

3D-9

分布系シミュレーションシステム DISTRAN の
並列計算機 AP1000 への実装細川 達己 鈴木 清弘 朴 泰佑 川合 敏雄
慶應義塾大学 理工学部

はじめに

近年のデバイス技術の向上により、比較的安価で高性能なマイクロプロセッサを多数用いた並列計算機の実現が可能となってきた。だが計算機を専門としない一般の研究者にとって計算機を用いた数値計算は、プログラミングの複雑さ、解法の選択、結果の妥当性の評価などについて専門的な知識が必要であり、必ずしも容易ではない。さらに並列計算機上のプログラミングは逐次計算機上のそれに比べ、効率的なプログラムを作ることが難しく、その性能を引き出すことはさらに困難である。

このような背景をもって作られたシステムが DISTRAN (Distributed Systems TRANslator) である [1]。DISTRAN は、数値計算に関して専門的な知識のないユーザにも、複雑なプログラミングをすることなしに、信頼性の高いシミュレーションを提供することを目的としている。

DISTRAN は、時間に関して1階、空間に関して2階までの連立偏微分方程式で表される2次元分布系問題を所定の様式で記述することにより、並列計算機用のプログラムを自動生成する。

DISTRAN は数値計算のアルゴリズムとして陽解法を採用している。陽解法は他の解法と比べて計算量が多いという欠点はあるが、近接性、自律性、同時性を兼ね備えていることから並列化に適しており、その欠点は大規模並列計算機によって解決できる [2]。

DISTRAN はすでに並列計算機 MiPAX および Transputer 上に実装されている [3]。今回 DISTRAN を富士通研究所の並列計算機 AP1000 上に実装した。本稿ではその実装法及びシステムの基本性能に関する評価について報告する。

並列計算機 AP1000

AP1000 は、SPARC を CPU とするセル・プロセッサ(以下セルと略す)を多数結合したマルチプロセッサである。セル間結合のネットワークトポロジーとしては二次元トーラスが採用されているが、ハードウェア・ルータが用意されているため、ソフトウェア上でデータのルー

ティングを意識する必要はない。それぞれのセルはまた、ブロードキャストのためのネットワークと同期のためのネットワークで結ばれている。各セルには 16MB のメモリと 128KB のキャッシュが用意されており、セルの台数は 64 ~ 1024 台となっている [4]。本実装においては 64 台構成のシステムを使用した。

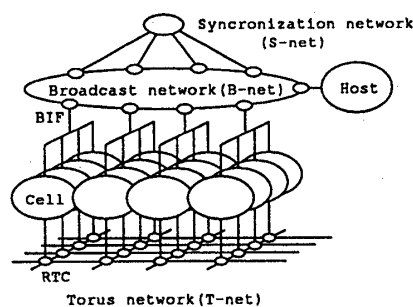


図1: AP1000 のアーキテクチャ

AP1000 におけるプログラミング言語は、C または FORTRAN である。これらの言語には並列ライブラリが用意されており、さらにデバッガ等のツールが用意されている。AP1000 におけるプロセスの並列化は、いわゆる並列化コンパイラによって自動的に行なわれるのではなく、ユーザが各セルの上で動作する逐次プログラムを記述し、それを全セルが並列実行することにより行なわれる。したがってユーザはプログラムを明示的に並列化して記述する必要がある。

セル間の通信、セル・ホスト間の通信は全てメッセージパッシングに基づいて行なわれ、明示的な同期操作はほとんど要求されない。また、先述のルータの働きにより、隣接していないセル間のメッセージパッシングもそのまま記述できる。さらにネットワークトポロジーにおいても、ユーザは「物理的なセルの数を越えない」というゆるい条件の範囲で、仮想的な2次元トーラスを自由に設定することが可能となっている。

DISTRAN/AP1000 の実装

DISTRAN は大きく分けてトランスレートプログラムとテンプレートプログラムの2つの部分から構成さ

れている。トランスレートプログラムは Prolog によって記述されており、ユーザが書いた仕様書から、問題固有のプログラム及びデータが C 言語の形で生成される。これに、やはり C 言語で書かれたテンプレートプログラムを組合せてコンパイルすることにより、並列計算機上で実行可能なプログラムが生成される [3]。

どこまでの処理をセルで行なうかという判断は非常に重要である。現在の DISTRAN/AP1000 においては、ホスト側で境界の判定、データの出力を行ない、セル側で時間発展の計算、境界の計算をそれぞれ行なっている。

性能の評価

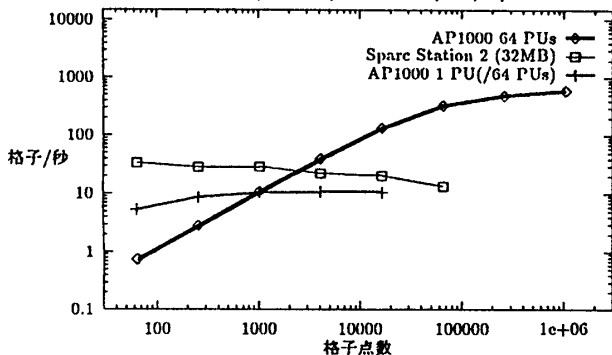
DISTRAN/AP1000 の性能を、次のような観点から評価した。

1. 問題規模を大きくしていった場合の効率
2. 問題規模を一定にし、使用セル数を増やしていった場合の効率
3. コミュニケーションコスト

問題規模と計算速度との関係を図 2 に示す。これはプログラムの立ち上がり時間、出力時間の部分を取り除くため、同じ出力回数で空洞内の流体を 1000 ステップと 2000 ステップ計算し、その計算時間の差から実際の計算速度を見積もったものである。

1PU あたり同数の格子点を含む場合として、 $128 \times 128 = 2^{14}$ 格子を持つ問題を解かせた時の 1PU AP1000 の計算速度と、 $1024 \times 1024 = 2^{20}$ 格子を持つ問題を解かせた時の 64PU AP1000 の計算速度を比較すると、前者の性能は 10.5 [格子/秒] であるのに対して、後者は 606 [格子/秒] と 57.7 倍の速度を達成している。したがってこの場合の効率は $606 / (10.5 \times 64) = 90.2\%$ であり、かなり高い。

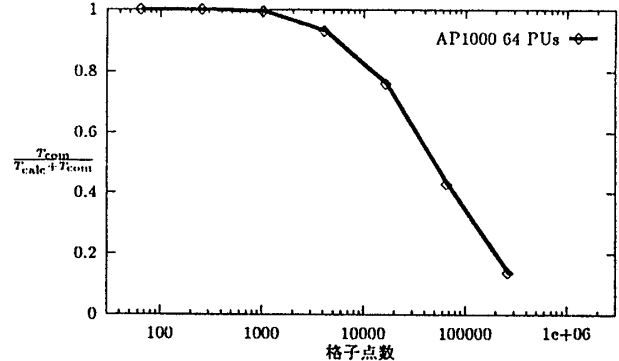
図 2: 問題規模 [格子点数] と計算速度 [格子/秒] との関係



さらに、並列化に伴うコミュニケーションのオーバーヘッドを定量化するために、 $T_{com} / (T_{calc} + T_{com})$ という比を測定した。ここで T_{com} はセル間コミュニケーションにかかった時間、 T_{calc} は計算にかかった時間である。この結果を図 3 に示す。512 × 512 = 2^{18} 格子の問題を

解いた場合、この値は 13.8% となっており、比較的オーバーヘッドは小さいと言える。

図 3: 問題規模 [格子点数] とコミュニケーションオーバーヘッドの関係



今後の課題

現在の DISTRAN/AP1000 は領域判定や境界条件の判定をホスト側で行なっている。しかしこれは AP1000 のような大規模マルチプロセッサにおいてはホスト側に巨大な配列をを要求することになる上に、動的に変化する境界を扱うことができない。これらの処理をセル側に移行したい。

また、現在は数値データをそのままテキストファイルとして出力しているが、大規模計算においてはこれらはしばしば数百 MB にもなることがあるので、もっと効率の良い出力方式を検討したい。

おわりに

分布系シミュレーションシステム DISTRAN を並列計算機 AP1000 上に実装した。計算はかなり高速であり、大規模な問題に対してはかなり高い効率で数値計算を行なうことが可能である。

参考文献

- [1] M. Kubota, et. al., "Automatic Distributed Systems Simulator by Nature Algorithm Using Highly Parallel Computer," 1st Int. Conf. on Supercomputing in Nuclear Applications, 1990.
- [2] 星野 力, "PAX コンピュータ —高並列処理と科学計算—", オーム社, 1985.
- [3] 鈴木 清弘 他, "DISTRAN システムの並列計算機上への実装", 並列処理シンポジウム JSPSP '91 論文集, 1991.
- [4] T. Horie, et. al., "AP1000 Architecture and Performance of LU Decomposition," 1991 Int. Conf. of Parallel Processing, 1991.