュレーション、衛星画像の3次元表示および分子振動シミュレーションの可視化ソフトウェアを本並列方式により記述し動作させた。その結果から本並列処理方式の台数増加による定量的評価を行い、可視化処理における有効性と今後の課題が確認できたのでここに報告する。

1. はじめに

画像生成は^{1),2)}、1950年代の後半より一貫してコンピュータとの有効な対話技術として進展してきた。当初はCADのような専門分野で高価な対話のツールとして利用されたが、1980年代初期のBitmap Displayの発明によりウィンドウシステムに應用され一般に普及した³⁾。近年自然な3次元画像を生成するアルゴリズムも多く提案され、コンピュータの性能が加速されるにともない新たな応用が現実のものとなってきた。

可視化(Visualization)^{4),5)}は流体力学、分子モデリング、プラズマ物理、有限要素解析などの科学技術計算分野において、シミュレーションの結果を数値に代わって視覚的に表示することを意味する。可視化は不可視なものを見るための手段を提供し、研究者がシミュレーションや計算結果を観察することを可能とする。この可視化処理では3次元空間における事象の変化を幾何学的にモデル化し、研究者が解釈しやすい速度でアニメーション表示することが要求される。しかしこの画像生成には膨大なデータを高速に演算することが必要であり、従来試みられた方式では実用的に満足する時間内で処理することは困難であった。

本研究は可視化における画像生成処理の高速化を目的として、以下の手順で並列処理方式の導入を行っ

た。

(1) 様々な画像生成アルゴリズムに内在する並列性として、機能分割、データ分割、画面領域分割とフレーム分割を考慮し並列モデルを構築した。

(2) このモデルを対象に多層並列処理に基づく装置を試作し、複数の可視化処理プログラムを動作させた。

そして本並列処理方式の定量的評価と定性的評価を行い、併せて並列固有の問題であるソフトウェア開発環境について評価し、可視化処理における本方式の有効性と今後の課題を確認した。

2. 画像生成処理

2.1 可視化処理の内容

一般に画像生成処理とはアプリケーションが固有に持つ数値データから、与えられた視点や光源情報とともに画像を出力デバイスに生成することである。さらに入力デバイスを操作しアプリケーションプログラムと対話する処理も含まれる。科学技術計算プログラムが内部的にもつ可視化対象データとして、次の表現形式がある。

(1) 2次元/3次元空間内の格子点でのスカラまたはベクタ値。

(2) 構造的関係と位置に関する情報。

(3) 2次元/3次元空間のポリゴンの集合。

可視化処理ではまずこれらのデータから画像生成に適した幾何モデルを生成する処理が必要である。この処理をモデル生成処理と呼ぶ。

† A Parallel Method of Image Generation for Scientific Visualization and Its Evaluation by IKUO OYAKE, TAKAYOSHI YOSHIDA and YASUSHI WAUKE (Systems Laboratory R&D Group, Oki Electric Industry Co., Ltd.).

†† 沖電気工業(株)研究開発本部総合システム研究所

画像生成の処理方式としては、画面上の一点（画素）ごとに視点から光線を出し対象モデルとの交差点で色情報を計算する光線追跡法⁶⁾（レイトレーシング）と、対象モデルをポリゴン近似しこのポリゴンを基本的に座標変換、色情報の計算を行う物体変換法²⁾がある。可視化処理では、モデル生成処理が容易でかつ光線追跡法と比較して高速性があることから物体変換法が多く利用されている。物体変換法はポリゴンベースに光の反射モデルを近似するもので、

- (1) Constant Shading
- (2) Gouraud Shading
- (3) Phong Shading
- (4) Radiosity Method

などの濃淡付けモデル²⁾が提案され、利用目的に合った画像が生成できるが計算量は順次多くなる。また物体変換法では3次元空間の立体感を出すため、隠面消去法や Depth Cueing¹⁾ 線画表示が使用されている。計算量の桁数としては、約 10 万ポリゴンのモデルを Gouraud Shading により 1K×1K 解像度の CRT に準動画 (10 画面/秒) 表示する場合、最低 100 MFLOPS の能力を必要とする。光線追跡法ではさらに計算量が 100 GFLOPS 以上必要となる。

2.2 従来方式の問題

可視化処理へ従来多く適用されてきた物体変換法のアーキテクチャを見直し、従来方式の拡張性における問題点を明確にする。

以下に物体変換法による画像生成の処理過程を示す。

- (1) アプリケーションプログラムが固有に持つデータを幾何モデルであるポリゴンで近似する。
- (2) ポリゴンは局所座標系で表現されていることがあり、世界座標系に統一するため必要なモデル変換を行う。このとき光源情報からポリゴンの各頂点での色情報の計算を行う。さらに視点情報をもとにビューイング変換を行う。

(3) 画面は矩形領域であるので、その枠の外にでるポリゴン領域をクリッピングする。

(4) クリッピングされたポリゴンに隠面処理を行い、濃淡付けモデルに基づきポリゴン内部のピクセル（画素）の色情報を計算し画像メモリに書き込む。

以上の処理を、従来は逐次処理するか複数の演算ユニットでパイプライン的に実行していた⁷⁾。具体例としてグラフィック WS の Geometry Engine⁸⁾ における幾何処理を対象とした 12 段のパイプライン処理が

ある。パイプライン実行をもとに可視化処理全体の高速化をはかる場合、次のような問題に直面する。

(1) N 段のパイプライン処理は各ステップの処理時間が等しくなったとき最大の性能がでる。しかし各ステップの処理時間は、ポリゴンの形状（頂点数）やサイズ（画面でのピクセル数）に大きく依存する。様々なポリゴンが大量に発生する可視化処理では、このパイプラインに乱れが生じ本来の性能が達成されない。

(2) それぞれの可視化処理プログラムにおいて、ポリゴン近似に要する処理や、利用する様々な濃淡付けモデルからも各ステップの処理内容と処理時間が変化する。それゆえ各ステップの処理時間の短縮化だけ追求するアプローチには限界がある。

このようなパイプライン方式に見られる高速化の壁を打破するため、様々な並列処理への試みがなされてきた。

中規模、大規模並列に向けた汎用並列処理装置は分散メモリ型の MIMD タイプとデータ並列型の SIMD タイプが開発された。前者のタイプでは NCUBE⁹⁾、iPSC¹⁰⁾ などが、また後者のタイプとして Connection Machine¹¹⁾ などが市販されている。可視化処理への応用を考えたとき、まずその膨大な計算量をカバーするために Scalability を有する計算モデルの確立が重要となる。さらにアーキテクチャからみれば MIMD マシンはモデリングや座標計算などの数値計算に向いているが通信オーバーヘッドの問題があり、かつピクセル展開での膨大な画素単位演算に弱い点が指摘される。

また SIMD マシンは画素演算や座標変換など大量データへの同一演算の繰返しに効力を発揮するが、モデル生成処理と画像生成処理内部で生じる大量のデータ転送処理に不向きな点がある。

一方、画像生成向け専用マシン¹²⁾も多く提案されてきた。ここでの特徴は初めて画像生成の並列アルゴリズムを考案し、その処理アルゴリズムとデータフローに適したアーキテクチャを開発している。光線追跡法を対象とした例に LINKS¹³⁾、SIGHT¹⁴⁾、CAP¹⁵⁾ マシンがある。またポリゴンのスキャンコンバージョン（ピクセル展開）の並列アルゴリズムに基づく HPRG¹⁶⁾、EXPERT¹⁷⁾、PIXEL¹⁸⁾ マシンなどがある¹⁹⁾。これらは表示対象により異なる様々なモデル生成処理や、線面を含む基本的な表示処理から高度なレンダリング機能を要求する可視化処理の範疇では必然

的に効果が限定される。

本研究は科学技術計算における可視化処理を対象に、モデル生成処理、座標変換やレンダリング処理などの膨大な計算能力が必要な処理を、メッセージ通信²⁰⁾に基づく高レベル並列処理方式と新素子デバイスにより高速化することを目標としている。並列処理方式ではポリゴンベースの物体変換法を基本におき、様々な画像生成アルゴリズムにも共通した並列性に着目し、かつ Scalability を追求した計算モデルを検討した。

3. 並列モデル

3.1 画像生成モデルの分割

画像生成の抽象モデルは可視化を対象に考えると、

$$I_k = F(E_k, GD_k) \quad (1)$$

で表現される。ここで k はフレーム（連続画面のなかの1画面を意味する）番号である。各フレームに関し画像生成関数 (F) は、ビュー情報や光源などの環境情報 (E) と表示対象となるデータ (GD) を入力とし、表示画面 (I) を出力する。出力が入力の関数となることから、画像生成は一種のフィルタとみることができる。以下に画像生成に内在する分割可能な並列性を順次検討する。

(1) 機能分割

パイプライン方式による固定分割ではなく、画像生成処理を演算の種類により2つに分割する方法をとる。1つはモデル生成処理、座標変換やビューイング変換などを行う浮動小数点演算が主体の幾何計算処理である。他は2次元画像メモリをアクセスする画素演算が主体のピクセル処理である。機能分割により式(1)の関数 F は以下のように表現される。

$$F = F_g \cdot F_p \quad (2)$$

ここで F_g は幾何計算処理であり、 F_p はピクセル処理である。 F_g の出力は F_p の入力となる。この両者のインタフェースを中間データ (DD) と呼ぶ。 DD の内容は物体変換法ではポリゴン表現が基本である。光線追跡法の場合は画素データ集合となる。 DD の詳細な情報は F_g と F_p の処理内容により決められる。機能分割は次の分割に進むための準備となる。

(2) データ分割

可視化処理の対象となる各フレームのデータ (GD) を分割する。例えばポリゴンの集合を $(n+1)$ 個の部分集合に分割する。そして異なる部分集合への幾何計算処理を並列に実行する環境を考える。

$$\begin{aligned} F_g &= (F_{g0}, F_{g1}, \dots, F_{gn}), \\ &= \{F_{gi}, i=0, n\}. \end{aligned} \quad (3)$$

図1に分割モデルを示す。この分割により新たに必要となる通信は、各 F_{gi} にデータの部分集合を分配すること、環境情報 (E) を各 F_{gi} にブロードキャストすること、および各 F_{gi} は処理結果の DD を F_p に転送することである。 Z バッファ法のように隠面消去処理が F_p で解決される場合、各 F_{gi} は幾何計算の途中での相互干渉はなく独立に計算できる。光線追跡法では光線と交差する物体の最適配置問題が生じ、自身のメモリにない物体は相互の転送が必要となる。光線追跡法の DD は画素集合となり対応する画像メモリ F_p に転送される。

(3) 画面領域分割

画面領域空間 (I) を部分 (矩形) 空間に分割し、各部分空間 h にピクセル処理機能 f_{ph} をもたせる方式である。

$$\begin{aligned} F_p &= (f_{p0}, f_{p1}, \dots, f_{pi}), \\ &= \{f_{ph}, h=0, l\}. \end{aligned} \quad (4)$$

一般に光線追跡法による並列処理に用いられる方法であるが、ポリゴンベースの物体変換法に適用すると、部分画面にまたがるポリゴンの特別な処理が必要になる。この処理には以下の2通りが考えられる。

(a) F_g の出力をすべての f_{ph} にブロードキャストする。 f_{ph} は担当する部分空間に包含されるポリゴン領域だけをピクセル展開する。

(b) F_g の処理に部分空間を考慮したクリッピング機能を追加し、部分画面にまたがるポリゴンを分割する。分割されたポリゴンはそれが属する部分空間を担当する f_{ph} に転送する。

前者(a)は通信と f_{ph} の処理の負荷を増大させ、後者(b)は F_g の処理の負荷を増大させることになる。

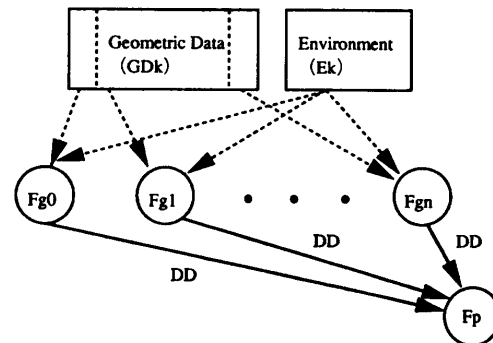


図1 データ分割
Fig. 1 Data decomposition.

(4) フレーム分割

これは従来ダブルバッファ法²⁾と呼ばれた方式の拡張として考えられる。(m+1)画面分の表示空間(画像メモリ)と各表示空間にピクセル処理機能を持たせる(図2)。

$$Fp = (Fp_0, Fp_1, \dots, Fp_m),$$

$$= \{Fp_j, j=0, m\}. \tag{5}$$

そして各フレームkの処理を $j = k \bmod(m+1)$ の Fp_j に割り当て、ピクセル展開をフレーム別に並列に実行させる。アニメーション表示では各表示空間を順次切替表示する制御が必要となる。これを1つの Fp_j が集中制御すると、mの増大に伴い同期制御の通信負荷が増える。図2に画面切替におけるトークンパッシング²⁰⁾を用いた分散制御方式を示す。すなわち表示トークンを設定し Fp_j 間を順番に巡回させる。表示トークンを獲得することは画面切替と表示の権限を得ることであり、一定時間表示後このトークンを次に送信する。表示トークンが他の Fp_j を巡回している間に、次のフレームに関するピクセル処理を実行する。

3.2 並列モデル

以上の分割を可視化処理の並列モデルに取り入れる。すなわち各フレームごとに式(3)、式(4)で示されるデータ分割、画面領域分割を行い、フレーム全体には式(5)のフレーム分割を行う並列モデルは以下の式で表現される。

$$F = \{\{Fg_{ji}, i=0, n\} \cdot Fp_j, j=0, m\} \tag{6}$$

$$= \{\{Fg_{ji}, i=0, n\} \cdot \{fp_h, h=0, l\}_j, j=0, m\}. \tag{7}$$

この並列モデルは、全体として2段(幾何計算処理とピクセル処理)の階層フィルタで構成され、そのなかで $\{Fg_{ji}, i=0, n\}$ と Fp_j の関係はマルチクライアントとサーバモデルとなる。ここでフレームkの幾何計算処理は $j = k \bmod(m+1)$ に対応する $\{Fg_{ji}, i=0, n\}$ の間で並列処理(データ分割)されることになる。 $\{Fp_j, j=0, m\}$ 全体は内部で画面切替に関する同期制御を行う分散サーバとなり、各 Fp_j はクライアントである $\{Fg_{ji}, i=0, n\}$ に対しフレームの開始終了を通知し、DDを受信しピクセル処理を行う。また各 Fp_j は画面分割により $\{fp_h, h=0, l\}_j$ で並列処理される。以上の並列モデルを図3に示す。

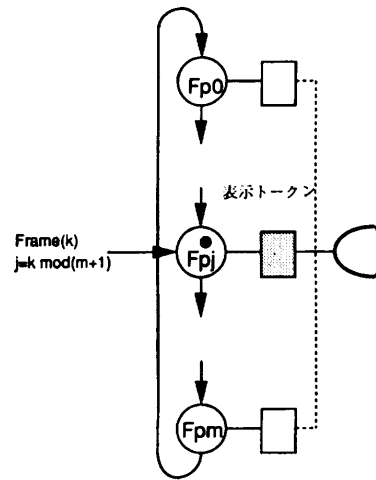


図2 フレーム分割
Fig. 2 Frame decomposition.

4. 並列処理アーキテクチャ

以上の並列モデルに基づき可視化処理プログラムを実行する並列処理装置を試作した^{21), 22)}。本装置の並列アーキテクチャの特徴として以下の2点があげられる。まず物体変換法における幾何計算処理とピクセル処理を効果的に並列処理するために前者に MIMD タイプを後者には SIMD タイプの並列方式を導入した。また幾何計算処理部内のデータ転送速度を高速化するためプロセッサ間の通信ポートに新素子 (GaAs) を採用した。さらに試作の前提条件として並列度(プロセッサ台数)に制約を設けたこと、および装置本体は汎用ワークステーション(以下 WS と呼ぶ)のバツ

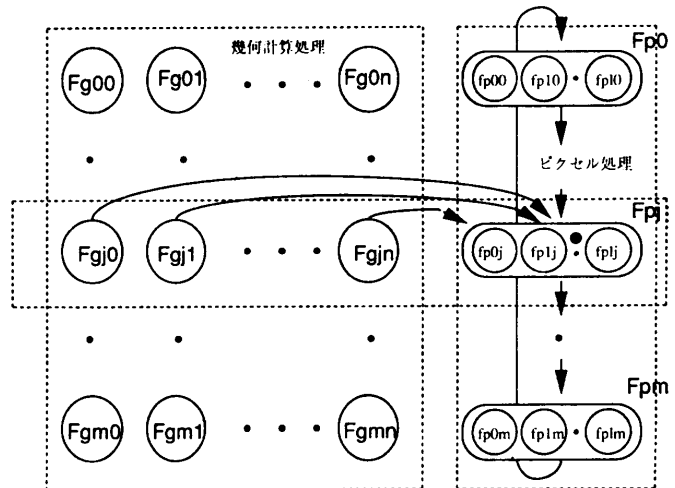


図3 並列モデル
Fig. 3 Parallel model.

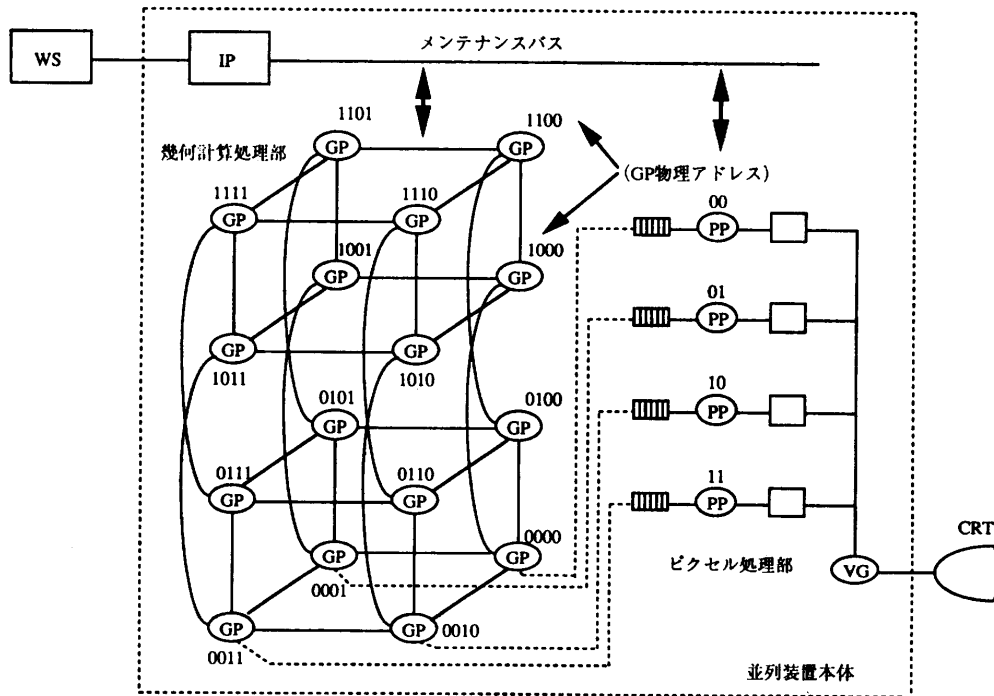


図 4 システム構成
Fig. 4 System configuration.

表 1 ハードウェア仕様概略
Table 1 General hardware specification.

構成要素	仕様
WS	-UNIX ワークステーション -Disk 容量/800 MByte -LAN Interface
IP	-マイクロプロセッサ内蔵 -WS とは SCSI Interface で接続
GP 8, 16 台構成	-32 bit RISC プロセッサ (8 Mips) データメモリ 4 MByte コードメモリ 2 MByte -浮動小数点プロセッサ (8 MFlops) -専用通信処理プロセッサ -通信用 LSI/Bit Serial, フレーム同期方式, 2ノード間通信速度 160 Mbps
PP 4 台構成	-専用グラフィックプロセッサ -Zバッファメモリ内蔵 -画像メモリ/1,024*1,024 画素, (R, G, B, I) 構成 -25 Mpixel update/sec
CRT	-20*20 インチ大画面

クエンド装置として動作することをあげておく。

4.1 ハードウェアアーキテクチャ

全体システム構成を図 4 に、ハードウェアの外部仕

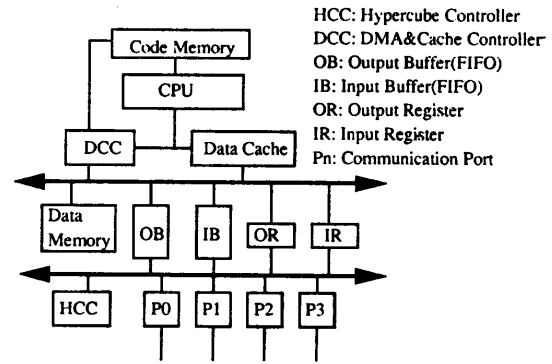


図 5 GP ノード構成
Fig. 5 GP node structure.

様を表 1 に示す。全体システムは WS, 並列処理装置本体および CRT から成る。装置本体は幾何計算処理部とピクセル処理部に分れる。

(1) 幾何計算処理部

幾何モデルの各種演算と座標変換, クリッピング処理を実行する幾何計算処理部は, 浮動小数点演算と通信処理を強化した最大 16 台のプロセッサノード (以下 GP と呼ぶ) から構成される。各 GP ノードの構成は図 5 に示すように演算部と通信部のバスを分離し, FIFO で両者を接続している。これにより演算部側での通信の負荷を軽減している。GP 間の結合はネット

ワークトポロジに柔軟性を持つハイパーキューブ接続方式²³⁾⁻²⁵⁾を採用している。GP 間は中間データ (DD) をパケットとして短時間に多回数送受信するので、データ転送の高速化をはかるため GaAs デバイスで通信ポート (送信, 受信 2 種) を構成している。転送速度は 160 Mbps で 40 bit/frame のビットシリアル転送を実現する。この通信ポートの制御に、データリンクレベルのプロトコル処理とネットワークのルーティング処理を実行する専用の通信制御用 LSI を開発した²⁶⁾。

(2) ピクセル処理部

DD を入力しピクセルに展開するピクセル処理部は、図 6 に示すように 4 台のピクセルプロセッサノード (以下 PP と呼ぶ) と 1 台のビデオ発生ユニット (以下 VG と呼ぶ) から構成される。各 PP は濃淡付け処理と隠面消去処理 (Z バッファ法) を行うプロセッサを内蔵し、画像メモリ構成としては 1,024×1,024 画素の R, G, B, I, Z フィールドを有する。VG は (R, G, B) と I の乗算をビデオ発生時に行う機能をもつ。さらに VG は 2bit のカウンタをもち、その値に対応した PP から画像データを読み出す。各 PP はカウンタの値を独立に操作できるが、アクセス競合はフレーム分割で述べた表示トークンを利用し解決する。

図 7 に 1 台のピクセルプロセッサノードの構成を示す。このプロセッサはプログラム可能な専用 LSI であり、ビューイング変換後のポリゴンデータのピクセル展開処理などの機能を有する。濃淡付け処理と Z バッファ制御は 16 台のコプロセッサを SIMD 方式で並列動作させ 16 画素同時に演算処理する。Constant および Gouraud Shading はこれらのコプロセッサ側のハードウェア機能で実行される。その他ピクセルプロセッサ機能として線画表示や画素データの画像メモリ転送機能がある。

(3) 処理部間インタフェース

各 PP は任意の GP とそれぞれ FIFO バッファを介し接続する。4 台の PP 間には直接的な接続がない

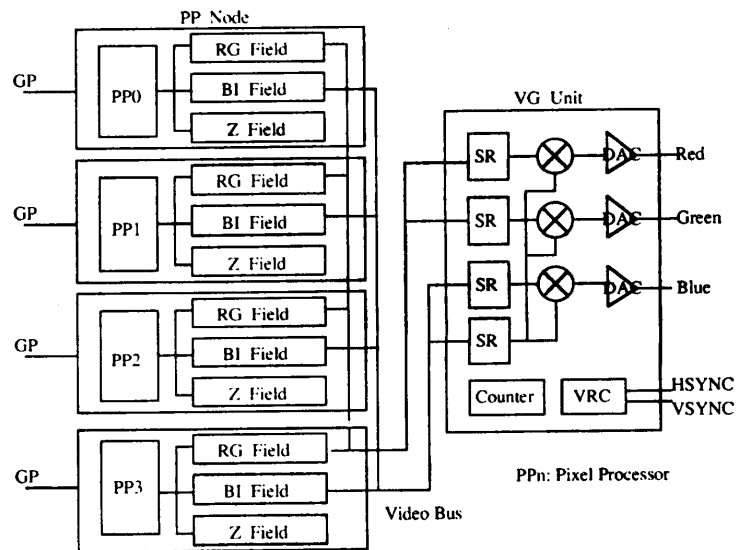


図 6 ピクセル処理部の構成
Fig. 6 Pixel processing block.

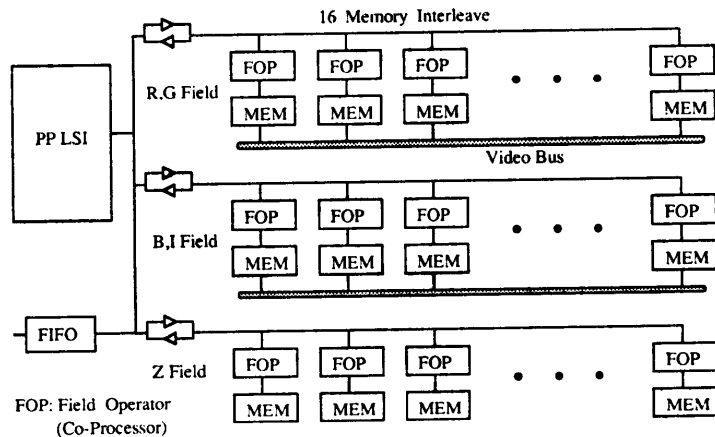


図 7 SIMD 型ピクセルプロセッサ
Fig. 7 SIMD-type pixel processor.

ため、各 PP に接続する GP が相互に通信し画面切換制御を代行する。また装置本体と WS 間にはインタフェースプロセッサ (以下 IP と呼ぶ) が介在する。この IP はメンテナンスバスによりすべての GP と PP に接続しており、プログラムロード、診断テストおよび実行時における通信とエラー処理に使用される。

4.2 ソフトウェアアーキテクチャ

全体のソフトウェア構成を図 8 に示す。WS 側には実行監視プログラムおよび可視化処理プログラム (以下 VP と呼ぶ) の一部があり、装置本体には並列モニタ、表示処理ライブラリ、画面制御プログラムおよび VP が搭載される。

(1) 実行監視プログラム

実行監視プログラムは装置本体の初期化、プログラムロード、AP の実行開始、装置本体からのメッセージ通信要求の監視、および AP の終了、異常終了の処理を行う。装置本体の初期化では GP 台数 (8, 16) と GP の物理アドレス/論理アドレスの変換テーブルを指定する。物理アドレスは装置のslot番号から決まり図4に示される。論理アドレスはメッセージ通信で利用されるプロセス番号に相当する。実行監視プログラムは WS の UNIX* 上の1プロセスとして動作する。

(2) 並列モニタ

各 GP に搭載されるモニタ機能には、プロセス間のメッセージ通信、自ノード内の資源 (メモリ、タイマ) 管理およびエラー処理がある。GP で実行される VP は1プロセスとして動作する。プロセス間 (WS 上の実行監視プロセスを含む) のメッセージ通信の基本はパケットである。パケットはその長さにより短パケット (4 Byte) と長パケット (1,024 Byte) の2種類があり、処理方式により即時処理パケットと順処理パケットに分れる。メッセージ通信におけるアドレスは、プロセス番号を用いる。プロセス番号と GP との対応は、装置構成指定の変換テーブルで指定される。メッセージの送受信はコネクションレスプロトコルの非同期通信であり、送信された順処理パケットは受信側モニタ内のバッファに一時的に格納され、受信プロセスのメッセージリードでこの待行列がスキャンされ、該当メッセージが読み出される。即時処理パケットは割り込み機能で処理される。

エラー処理は、実行時にシステムエラーを検出したモニタが IP に対しエラー通知メッセージを送る。これを受けた IP はメンテナンスバスを介し全 GP にエラー発生をブロードキャストし、GP 内の特定メモリ領域のスナップショットをとる。

(3) 表示処理ライブラリ

このライブラリは定義された3次元ポリゴン (閉ポリゴン、線画用開ポリゴン) を視点情報をもとに座標変換とクリッピング処理を行い DD を生成する機能を提供する。3次元ポリゴンの定義では各頂点 (X, Y, Z) に色情報が追加される。出力である DD の色表現には用途により2つのモードがある。1つは (R,

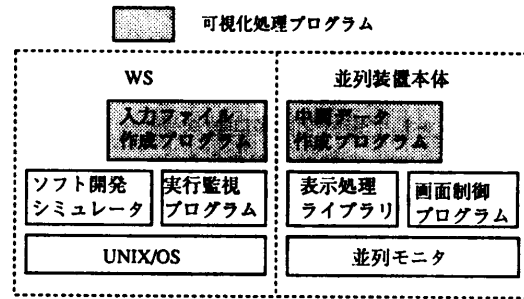


図8 ソフトウェア構成
Fig. 8 Software structure.

G, B) モードで、他は (r, g, b, I) モードである。ここで $R=r \times I$ の関係 (G, B も同様) がある。ポリゴンの各頂点に直接異なった色情報が与えられる場合は前者を、各頂点は同一の色情報を持ち光源や深度 (Depth) の影響を扱う時は輝度を含む後者のモードを使う。この場合輝度はライブラリ内で計算により求められる。またこれらの色指定に対し濃淡付け処理の指示は、線描画と面描画ともに可能である。

(4) 画面制御プログラム (以下 SCP と呼ぶ)

これは PP に接続する4台の GP 上に搭載され、他の GP から送信された DD を PP の FIFO バッファに転送しながら、フレーム分割における画面切換制御を実施する。4台の GP 上のプログラムが協調し、他の GP ノードに対し分散サーバを構成する。

(5) ソフトウェア開発シミュレータ

VP の開発、デバッグを直接並列装置本体で行うことは困難を伴う。そのため WS 上に並列装置のシミュレータを構築した。シミュレータの構成は1つの制御プロセスと AP プロセス群から成る。制御プロセスがユーザからの実行コマンドを受け、任意の AP プロセスの起動、中断、停止などの処理をする。さらに AP プロセス間のメッセージ通信の履歴などの表示を行う。このシミュレータにより、AP のデバッグを個別の AP プロセスのデバッグから、徐々に並列動作を含む複数の AP プロセス間のデバッグに拡張していくことができる。

5. 可視化処理プログラム

以上の基本的なソフトウェアを用い3つの VP を並列装置本体に搭載した。以下に VP の概要を示す。

(1) プラズマ磁気流体シミュレーション^{27)**}

原子炉内のプラズマの状態の時系列的な変化を表示する。プラズマの状態は等圧部を線画により表現する。はじめに平衡状態 (図9(a)) だったプラズマが

* UNIX は米国 AT&T 社ベル研究所で開発されたソフトウェアの名称です。

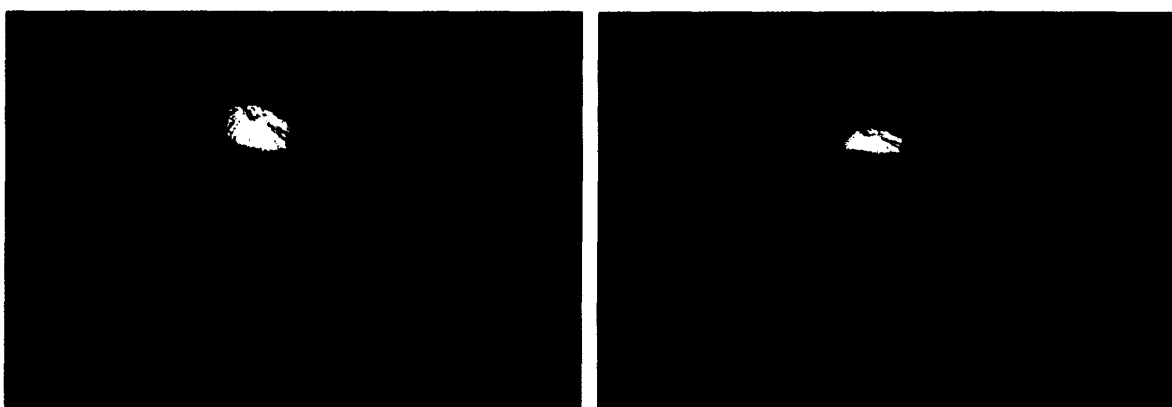
** 名古屋大学佐藤哲也教授からデータ提供された。



(a)

(b)

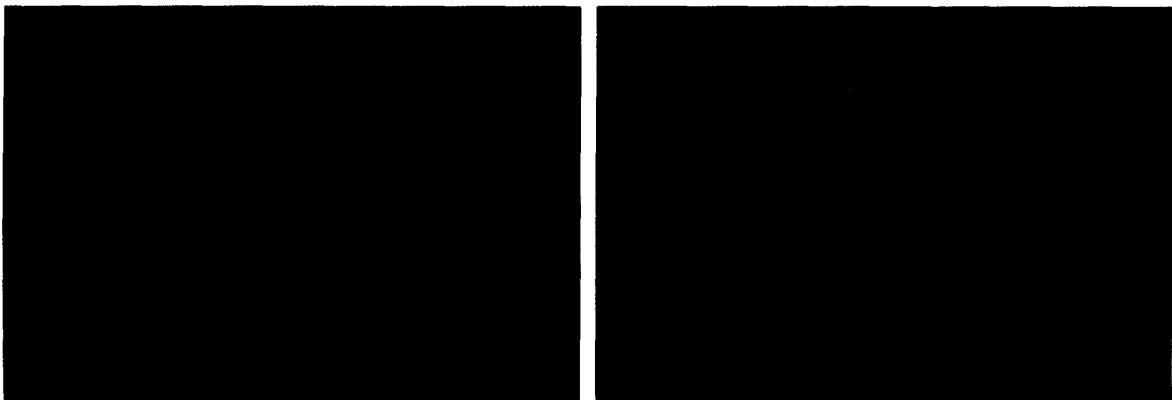
図 9 プラズマ流体の表示
Fig. 9 Display of plasma.



(a)

(b)

図 10 衛星画像の表示
Fig. 10 Display of satellite image.



(a)

(b)

図 11 分子振動の表示
Fig. 11 Display of molecular modeling.

徐々に不安定 (図 9 (b)) になり変形していく様子をアニメーション表示する。

(2) 資源探査用衛星画像の3次元表示²⁸⁾⁻³¹⁾

人工衛星から送られた画像データが濃度歪補正, 幾何補正の画像処理を受け, 地図の高さ情報 (Z 値) とマージされた3次元地形データが VP への入力となる。視点を連続的に変化 (図 10 (a, b)) させることにより, 地形の3次元表示を行う。

(3) SO₂ 振動シミュレーション³²⁾*

SO₂ 分子の分子間振動 (分子間の位置関係) および O-S-O 間の原子間振動 (距離, 角度) を時系列的に表示する。入力は時系列の原子の位置情報で, 形状は GP 上で実行時に生成される。図 11 に分子の異なった形状表現例を示す。アニメーション表示では視点を固定し振動する分子集団全体を回転している。

5.1 VP プログラムの構成

図 8 のソフトウェア構成で示したように WS と並列装置本体に2種類のプログラムが搭載される。

(a) 入力ファイル作成プログラム (IFP)

これは可視化データから GP への入力 (GD データと視点, 光源情報) を生成し, 台数 (8, 16) 指定により各 GP に分配するデータファイルを作成する。

(b) DD 作成プログラム (IDP)

各 GP に分配されたデータファイルから表示処理ライブラリを用いて幾何計算処理を行い DD を作成する。フレームの開始と終了の制御はこのプログラム内で SCP とのメッセージ通信により行う。

8 台および 16 台構成におけるプロセスの配置を図 12 に示す。図 12 では以下の説明の都合上, GP 集合の部分集合としてフレーム分割に関する系列とデータ分割に関するグループの定義を行っている。8 台構成には 4 系列, 1 グループが含まれ, 16 台構成には 4 系列, 3 グループが含まれる。

5.2 VP プログラムの処理内容

(1) プラズマ流体シミュレーション

シミュレーション結果として, 約 70 フレーム分の

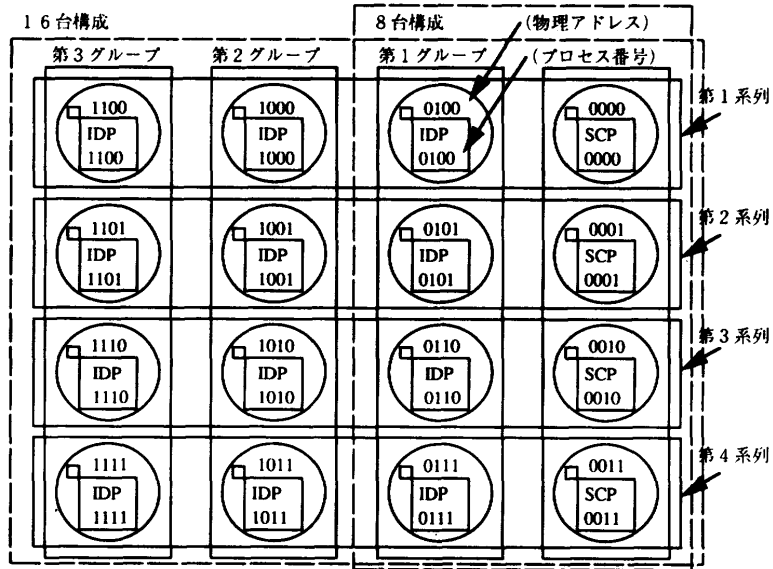


図 12 プログラム配置
Fig. 12 Program allocation.

データが出力される。1 フレームのデータは, 円筒座標系の各格子点の圧力値データにより与えられる。IFP において, 指定された圧力値の等高線となるポリゴンデータを図 13 に示すように垂直断面ごとに作成する。1 フレームを構成するポリゴンデータの容量は, 補間断面数 90, 1 断面当たりのポリゴンの線分数を約 180 とすると, 約 130 KByte となる。70 フレーム分のポリゴンデータを 4 系列, 3 グループの GP 群に分割すると, 各 GP 内で約 700 KByte のデータ領域を占有する。

IDP 内では 3 色 R, G, B のスポットライトによる色彩表示の効果を出すため, 図 13 内に示すように各光線とポリゴン線分との角度から r, g, b 成分を決定

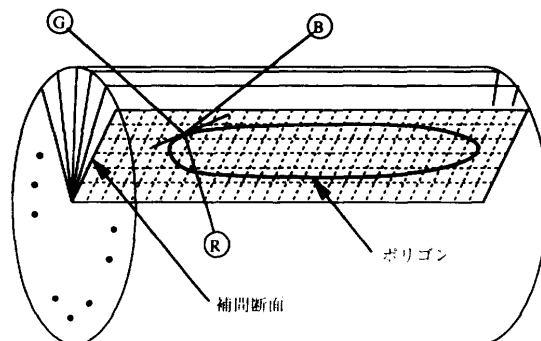


図 13 プラズマ流体のデータ表現と表示方式
Fig. 13 Data representation and rendering techniques in plasma.

* 京大大学町田勝之輔教授からデータ提供された。

している。さらに Depth Cueing 線画表示を行うため Z 値から I 値を計算し、(r, g, b, I) モードによる線面の濃淡付け処理を DD 内に指示している。

(2) 衛星画像の3次元表示

入力となる幾何データは各節点の (X, Y, Z, R, G, B) 値の画像データである。256×256 節点の容量は約 780 KByte である。アニメーション表示では視点データが時間の経過と共に変化するので、IFP では 4 系列の GP 群に対し同一の画像データとフレームごとに変化する視点データを作成する。GP 16 台構成では同一の画像データを 3 グループの GP に分割するので、約 260 KByte のメモリ領域を各 GP 内で占有する。

図 14 に示すように、IDP では 3 角形ポリゴンの (R, G, B) モードによる面描画の濃淡付け処理を DD 内に指示している。

(3) SO₂ 振動シミュレーション

IFP への入力データは、総フレーム数 (3,000) と 1 フレーム内に存在する分子数 (約 125) と各原子の中心座標値で構成される。1 フレームのデータ容量は約 5 KByte である。図 11 (a) のようにすべての分子を一度に表示すると原子間の関係が観察しづらいので、ユーザインタフェースの観点から分子数の間引き処理 (例えば 1/10) をパラメータ指定で行っている (図 11 (b))。またフレームの間引き処理も行う。4 系列の GP 群にフレームを順番に割り当て、16 台構成では各フレームの分子を 3 グループの GP に分割している。

IDP では各原子の球形状を 3 角, 4 角ポリゴンで実行時に近似し、(r, g, b, I) モードによる面描画の濃淡付け処理を DD 内に指示している。

6. 総合評価

6.1 性能測定

3つのVPの並列装置本体における性能測定を行った。まず幾何計算処理部とピクセル処理部のインタフェースとなるFIFOを実行時に監視することにより負荷状況を調べた。その結果8台と16台構成ともに幾

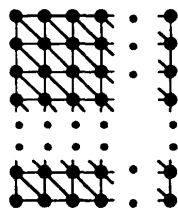


図 14 衛星画像表示

Fig. 14 Display of satellite image.

何計算処理部側に多くの負荷がかかっていた。これは次にみるデータ分割の効果を導く前提となっている。

16 台構成は図 12 のプロセス配置に示されるように 8 台構成に比べ各データを 3 グループに分割し並列処理している。3つのVPにおいてそれぞれ1フレームの生成にどの程度の処理時間を要するかを表2に示す。8台と16台構成の比較で結果はほぼ3倍に近い性能が達成されており、データ分割の効果を確認することができる。

フレーム分割の効果は既に8台構成(4系列)に含まれている。一般に(m+1)系列にしたとき表示トークンの巡回方法から計算上m倍の性能を達成することができる。表3の8台構成における全体の実行時間からこの性能を確認することができる。ここでフレーム分割の効果で示した値(T)は、フレーム全体を逐次処理した場合の時間(T1)を表2の1フレーム処理

表 2(a) プラズマ流体のフレーム処理時間
Table 2(a) Frame generation time in plasma.

	フレーム 1	フレーム 2	フレーム 3	フレーム 4
8 台 構 成	3.4(sec)	3.4(sec)	3.3(sec)	3.4(sec)
16 台 構 成	1.1(sec)	1.1(sec)	1.1(sec)	1.1(sec)
並列効果	3.1	3.1	3.0	3.1

表 2(b) 衛星画像のフレーム処理時間
Table 2(b) Frame generation time in satellite image.

	フレーム 1	フレーム 2	フレーム 3	フレーム 4
8 台 構 成	171.5(sec)	170.2(sec)	174.6(sec)	182.6(sec)
16 台 構 成	56.9(sec)	61.5(sec)	60.7(sec)	59.6(sec)
並列効果	3.0	2.8	2.9	3.1

表 2(c) 分子振動のフレーム処理時間
Table 2(c) Frame generation time in molecular modeling.

	フレーム 1	フレーム 2	フレーム 3	フレーム 4
8 台 構 成	12.4(sec)	12.3(sec)	12.3(sec)	12.3(sec)
16 台 構 成	3.9(sec)	3.9(sec)	3.9(sec)	3.8(sec)
並列効果	3.1	3.1	3.1	3.2

表 3 アニメーション処理時間
Table 3 Execution time in animations.

	Plasma (72 frames)	Satellite image (16)	Molecular modeling (20)
8 台 構 成	81.7(sec)	877.8(sec)	89.6(sec)
フレーム分割効果	3.0	3.2	2.7
16 台 構 成	26.9(sec)	283.1(sec)	30.0(sec)
Total 実行時間比	3.0	3.1	3.0

時間 (平均) と全体フレーム数の積から求め、それをフレーム分割の実行時間 (T_2) で割って算出している ($T=T_1/T_2$)。さらに表 3 に 8 台, 16 台構成での Total な実行時間比を示す。

6.2 並列モデルの定量的評価

以上の測定結果からみて、プロトタイプ装置の並列規模では実際の様々な可視化処理アニメーションをリアルタイムに実現することは依然不可能といえる。プロセッサ速度の向上も今後確実に予想されるが、ここでは台数をさらに 32 台, 64 台, ..., 1,024 台と増加させたときの定量的評価を行ってみる。式 (7) で表現される並列モデルのパラメータ n, m, l の値はプロトタイプ装置では、

$$n=2, m=3, l=0$$

であり、画面領域分割は用いていない。

台数増加に伴う実行性能を予測するため、式 (7) で表現される並列モデルの各構成要素の共通性能指標として、秒当たり何ポリゴン処理できるかをトランザクションの単位として考える。

T_g ; 各 F_g で生成されるポリゴン数/秒

T_p ; 各 F_p で展開されるポリゴン数/秒

T_c ; 各 F_p に F_g 群から転送可能なポリゴン数/秒
3つの VP プログラムの実行時 (16 台) において、

$$T_g \geq 1,000$$

$$T_p \geq 3,000$$

$$T_c = 24,400$$

の測定値が得られている。 T_g はモデル生成処理を含み T_p はポリゴンの大きい (画素数) による変化量を考慮し、どちらも VP 共通な最小値をとっている。 T_c は通信パターンを VP 実行と同一な条件としパケット送信に最大の負荷をかけたときの各 F_p の平均受信可能ポリゴン数である。

T_c の台数増加による変化を算出するため、通信パターンとして通信網に最大負荷をかける環境を設定する³³⁾。すなわち次元数 p で最大距離ノード間のペア (2^{p-1} ペア) が相互にパケットの送受信を最大限行った場合を考える。この条件が網の通信制御部内でパケットの相互干渉が最も多く発生するケースと見なされる。次に本装置 ($p \leq 4$) を用いた上記環境下での通信速度の測定値を示す。

$$p=1 \text{ のとき, } v=0.18 \text{ msec}$$

$$p=2 \text{ のとき, } v=0.23 \text{ msec}$$

$$p=3 \text{ のとき, } v=0.53 \text{ msec}$$

$$p=4 \text{ のとき, } v=0.99 \text{ msec}$$

$p=1$ のときはパケットの衝突処理のオーバーヘッドが現れているが、全体の変化から判断すると最大負荷時における平均速度はノード間距離の 2 乗に逆比例しているといえる。各 F_p に F_g 群から転送される VP 実行時の通信パターンは、単方向通信であることから明らかなように、パケット相互干渉は上記最大負荷時の環境より少ない。そこで台数増加による T_c の外挿を、最大負荷時における値を利用して算出することを考える。まず上記最速ノード間の通信速度 (測定値) の次元数拡張として次の配列を設定する。

$$V(10) = (0, 0.18, 0.23, 0.5, 1, 2, 4, 8, 16, 32, 64).$$

r 項 ($0 \leq r \leq 10$) は距離 r のノード間の最大負荷時における 1 パケットの平均転送時間 (msec) を示す。

次にデータ分割 (n) と新たに画面領域分割 (l) を増加させた時の T_c を以下に計算する。画面領域分割は 3.1 節 (3) 項で述べた (b) 方式を用いる。このとき F_g の処理内容に新たな画面領域分割によるクリッピング処理が追加されるので、全体のハイパーキューブノードを F_g 台数 $>$ F_p 台数 $\times 3$ となるようにサブハイパーキューブに分割する。次元数 P ($3 \leq p \leq 10$) では

$$F_p \text{ 台数 } (np) = 2^{p-3},$$

$$F_g \text{ 台数 } (ng) = (2^p - 2^{p-3})$$

のノード数分割を選ぶ。すなわち $ng/np = 7/1$ の比である。

また次元数 p のハイパーキューブで、あるノードから距離 r にあるノード数を示す配列を定義する。

$$A(10)_r = (i_0, i_1, i_2, i_3, i_4, \dots, i_{10}).$$

$$\text{ここで } ir = (r) \quad r \leq p \leq 10,$$

$$ir = 0 \quad p < r \leq 10.$$

このとき次元数 p の F_g 群から各 F_p への 1 パケットの平均通信時間は次式で計算される。

$$v = (A(10)_p - A(10)_{np}) * V/ng.$$

ここで項 $(A(10)_p - A(10)_{np})$ は F_p から距離 r の F_g ノード数を表し、それと V との内積は通信時間による重み付けを行っている。さらに ng で割ることにより正規化し、平均通信時間を算出している。

それゆえ平均転送パケット数は $10^3/v$ となり、また平均転送ポリゴン数は、

$$T_c = (10^3/v) * 4$$

で表現される。次元数と平均ポリゴン転送能力の関係をグラフにすると図 15 で示される。この図には ng/np 比が 7/1 だけでなく、3/1, 1/1 についても比較のため図示している。

並列モデルの計算で通信ネックを起こさないために

は、少なくとも $T_0 < T_c$, $T_p < T_c$ の関係を成立させる必要がある。図 15 から $p \leq 7$ の範囲で並列モデルのデータ分割と画面領域分割が可能といえる。それ以上はフレーム分割を導入することになる。式(7)で、

台数 512 のとき, $n=111$, $h=15$, $m=3$,

台数 1,024 のとき, $n=111$, $h=15$, $m=7$

となる。得られる並列効果は 16 台構成に比較し、512 台構成で最大 16 倍、1,024 台構成で最大 37 倍となる。

以上の分割は 1 つの方法であり、フレーム分割を増やす構成も考えられる。フレーム分割は並列効果が大きいであるが実装されるメモリ量も増大するので、ここではデータ分割、画面領域分割を最大限利用した並列分割案を提示している。

6.3 並列モデルの定性的評価

本並列モデルは並列装置本体および可視化処理プログラムを設計するための重要な指針となった。モデルの検討のなかで今回対話機能については簡略化した。実際にはアニメーション表示の途中で、順表示を静止させたり逆表示することも考えられる。さらに視点情報を変更したり、表示スピードを制御する機能が望まれる。このためには本並列モデルの画面切替制御部分を中心に拡張することが必要となる。少しモデルは複

雑になるがこの拡張は可能であり、本並列装置上でこれらの対話機能を含むプログラムを実現することも可能である。

さらに本装置上に搭載した VP のレンダリング処理では、陰影や反射などの写実性を追及したアルゴリズムは用いていない。本並列モデルに従い、例えばレイトレーシング表示を実現することも可能であるが、可視化処理への応用としては、必要な計算量から上記の台数増加は必須であり今後の課題といえる。

6.4 ソフトウェア開発環境

ソフトウェア開発環境ではシミュレータが特に有効であった。論理的なデバッグは大半このシミュレータ上で行うことができた。一方、実機上の唯一のデバッグ手段はシステムのエラー処理機構を利用する方法であった。利用者にとってさらに使いやすい環境とは、実機上にこのシミュレータ環境を載せプロセスの実行制御とメッセージのモニタリングを可能とすることである。さらに一步進んでメッセージ起動型の実行制御³⁴⁾も有効と思われる。また並列モデルの定量的評価で述べたような実行時における各種性能測定ツールの充実も今後必要不可欠である。

また構成台数に関連したデバッグの容易性について述べる。本装置上では 8 台構成でプログラムを動作させ、次に 16 台構成にプログラムを移行させた。初めに 8 台構成でプログラムを並列動作させることに對し、16 台への移行は比較的容易であった。ハイパーキューブのサブ集合で動作させたプログラムをさらに高次元の集合で動作させることは、拡張性を持つ並列モデルを厳密に確立しておけばそれほど困難さを伴うものではないであろう。

7. おわりに

本論文では画像生成の観点から可視化処理の機能を明確にし、その高速化にあたり画像生成に内在する分割可能な並列性に着目した並列処理方式を開発した。この並列方式は高レベル並列処理 (Coarse Grain) であり、かつ Scalability を追求したものである。

そして実際の可視化処理プログラムを開発し、プロトタイプ装置での並列処理の効果を確認し、台数増加による性能評価を行った。今後のプロセッサと通信処理の性能向上、および実装技術の向上による台数効果を考慮すると、膨大な演算を要する可視化処理のアニメーション表示も、ユーザが分散環境での並列装置上で手軽に行えるようになる。一方、ソフトウェア開発

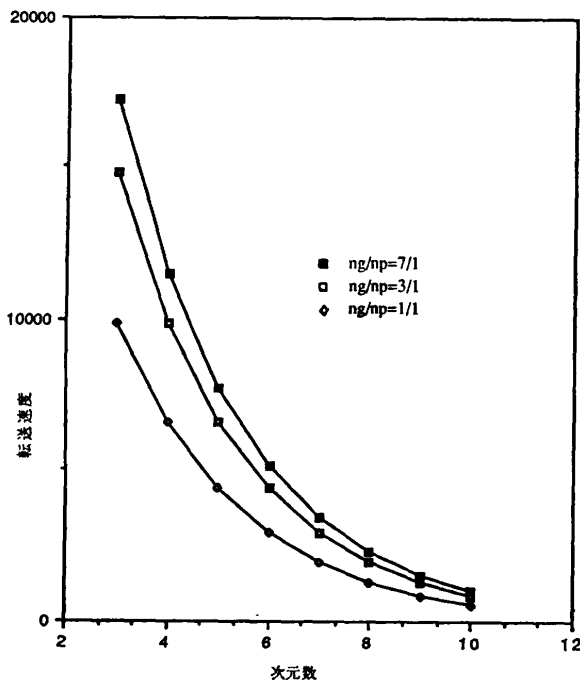


図 15 ポリゴン転送速度
Fig. 15 Polygon transfer speed.

環境ではより高度なツールの研究も必須である。このテーマは可視化処理に限定されず並列処理技術を様々な応用分野で普及させる上でも重要な課題と思われる。

謝辞 本研究開発の大部分は、通産省工業技術院大型プロジェクト「科学技術用高速計算システムの研究開発」の一環として、沖電気工業(株)が新エネルギー・産業技術総合開発機構 (NEDO) から委託を受けて実施したものです。本研究に貴重な意見をいただいたプロジェクト関係者に深く感謝します。

参 考 文 献

- 1) Newman, M. W. and Sproull, F. R.: *Principles of Interactive Computer Graphics*, Second ed., pp. 6-7, pp. 293-307, pp. 369-371, McGraw-Hill (1979).
- 2) Foley, D. J., Dam, V. A., Feiner, K. S. and Hughes, F. J.: *Computer Graphics*, Second ed., pp. 1-24, pp. 201-284, pp. 337-338, pp. 649-813, Addison-Wesley (1990).
- 3) 大宅ほか: 高機能ワークステーション, 情報処理, Vol. 24, No. 10, pp. 1247-1254 (1983).
- 4) DeFanti, A. T., Brown, D. M. and McCormick, H. B.: Visualization, *IEEE Computer*, pp. 12-25 (Aug. 1989).
- 5) Frenkel, A. K.: The Art and Science of Visualizing Data, *CACM*, Vol. 31, No. 2, pp. 111-121 (1988).
- 6) Whitted, T.: An Improved Illumination Model for Shaded Display, *CACM*, Vol. 23, No. 6, pp. 343-349 (1980).
- 7) Myers, W.: Staking Out Graphics Display Pipeline, *IEEE Computer Graphics and Applications*, pp. 60-65 (Jul. 1984).
- 8) Clark, J. H.: The Geometry Engine; A VLSI Geometry System for Graphics, *SIGGRAPH*, pp. 127-134 (1982).
- 9) Hayes, J. P. and Mudge, T.: Hypercube Supercomputers, *Proc. IEEE*, Vol. 77, No. 12, pp. 1929-1941 (1989).
- 10) 阿部: ハイパーキューブ・マルチプロセッサ, *bit*, Vol. 21, No. 5, pp. 640-651 (1989).
- 11) ダニエル・ヒリス: コネクションマシン, パーソナルメディア (1990).
- 12) 新實, 富田: グラフィックスとその専用マシン, *bit*, Vol. 17, No. 10, pp. 73-80 (1985).
- 13) 出口: 並列画像生成システム LINKS, *bit* 臨時増刊, 並列コンピュータ・アーキテクチャ, 共立出版, 東京 (1990).
- 14) 吉田ほか: グラフィック計算機 SIGHT の基本構成, 情報処理学会計算機アーキテクチャ研究会, Vol. 85, No. 53, 60-4 (1985).
- 15) Sato, H., Ishii, M. et al.: Fast Image Generation of Constructive Solid Geometry Using a Cellular Array Processor, *SIGGRAPH*, Vol. 19, No. 3, pp. 95-102 (1985).
- 16) 和宇慶ほか: 3次元色彩表示専用プロセッサ HPRG, 情報処理学会計算機アーキテクチャ研究会, Vol. 85, No. 15, 58-4 (1985).
- 17) Niimi, H., Murakami, Y. et al.: A Parallel Processor System for Three-Dimensional Color Graphics, *SIGGRAPH*, Vol. 18, No. 3, pp. 67-76 (1984).
- 18) Potmesil, M. and Hoffert, M. E.: The Pixel Machine: A Parallel Image Computer, *SIGGRAPH*, Vol. 23, No. 3, pp. 69-78 (1989).
- 19) Swanson, W. R. and Thayer, J. L.: A Fast Shaded-Polygon Renderer, *SIGGRAPH*, Vol. 20, No. 4, pp. 95-100 (1986).
- 20) Andrews, R. V.: Paradigms for Process Interaction in Distributed Programs, *ACM Computer Surveys*, Vol. 23, No. 1, pp. 49-90 (1991).
- 21) 和宇慶ほか: 並列画像生成方式とそのアーキテクチャ, 情報処理学会計算機アーキテクチャ研究会, Vol. 73, No. 10 (1988).
- 22) 守谷ほか: 3次元画像表示装置, 信学会, CPSY 90-62, pp. 145-150 (1990).
- 23) Seitz, L. C.: The Cosmic Cube, *CACM*, Vol. 28, No. 2, pp. 22-33 (1985).
- 24) Peterson, C. J. et al.: The Mark III Hypercube-Ensemble Concurrent Processor, *Proc. Int'l Conf. on Parallel Processing*, pp. 71-73 (1985).
- 25) Reed, A. D. and Grunwald, C. D.: The Performance of Multicomputer Interconnection Networks, *IEEE Computer*, pp. 63-73 (Jun. 1987).
- 26) 小山ほか: 高速ハイパーキューブ網と制御方式, 並列シンポジウム JSPP '80, pp. 367-374 (1989).
- 27) 佐藤: 核融合プラズマ研究と3次元コンピュータシミュレーション, *bit*, Vol. 19, No. 13, pp. 200-211 (1987).
- 28) 宮田ほか: Distributed Parallel Processor for Satellite Image Processing (1), 信学技報, Vol. 90, No. 100, pp. 25-32 (1990).
- 29) 前田ほか: Distributed Parallel Processor for Satellite Image Processing (2), 信学技報, Vol. 90, No. 100, pp. 33-40 (1990).
- 30) 和宇慶ほか: Distributed Parallel Processor for Satellite Image Processing (3), 信学技報, Vol. 90, No. 100, pp. 41-48 (1990).
- 31) 杉村ほか: ランドサット TM データおよび数値地形モデルを使った土地景観の立体画像作成, 日本リモートセンシング学会誌, Vol. 5, No. 4, pp. 97-100 (1985).
- 32) Hayashi, S. et al.: Molecular Dynamics Using Internal Coordinates, I. Infrared Spectra of

Sulfur Dioxide, *The Chemical Society of Japan*, Vol. 58, No. 4, pp. 1105-1108 (1985).

- 33) 和宇慶ほか: ハイパーキューブネットワーク制御とその評価, 信学会, CPSY 91-5, pp. 9-16 (1991).
- 34) McDowell, E.C. and Helmbold, P.D.: Debugging Concurrent Programs, *ACM Computing Surveys*, Vol. 21 No. 4, pp. 593-622 (1989).

(平成3年8月26日受付)

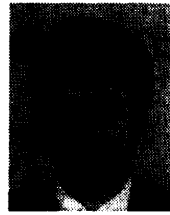
(平成4年5月14日採録)



大宅伊久雄 (正会員)

昭和22年生. 昭和44年京都大学理学部数学科卒業. 同年沖電気工業(株)入社. CAD/CAM, グラフィックス, 分散処理, 並列処理の研究開発に従事. 現在, 総合システム研究

所に勤務. ACM 会員.



吉田 隆義 (正会員)

昭和21年生. 昭和46年東京工業大学理学部電気工学科卒業. 同年沖電気工業(株)入社. 水中音響映像, 信号処理, 3次元グラフィックスの研究開発に従事. 現在, 総合シ

ステム研究所に勤務.



和宇慶 康 (正会員)

昭和32年生. 昭和55年琉球大学工学部電気工学科卒業. 昭和58年名古屋工業大学大学院電子工学修士課程修了. 同年沖電気工業(株)入社. 総合システム研究所にて, 3次

元グラフィックス, 並列処理, マルチメディア処理の研究開発に従事. 電子情報通信学会会員.