ブロードキャストを利用した大規模行列の緩和法解析の分散処理

1X−7

小石 昇　千種 康民　矢吹 道郎　孫 堅
上智大学 東京工科大学 上智大学 上智大学

1 はじめに

大規模行列の緩和法解析においては，各行を個別に計算することが可能なため，疎結合で接続された複数台の計算機を用いた負荷分散による解析の高速化が可能になる．この点に着目し，線形方程式を空間分割し，細分化された処理を，疎結合のバス型ネットワーク上の複数台の計算機に割り当て，負荷分散を試みる．この場合，各計算機における更新値を迅速に他の計算機に反映させることと，この通信の際の低コストの通信方式の実現が高速化のキーとなる．

そこで本研究では，まず，UNIX の通信プログラムに関して深い知識を持たなくとも，数値解析プログラマが数値解析プログラムを容易に分散化できる汎用性のある通信ライブラリを作成した．次に，これを用いて，通信コストの低い計算機間のデータ通信にブロードキャストを使用した緩和法解析の分散処理モデルを提案し，実験によりその有効性を確認した．

2 緩和法と分散処理モデル

2.1 緩和法アルゴリズム

線形方程式を，

$$\mathbf{Ax} = \mathbf{b}, \quad \mathbf{A} = \mathbf{L} + \mathbf{D} + \mathbf{U}, \tag{1}$$

ただし，

$$\mathbf{A} = [a_{ij}],$$

$$\mathbf{L} = [l_{ij}], \quad l_{ij} = \begin{cases} a_{ij} & \text{if } i > j \\ 0 & \text{otherwise,} \end{cases}$$

$$\mathbf{U} = [u_{ij}], \quad u_{ij} = \begin{cases} a_{ij} & \text{if } i < j \\ 0 & \text{otherwise,} \end{cases}$$

$$\mathbf{D} = [d_{ij}], \quad d_{ij} = \begin{cases} a_{ij} & \text{if } i = j \\ 0 & \text{otherwise,} \end{cases}$$

とするとき，ガウス・ヤコビ法における x についての反復式は，

$$\mathbf{x}^{(k+1)} = \mathbf{D}^{-1}(\mathbf{b} - \mathbf{L}\mathbf{x}^{(k)} - \mathbf{U}\mathbf{x}^{(k)}), \tag{2}$$

となる．一方，ガウス・ザイデル法における x についての反復式は，

$$\mathbf{x}^{(k+1)} = \mathbf{D}^{-1}(\mathbf{b} - \mathbf{L}\mathbf{x}^{(k+1)} - \mathbf{U}\mathbf{x}^{(k)}), \tag{3}$$

と表現される．ガウス・ザイデル法に注目すると，(3) 式の反復式が有効になるのは，x_i の添え字の小さい方から順に更新値を求める場合のみである．一方，その逆順で反復式を求めると，(2) 式のガウス・ヤコビ法と等価になる．それ以外の順序で (3) 式の反復式を計算するとき，$\mathbf{L}\mathbf{x}^{(k+1)}$の項で使用する $\mathbf{x}^{(k+1)}$は $x_i^{(k+1)}$と $x_j^{(k)}$の混在したものとなる．この場合の収束速度は，ガウス・ヤコビ法より加速されており，ガウス・ザイデル法より減速されることになる．

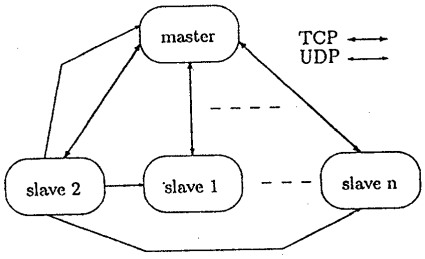

図 1: 緩和法解析の分散処理モデル

2.2 緩和法解析の分散処理モデル

緩和法アルゴリズムでは，求める値が収束するまで $\mathbf{x}^{(k)}$ の値を更新しながら反復計算を行なう．ここで (1) 式を空間分割し，複数の計算機を用いて負荷分散を行なう．このとき，各計算機における更新値を他の計算機に送信し，すべての計算機の持つ値を更新する通信方式として，通信コストの低い UDP のブロードキャストを使用する．緩和法の場合，データ通信が一部喪失しても，反復回数がわずかながら増加するだけである．

また，分散処理を行なう計算機において初期設定をするために必要な情報を送信する計算機を 1 台用意する．この通信には高い信頼性が要求されるので，TCP による通信を用いる．また，緩和法解析の収束判定も行なう．

この緩和法解析の分散処理モデルを図 1に示す．

以後，収束判定を行なう計算機をマスター，反復計算を行なう計算機をスレーブと呼ぶことにする．

3 通信ライブラリとブロードキャスト・ルーター

本研究では，各計算機間のデータ通信を容易に実現できるような通信ライブラリを作成し，これを用いて緩和法解析の分散処理を実現した．これらのライブラリは，ネットワークにおける通信プログラムに関して深い知識を持たない数値解析プログラマが，同様な数値解析プログラムを作成する際に容易に利用できるように汎用性のあるものにした．

緩和法では値の更新速度が速いほど全体の収束速度も速くなる．そのためブロードキャストによる更新値の通信は非同期に行なった．

以下に主な通信ライブラリとその機能を説明する．

setup_master, setup_slave
　マスターおよびスレーブのアドレスを設定する．

Distributed System for Relaxation Method using UNIX Broadcast
Noboru KOISHI
Faculty of Science and Technology, Sophia Univ.

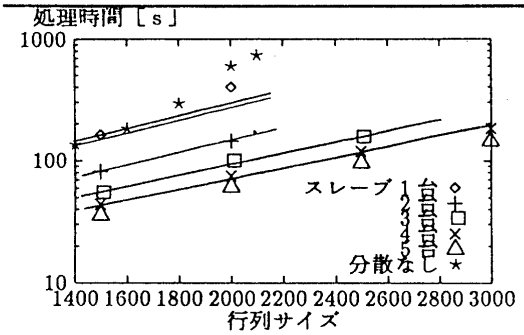

図 2: 行列サイズと処理時間

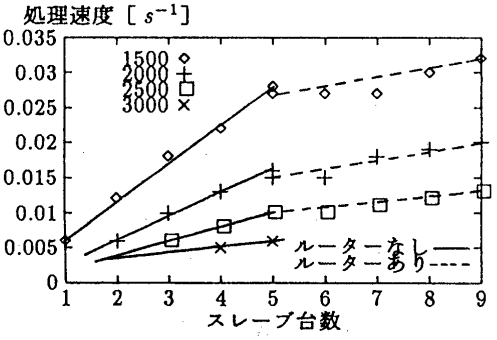

図 3: スレーブ台数と処理速度

wait_slave
　マスター側でスレーブとの TCP 接続を行なう.
connect_to_master
　マスターに対して TCP 接続の要求を行なう.
setup_broad_fd, get_common_fd
　ブロードキャストのためのソケットの設定を行なう.
send_to_slave, read_from_slave
　TCP を使ってスレーブとデータの送受信を行なう.
send_to_master, read_from_master
　TCP を使ってマスターとデータの送受信を行なう.
async_common_dg,async_master
　非同期ブロードキャストのための設定を行なう.
broad_send, broad_recv
　ブロードキャストを使ってデータを送受信する.

UDP のブロードキャスト通信は 1 つのネットワーク上でしか行なうことができないため, スレーブの台数に限界が生じる. そこで, ゲートウェイ上でブロードキャスト・パケットの交換を行なう, ブロードキャスト・ルーターを作成した. このブロードキャスト・ルーターは単に受信したブロードキャスト・パケットを, 反対側のネットワークに送り出すといった処理を行なっているだけなので非常に汎用性が高い.

4　実験結果

本実験は,10Mbps のイーサネットに接続された,SUN マイクロ・システムズの SPARC STATION1,SPARC STATION2,IPC,SLC,ELC 上で行なった. また各計算機の実記憶の容量は,12 〜 40Mbyte である.

1 つのネットワーク上, あるいはブロードキャスト・ルーターを用いて 2 つのネットワーク上で, スレーブの台数を変化させて処理時間を測定した結果を図 2, 図 3に示す. すべての測定は処理がメモリ上でのみ行なわれるように行列サイズを調節して行なった.

また比較のため, 図 2には分散処理せず 1 台の計算機で処理を行なった場合の処理時間も掲載した. 行列サイズが 1800 以上のものは仮想記憶が使用されているものである.

図 3においてスレーブ台数が 5 台までは 1 つのネットワーク上での測定, 6 台以降は 2 つのネットワークを使った測定である.

1つのネットワーク上での実験

図 2から分かるように, 分散処理を行なわない場合とスレーブが 1 台の場合では, 処理速度はほとんど変わらない. これは, ブロードキャストによる通信コストの低さを表しているといえる.

また, スレーブの台数が多くなるにつれて処理時間が短縮され, サイズが大きくなっても仮想記憶を使用することなく処理を行なうことができた. これにより, 従来大容量メモリが必要であった大規模行列の緩和法解析を, それほどメモリを搭載していない計算機を複数使って行なうことが可能であるといえる.

複数のネットワーク上での実験

図 3に示すように, 2 つのネットワークを使用した場合, スレーブ台数の増加に対する処理速度の増加の割合は下がるものの, 処理速度を増加することができた. これにより, 他のネットワーク上の計算機を利用して処理時間を短縮できるといえる.

5　まとめ

本研究では, 汎用性のある通信ライブラリを作成し, それらを用いてブロードキャスト通信を利用した緩和法解析の分散処理を実現した. また, ブロードキャスト・ルーターを使用することで, 2 つのネットワーク上の計算機を利用して分散処理を行なうことが可能になった. その結果, メモリアクセスの時間は,

　実記憶 < ネットワーク上の実記憶 << 仮想記憶

が成立し, 比較的メモリ容量の少ない計算機を複数使用することで, 緩和法解析の高速化を効率良く実現することができた.

参考文献

[1] 畠山正行, 矢吹道郎：プロセス間通信を用いた数値環境下でのシミュレーション及び数値計算の実現, 情報処理大会,6W-2, 昭和 6 2 年後期.

[2] 矢吹道郎, 畠山正行：UNIX LAN 環境における数値解析の分散処理,4th Annual Workshop SWoPP 大沼 '91,pp.9-16,1991.

[3] 清水, 程, 田中：緩和法プロセッサのパイプライン処理による回路シミュレーションの高速化, 信学技法, SDM-91-119, pp.57-64,1991.

[4] Y.Chigusa, M.Asai and M.Tanaka, "SRP:A Relaxation-Based Circuit Simulator with Error-Correcting-Learning Method by Adaptive Virtual Capacitors", IEEE ISCAS, pp.1149-1152, 1989.