

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

A study on parallelization of a distributed genetic algorithm using OpenMP

05506 伊藤 純
指導教員 内田 健

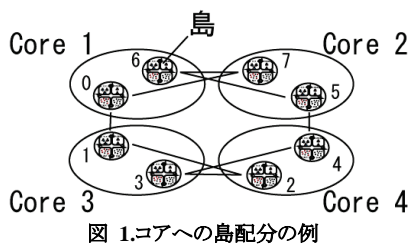
1. はじめに

近年、複数の CPU コアを持つ計算機が数多く登場し、高速な計算処理が可能になっている。分散遺伝的アルゴリズムをマルチコア CPU 上で実装する場合、コア間通信を必要とする移住が問題となる。そこで、本研究では島をコアへ配分する方法を工夫した新たな移住方法を提案し、その効果を数値計算により明らかにする。

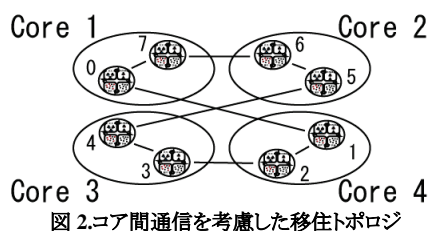
2. 提案法

分散遺伝的アルゴリズム(DGA)は、生物の進化を模倣した最適化手法である遺伝的アルゴリズムの分散処理モデルである。解候補(個体)の集団である母集団を複数の島に分割し、各島で遺伝的アルゴリズムを適用する。この時、島間で個体を交換するための移住という操作を行う。

DGA をマルチコアプロセッサ上で実装する場合、一般に複数の島を各コアに配分する(図 1)。この結果、移住がコア内通信及びコア間通信として実現される。コア間通信の場合キャッシュの動機が必要になるなどコア内通信に比べオーバーヘッドが大きいと考えられている。



そこで、ランダムリングトポロジを持つ移住において、移住を整列化することでコア間通信を最小化する移住方式を検討している。本研究では、移住を整列化するために各コアへの島の配分を考慮した場合(図 2)の並列化効果を数値実験より確認する。



3. 数値実験

OpenMP[1]を用いて DGA の移住処理を実装し、各コアに配分する島数の変化に対する提案手法の効果を確認する。各コアに配分する島数を変化させたときの最終世代数までの処理時間を 5 回測定し、その平均値を図 3 と図 4 に示す。なお、数値実験に使用した計算環境を以下に示す。

使用機器

PowerEdge T605

Quad-Core AMD Opteron™ × 2 基

Linux 2.6.27 (SUSE Linux)

OpenMP Version 2.5

表 1. 実験パラメタ設定値表

島数	4 core	4, 8, 16, 32
	8 core	8, 16, 32, 64
個体数	512	
最大世代数	25000	
交叉率	1.0	
移住間隔	1	
移住率	0.25	

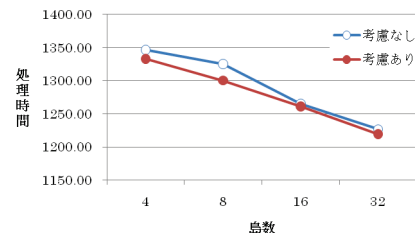


図 3. 4core での処理時間

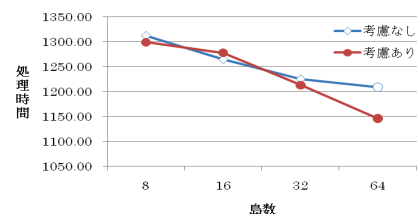


図 4. 8core での処理時間

4. おわりに

コア間通信を低減した移住トポロジを持つ DGA では、島数が多い場合移住トポロジを考慮しない場合に比べ並列化の効果が期待できる。しかし、島数が少ない場合においては並列化の効果が期待できないため、別の工夫が必要になると思われる。

参考文献

- [1] 黒田久泰, “C 言語による OpenMP 入門”, 東京大学情報基盤センタープログラミング講習会資料, <http://www.cc.u-tokyo.ac.jp/>