

# 論文の内容の要旨

論文題目 並列計算機における相互結合網の高速化の研究

氏名 堀江 健志

現在、並列計算機は科学技術計算をはじめ、多くの応用で利用されている。そのなかでも、メッセージ通信で並列処理を行なう分散メモリ型並列計算機は、その拡張性、柔軟性、信頼性に優れ、ノードとして PC、ワークステーション、サーバなど多様な形態をとることが可能であり、多くのシステムに適用されている。並列処理の高速化には、各ノードの演算性能を向上させるだけでは応用性能は向上せず、通信性能の向上が必要である。そこで、本研究では、分散メモリ型並列計算機における相互結合網の性能向上について研究を行った。その結果、

1. 構造化チャネルルーティングによる低レイテンシ、高スループット、デッドロックフリーな通信方式の実現
2. トーラスネットワークでの最適な全対全通信方式の実現
3. メッセージ長に比例するメッセージ処理オーバーヘッドの削減
4. 並列化コンパイラが必要とする通信機能の実現

の 4 つの成果を得た。

並列計算機のノード間を接続する相互結合網は、その層構成をみると、物理層、リンク層、トランスポート層に分けることができる。物理層の機能として、信号伝送、コーディングがある。リンク層の機能として、

フロー制御, 仮想チャネル, ルーティングがある. また, トランスポート層の機能として, 送達保証, パケット化, メッセージ処理がある. 本研究では, リンク層とトランスポート層での性能向上について検討を行った.

構造化チャネルルーティングは, リンク層の仮想チャネルとルーティングに対応するものである. ワームホールルーティングは, ストアアンドフォワードに比べてレイテンシの小さい通信を実現できるが, スループットの低下とデッドロックの発生という 2 つの問題を持っており, これまで, この 2 つの問題を同時に解決する方式は提案されていなかった. 本研究で提案した構造化チャネルルーティングは, ワームホールルーティングに構造化バッファプールのアルゴリズムを取り入れることにより, この 2 つの問題を同時に解決する. さらに, 本方式を実現するスイッチ LSI を開発し, 実システムでの評価を行ない, 性能を検証した.

トラスネットワークでの最適な全対全通信方式は, リンク層のルーティングに対応するもので, ネットワークの性能を最大限に引き出す全対全通信を実現する. 多くの応用で頻繁に使用される通信パターンの一つである全対全通信は, 通信転送量が多く, マトリクスの転置, 二次元 FFT, ADI などで使用されている. これまで, ストアアンドフォワードあるいはワームホールルーティングを使った方式も含め多くの研究がなされているが, メッシュの端と端が接続されているトラスネットワークでの最適な全対全通信方式は提案されていなかった. 本研究では, ワームホールルーティングを使ったトラスネットワーク上での最適な全対全通信を提案した. トラスネットワークでは, メッシュネットワークの 2 倍の性能を実現することができる. 本方式は, 1 次元トラスから多次元トラスのネットワークに適用することができ, また, 正方形だけでなく, 長方形のネットワークにも適用することができる. さらに, 接続チャネルが単方向あるいは両方向に対しても最適な方式を提案している. 提案した方式では, ノードと相互結合網とのインタフェースのバンド幅は相互結合網の 1 チャネルのバンド幅と同じとしており, トラスネットワークを持つ並列計算機に広く適用することができる.

トランスポート層のメッセージ処理の高速化では, まず, 分散メモリ型並列計算機の性能を向上させるために必要な演算性能と通信性能のバランスについて定量的な評価を行った. その評価のために, メッセージパッシングアーキテクチャを対象に应用問題への通信や演算性能の影響を容易に調査可能とするメッセージレベルシミュレータを開発した. 評価の結果, CPU 性能を 32 倍に上げても, メッセージ処理性能を向上させなければ 10 倍程度の性能向上しか望めないこと, ネットワーク性能の影響に比べると, メッセージ処理性能が応用性能に大きな影響を与えることを示した.

その上で, トランスポート層のメッセージ処理について, 大きく分けて二つの方式を検討した. まず, 第一は, メッセージ長に比例したメッセージ処理オーバーヘッドの削減である. メッセージ処理オーバーヘッドは通信機能の設定など一定の時間を要するものとメッセージ長に比例する時間を要するものがあり, 前者のオーバーヘッドに対してはハードウェア化などいくつかの方法が提案されているが, 後者のオーバーヘッド削減の

効果については研究されていなかった。本研究では、メッセージ長に比例したメッセージ処理オーバーヘッドを削減するため 3 種類の方法、送信と演算のオーバーラップ、メッセージをユーザ領域に直接受信する直接メッセージ受信、受信と演算とのオーバーラップを適用した場合の効果について定量的に評価を行った。その結果、送信と演算のオーバーラップは通信オーバーヘッドを削減するが、その効果はあまり大きくなかった。直接メッセージ受信の効果は、応用問題の性質に依存するが 22%性能が向上するものもあった。受信と演算とのオーバーラップでは、大幅に性能向上し、約 2 倍もの性能向上となるものもあった。これらの検討結果は、今後の相互結合網インタフェースの設計に重要な指針を与えるものと考えている。

第二に、並列化コンパイラが必要とする通信機能、つまり、並列化コンパイラが生成したコードを効率良く実行する通信機能を検討し、それをハードウェア化した場合の効果についてシミュレータを用いて評価した。従来、並列化コンパイラに必要な機能という観点からの検討がなされておらず、並列化コンパイラが生成したコードを効率良く実行することができなかった。本研究では、通信機能を検討した結果、ダイレクトリモートアクセスとその通信完了を通知するフラグ更新 (PUT/GET オペレーション)、バリア同期、グローバル演算が必要であると結論した。そして、PUT/GET をハードウェア化した場合の性能を、実際の並列化コンパイラが生成したコードなどの応用に対してシミュレーションで定量的に評価し、その有効性を示した。本研究により、並列化コンパイラが生成したコードの効率的な実行を実現し、分散メモリ型並列計算機の課題の一つであったプログラミングについて、並列化コンパイラの適用を促進できると考える。また、本研究で評価した PUT/GET をサポートするハードウェアは、PUT/GET によりメッセージバッファリングのオーバーヘッドを削減し、通信と演算をオーバーラップさせることでメッセージパッシングに比べて実行性能を向上させることができる。