

ネットワークベースの分散処理を用いた並列図形演算システム

3 P-6

辻本英二 平尾謙次 植田照生
(株) 日立製作所

1 はじめに

LSI のマスクパターン作成や設計規則検証において、図形演算と呼ばれる、レイアウトパターンデータに関する図形処理が頻繁に使用される。

近年は LSI の規模増大に伴い処理対象の図形量が $10^7 \sim 10^8$ ベクトルに増大し、図形演算時間の高速化が強く望まれている。

そこで、複数のワークステーションをネットワークで接続した、ネットワークベースの分散処理システムを開発し、図形演算処理において評価した。

2 図形演算

図形演算には、論理演算、位相演算、寸法チェック演算などがある。

論理演算はパターン実体間の和(OR)、積(AND)、差(SUB)、排他和(EOR)などの領域のパターンを生成する。

位相演算はパターン間の位相関係を、重なり合い(MEET)、パターン全体の一致(MATCH)、などに分類する演算である。

寸法チェック演算はパターンの幅(WIDTH)、異なるパターンの間隔(SPACE)などについて指定された判別式を満足する個所を検出する。これらの演算はマスクパターンデータの検証およびファイル作成処理（アートワーク）工程において、マスクパターンデータを対象とした種々の CAD システムにおいて使用されており、アルゴリズムは処理速度および使用メモリ量の点で優位であるスリット法[1][2]が主流となっている。

3 並列図形演算システム

3.1 ハードウェア構成

ハードウェア構成の概念図を図 1 に示す。処理装

置は 1 個のホスト処理装置と複数個のスレーブ処理装置に分けられる。各処理装置は Ethernet などの通信路により結合され、相互に通信することが可能である。両処理装置は 2 次記憶装置としてそれぞれ、グローバルディスク、ローカルディスクを備えている。

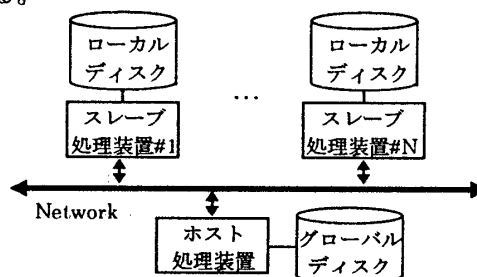


図 1 ハードウェア構成

3.2 ソフトウェア構成

ソフトウェア構成を図 2 に示す。ハードと同じくソフトも、ホスト処理装置上で動作するクライアントプログラムと、スレーブ処理装置上で動作するサーバプログラムに分けられる。実行時には、1 個のクライアントプログラムと複数個のサーバプログラムが同時に実行され、クライアントプログラムにより全体の制御が行われる。また、プロセス間の通信には Socket を用いた。

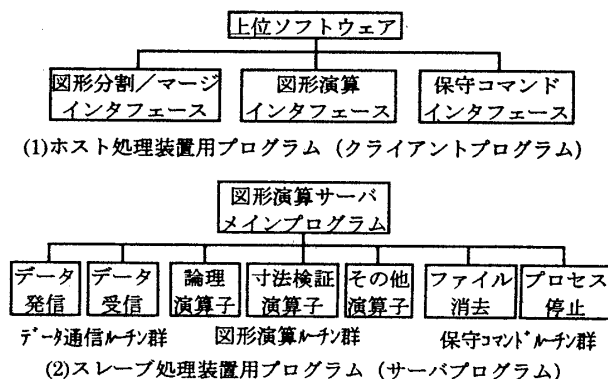


図 2 ソフトウェア構成

4 並列図形演算処理

並列図形演算処理の流れを図 3 により説明する。

Parallel Pattern Operation System using Distributed Network
Eiji Tsujimoto, Kenji Hirao, Teruo Ueda, Hitachi, Ltd.,

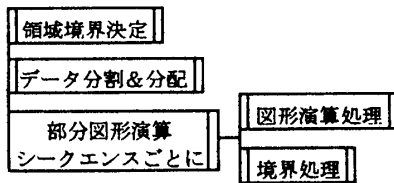


図3 処理フロー

処理は領域境界決定、データ分割&分配、図形演算処理、境界処理、の順番で行われる。

領域境界決定は並列処理の負荷分散に関するものであり、各領域に含まれるデータの数がほぼ等しくなるように境界を決定するとともに、初期境界マージン値だけの重複幅を持たせて領域を設定する。

次にこの境界に従ってデータを分割し、スレーブ処理装置に送信する。

図形演算処理では、図形演算を領域毎に独立に実行する。図形演算処理はシングルプロセッサで処理する場合のプログラムをそのまま利用するが、演算子によっては、その演算子を通常に処理した後で分割境界に関する補正処理（境界処理）を必要とする。また、他の演算の結果を入力とする時、境界近傍の図形が不正確である場合は入力図形から不正確な部分を切除することが必要である（入力切断処理）。図形演算処理は複数演算子からなる演算シーケンスであってもよい。境界処理の不要な演算子の演算結果データは、直接他の演算子に受け渡すことができるので、続く演算子も同じ図形演算処理に含めることができる。しかし、境界処理を要する場合は、続く演算子は別の図形演算処理で実行しなければならない。境界処理をはさまず、1回の図形演算処理で実行される図形演算シーケンスを、部分図形演算シーケンスと呼ぶ。

境界処理では、図形演算処理の結果データに関し、複数領域にまたがって存在する図形番号の統一、分割領域内での図形の位相情報から全体の位相情報の合成等を行う（整合性の確保）。また、最終結果については、分割データをマージして、逐次処理の図形演算システムで処理した場合と同等の結果を合成する。境界処理では各スレーブ処理装置の同期を必要とするので、処理速度の問題から、部分図形演算シーケンスの個数は少ないほうが良い。この

方法では、所望の演算シーケンスの処理に対し、データ分割を1回だけ、データマージを最終結果の数だけ行うという点に特徴がある。1演算子ごとに分割・演算・マージを行う方法は、演算ごとに入力/結果データを通信することを意味し、連続演算と比べ非常に低速となる。

5 評価結果

本システムで電子線描画データ作成処理を実行した結果を表1に示す。

表1 評価結果

データNo.	1	2	3
入力ベクトル数	4,545,454	875,225	538,035
逐次	cpu time (sec)	26,174	6,614
	elapsed time (sec)	54,530	10,335
並列	total elapsed time (sec)	2,480	557
	分割 (sec)	370	63
	演算 (sec)	1,849	422
	マージ (sec)	261	72
逐次処理比高速化率	21.99	18.55	9.18

スレーブ処理装置には Sun Sparc Station IPC を16台使用し、ネットワークには Ethernet を用いた。また、演算シーケンスは20演算子で構成されている。

データ1および2がデータ3に比べて高速化率が非常に高くなっているが、これは領域を分割したことにより取り扱いデータ量が減少したため、ディスク上で行っていたソート処理をメモリ上で行うことが可能となったためである。

6 おわりに

LSI のアートワーク工程で使用される図形演算処理について、並列処理を用いた実用レベルの実験システムを開発した。電子線描画データ作成処理にて評価したところ、逐次処理比9~22倍の高速化を達成し、良好な結果を得た。

【参考文献】

- [1] U. Lauther, "An $O(N \log N)$ Algorithm for Boolean mask operations." proc. 18th DAC, pp.555-562 (1981).
- [2] 築添明, 他, "VLSI マスクデータに対する論理演算と交点計算を同時処理するパターン論理演算手法" 信学論(D) pp.975-983 (1986).