

MPI の集団通信機能の改善とゲームプログラムの並列化

晴山光智[†] 山口広行[‡] 苦米地宣裕[†]

[†] 八戸工業大学 大学院 [‡] 八戸工業大学 工学部

1. はじめに

思考ゲームプログラムにおいて、最善手を決めるまでの時間が、先読み深さ（何手先まで読むか）のべき乗で増大するという問題がある。ゲームプログラムでは、リアルタイム応答が要求されるので、並列化による処理速度の向上が求められる^[1]。本研究では、MPI（Message-Passing Interface：分散メモリ型並列計算環境におけるメッセージ交換に関する仕様）^[2]を用いてゲームプログラムの並列化を検討する。

MPI の集団通信では、信頼性確保のため「ユニキャスト通信」（1対1通信）が利用されている。また、プロセス間で木構造を形成することで通信時間の増大を抑制している。しかし、プロセッサ数が多くなると、たとえ木構造を取ったとしても MPI の集団通信時間が増大し、これが並列化の効果を減殺する可能性がある。そのため、集団通信の機能改善が求められている。

本研究は、ゲームプログラムの並列化による性能の向上、及び MPI 集団通信の機能改善を目的とする。具体的には、開発済みの立体四目並べゲームプログラム^[3]を対象に、MPI を用いて並列化することで性能向上を図った。また、通信機能の向上に「ブロードキャスト通信」を用いた通信関数を開発し、同等機能の MPI 関数との性能評価を行った^[4]。本稿では、立体四目並べゲームプログラムの並列化の検討とその効果を検証した結果、及び MPI 集団通信の機能改善結果について報告する。

2. 計算環境の構築

マイクロソフト社（以下 MS）の WindowsCCS 2003 を用いて、4 ノード（8CPU）で構成される PC クラスタ環境を構築した。本環境の主な諸元は以下の通りである。

- ・計算ノード：HP Proliant ML110×4
(CPU：Pentium Dual Core E2160@1.8GHz、メモリ：1GB)
- ・インターコネクト：Ethernet
- ・スイッチングハブ：Cisco Catalyst 2950-24
- ・OS：MS Windows Server 2003 R2 (x64 版)

Improvement of the MPI Group Communication Performance and Parallelization of the Game Program

Mitsutomo Hareyama[†] Hiroyuki Yamaguchi[‡] Nobuhiro Tomabechi[†]

[†] Hachinohe Institute of Technology, graduate school.

[‡] Hachinohe Institute of Technology, department of engineering.

- ・MPI：MS Compute Cluster Pack
- ・開発環境：MS Visual Studio 2005

3. 立体四目並べゲームプログラムの並列化

3.1 ホットスポットの特定

並列化は一般に、プログラム全体を並列化するのではなく、ホットスポット（処理時間が最もかかる部分）を特定した上で、並列化を行うのが効果的である。そこで、既存プログラムの処理手順を解析し、先読み（何手先まで読むか）を変えながら各処理の計算時間を測定した。

6 手を先読みし、1 回の対局中の各計算時間を合計した測定結果を、表 1 に示す。この表から、計算時間が全体の 99% 以上の割合である『最善手探索』処理をホットスポットと特定し、並列化する対象とした。

表 1. 各処理の計算時間

処理内容	時間 (秒)	比率
4 連続検査	0.00026	0.0013%
3 追勝ち検査	0.00122	0.0060%
最善手探索	20.33348	99.8651%
着手	0.00013	0.0006%
その他	0.02586	0.1270%
全体	20.36095	100%

3.2 並列化方針の検討

並列化する方針には、各 CPU の処理時間を均等にすること、通信が頻繁に発生しないことの 2 つがある。これらの方針に基づいて、並列化の方法を 2 つ考案し、その比較・検討を行った。

[案 1] 各手数を別の CPU へ割り当てる、すなわち全手数を並列化する方法である。この方法の場合、各 CPU の処理時間を均等にすることが期待できるが、思考ゲームプログラムでは前の局面情報が必要となるため、CPU 間で頻繁に局面情報を交換（通信）することになる。また、交換した局面情報を各 CPU が保持するのに、大量のメモリも消費することになる。そこで、計算リソースの消費を防ぐために、次の案を考案した。

[案 2] 図 1 に示すように『1 手先』のみを並列化する方法である。この方法では、局面情報の交換や保持が不要となるので、計算リソースの消費を防ぐことができる。

そこで本研究では、[案 2] : 『1 手先』のみを並列化する方法を用いることにした。

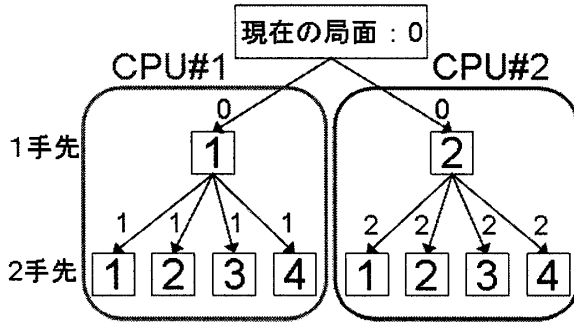


図 1. 並列化の方法

3.3 並列化の実装と効果の検証

並列化の実装のために、プログラムのループ処理を CPU 毎に分割する方法として、ブロック分割を採用した。ブロック分割とは、CPU 数が 4 個の場合、各プロセスに 1/4 ずつの連続した領域を割り当て、それぞれで処理を行う方法である。

次に、最善手を求めるために、MPI 集団通信関数の一つである MPI_Allreduce() を利用した。この関数により、各プロセスで求めた最善手を比較して、全プロセスへ決定した最善手の情報を送るようにした。

以上の各方法を実装し、測定した結果を図 2 に示す。それぞれの値は、実行時間比で表している。この図から、CPU 数が増えると計算時間が短縮していることが分かる。

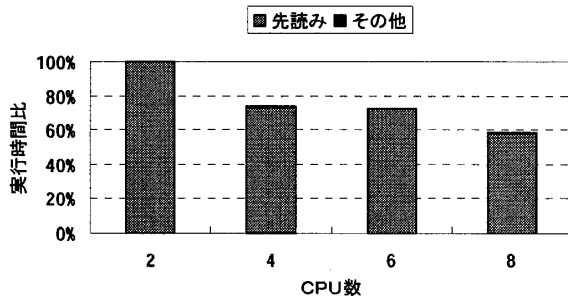


図 2. 並列化の効果

4. MPI の集団通信機能の改善

機能改善に向け、MPI 集団通信関数の一つである MPI_Bcast() を機能改善の比較対象とした。この関数は、あるプロセスが送信元となり、他の全プロセスへデータを一斉送信するための関数である。

本研究では、この MPI_Bcast() と同等の機能を有する通信関数を開発した。具体的には、①ソケットの作成とバインド処理を行う初期化関数、②ブロードキャスト通信によるデータの送受信と確認処理を行う通信関数、③ソケットのクローズ処理を行う終了用関数を開発した。また各関数の引数は、対応する MPI 関数とほぼ同様になるように設

計した。

性能評価は、通信メッセージ長を $1 \sim 2^{26}$ バイトと変化させ、100 回測定した通信時間の平均値を求める方法で行った。 2^{15} バイト時における通信時間の測定結果を図 3 に示す。この図から、MPI_Bcast() の場合 (◆印) は CPU 数 (ノード数) が増えるにつれて、通信時間が増大するのに対し、開発した通信関数の場合 (■印) は CPU 数 (ノード数) によらず通信時間が一定であることが分かる。すなわち、ブロードキャスト通信を用いることで機能改善が図れることを確認した。

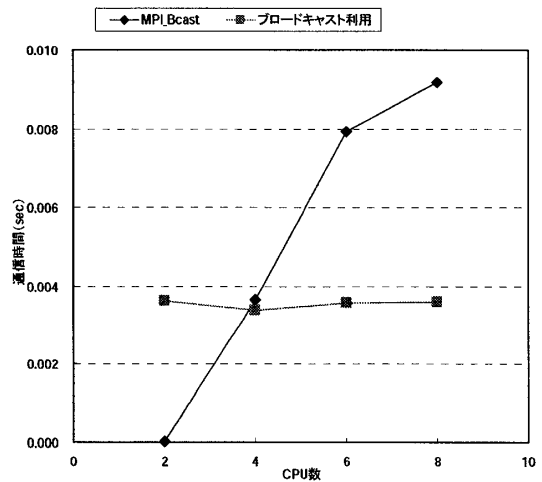


図 3. MPI_Bcast() の通信機能改善結果

5. まとめ

本研究では、計算環境として PC クラスタシステムを構築し、その環境下で MPI を用いたゲームプログラムの並列化、及び MPI 集団通信を行う場合の機能改善を行いその効果を検証した。

MPI を適用して立体四目並べゲームプログラムの並列化を行った結果、CPU 数が増えると計算時間が短縮していくことを確認した。また、MPI の集団通信機能の改善では、ブロードキャスト通信を用いた通信関数を開発し、性能評価を行った結果、通信機能が向上するという結果を得た。

課題点としては、立体四目並べゲームプログラムの並列化効率の向上や、開発した集団通信関数の信頼性の向上、開発した集団通信関数を立体四目並べゲームプログラムへ適用等が考えられる。

参考文献

- [1] M. Newborn, Kasparov versus Deep Blue, Springer, New York, 1997.
- [2] P. パチエコ「MPI 並列プログラミング」、培風館、1997.
- [3] 芥米地宣裕、他「立体四目並べプレイ知能ロボットシステムの構成法」、八戸工業大学紀要、Vol.25, No.25, pp. 119-126, 2006-3.
- [4] 晴山光智、他「MPI 集団通信の性能改善に向けた試み」、平成 20 年度情報処理学会東北支部第 2 回研究会、2008-12.