

## 1. 序論

近年、コンピュータの急速な性能向上やプログラム技術の発展に伴って、処理される情報も膨大になってきたため、その情報を高速に処理する必要が高まってきた。この改善策として、複数の計算機を並列に繋ぎ処理速度を向上させる手法が注目されている。本研究では、無料で提供されている並列計算のソフトウェアの一つである MPI(Message Passing Interface)<sup>2)</sup> を用いてグラフ問題の一つである最小全域木問題に対する処理時間を計測し、MPIによる並列計算の有用性を検証する。

## 2. 研究内容

### 2.1 最小全域木問題

本研究では、MPIによる並列計算の有用性を検証するために最小全域木問題を並列処理を行う。最小全域木とは重み付無向グラフ  $G$  が与えられたとき、 $G$  の全ての接点を含む  $G$  の部分木のうち、辺の重みの総和が最小なグラフであり、最小全域木問題とは重み付無向グラフ  $G$  が与えられたとき、 $G$  の最小全域木を求める問題である。

### 2.2 研究環境

本研究では、同一機種の計算機 5 台を LAN 接続し MPI 環境を構築し、頂点数が 5,10,20,40 のそれぞれの重み付無向グラフ  $G$  に対し、計算機数 1~5 台を用いて MPI 上で最小全域木を解く時間を計測した。

### 2.3 処理内容

本研究で作成した最小全域木問題を解く MPI プログラムは Sollin のアルゴリズム<sup>1)</sup> を元に行っている。また、本研究で作成した MPI プログラムは、計算機のうち 1 台をホスト PC として用いる。ホスト PC は入力となる重み付無向グラフ  $G$  を作成し、 $G$  の部分グラフを各サブ PC にネットワークを通して送信する。サブ PC は送信されたデータに対し Sollin のアルゴリズムに基づき処理を行い、その処理結果をホスト PC に返送する。ホスト PC は返送された処理内容からグラフを再構成する。

## 3. 結果・考察

頂点数が 5,10,20,40 のそれぞれのグラフに対して、MPI 上で計算機 1~5 台を用いて最小全域木問題を解いた場合の内部処理時間を図 1 に、また、全体の処理時間を図 2 に示す。図 1 より、内部処理時間はサブ PC が増えるとともに速くなっていることが示される。しかし図 2 に示される

ように、全体の処理時間はサブ PC が増えるとともに遅くなっている。これは各サブ PC からのデータ通信時間が影響していると思われる。また、3 台から 4 台に変更時の急激な時間の増加は、複数の PC を繋げたため LAN の負荷がかかったためだと推測される。

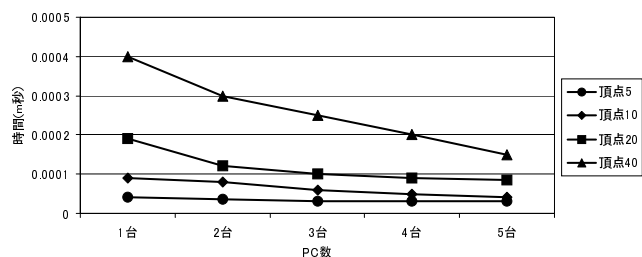


図1 内部処理時間と計算機数の関係

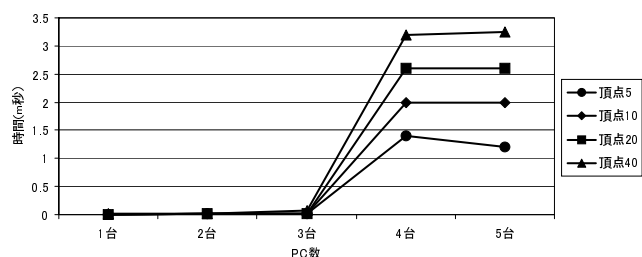


図2 全体の処理時間と計算機数の関係

## 4. 結論

本研究では、MPIによる並列化の有用性を検証するために MPI を用いて最小全域木を解く時間を計測した。しかし本研究では、MPIによる並列化によって内部演算の処理は向上には成功したが、それに伴う各 PC 間での通信時間により、全体の処理時間は計算機数を増やした場合、予想に反して時間がかかっており、MPIの有用性が充分示せたとはいえない。ネットワーク構築時の接続方法やデータ通信に係わるアルゴリズムの改良により、全体の処理時間の向上を得ることが今後の課題である。

## 参考文献

- 1) J.JáJá 著, An Introduction to Parallel Algorithms, Addison-Wesley Professional (1992).
- 2) P.Pacheco 著, 秋葉博訳, MPI 並列プログラム, 培風館 (2001).