

OpenMP を用いた分散遺伝的アルゴリズムの移住の 並列化に関する一考察

A study on parallelization of migrations in a distributed genetic algorithm using OpenMP

06510 久保田祐介
指導教員 内田 健

1. はじめに

分散遺伝的アルゴリズム (DGA) をマルチコア型の計算機で実行する場合、コア間通信を必要とする移住が問題となる。過去にコア間通信を最小にするリングトポロジを使った研究が行われたが、島数や母集団規模に関する検討が不十分であった[1]。

そこで、本研究では整列化した移住トポロジを用いて島数や母集団規模を変更した場合の並列化効果を数値実験により明らかにする。

2. 分散遺伝的アルゴリズムと移住

DGA には GA の遺伝的操作に移住という処理が追加される。移住処理は、個体を選別する「移住元の処理」、個体を受け入れる「移住先の処理」に加え、島間の個体移動のための「移動処理」から構成される。

DGA を並列実装する場合、この移住処理は計算ノード間の通信として並列処理のボトルネックとなる。マルチコア型の計算機の場合、コア内の島への移住とコア外の島への移住にキャッシュの同期などによる速度差があると考え、コアをまたぐ移住を最小とるように移住トポロジを整列し、並列化効果の向上を目的とする。

3. 数値実験

本実験では、OpenMP により母集団を島単位で複数のスレッドに分散し、DGA の移住処理を並列化する。以下の使用機材上で表 1 の設定値を用いて数値実験を行い、移住元、移住先、移動の各処理の並列化効果を調べる。

使用機材

Quad-Core AMD Opteron 2376 × 2(8core),
8GB RAM, Linux 2.6.34 (Open SUSE 11.3)

表 1: 数値実験パラメータ

移住間隔	1
移住率	0.25
島数	2,4,8,16,32,64,128,256
コア数	1,2,4,8
母集団サイズ	512, 1024, 2048, 4096, 8192
移住トポロジ	コア間通信を最小, 最大にしたリングトポロジ

図 1 に移動処理の並列化効果を示す。島数が 32 以上の場合、並列化の効果が表れている。しかし、移住トポロジの整列化による違いを確認

できない。一方、移住処理全体では、スレッド数が増加すると処理時間も増加しており、並列化の効果はない(図 2)。

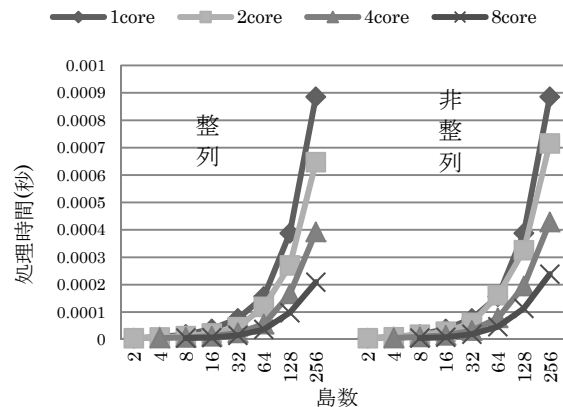


図 1: 個体移動処理の並列化効果

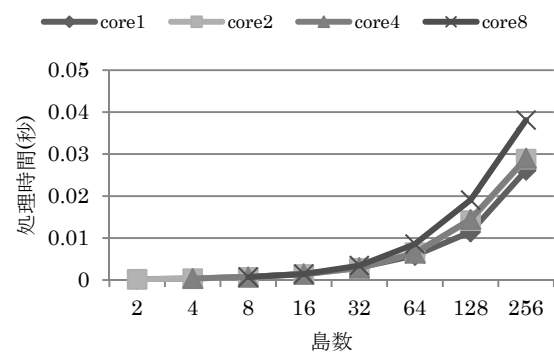


図 2: 移住全体の並列化効果

4. おわりに

本研究では、DGA の移住処理の OpenMP による並列化効果を調べた。数値実験の結果、個体の移動処理に対しては OpenMP による並列化効果を確認した。しかし、移住元・移住先の処理を含む移住処理に対しては OpenMP による並列化が処理時間を増大させた。さらに、個体の移動処理では、コア間通信を最小化した移住トポロジを用いた場合と、それを用いない場合とで、大きく変わらないことがわかった。この原因として、OpenMP ではコアの指定ができないため、移住の整列化が行われていないことが考えられる。

文献

- [1] 伊藤 純: “OpenMP を用いた分散遺伝的アルゴリズムの並列化に関する検討”, 卒業論文, サレジオ工業高等専門学校, 2009