

リング型マルチプロセッサにおける
Genetic Algorithmの並列処理

1. はじめに

このアルゴリズムは、次世代の親を生成する処理は独立性が高く、乱数を用いていることによる不確実性はあるもののマルチプロセッサによる並列処理によって性能向上が期待できる。今回トランスピユータを用いたリンク型マルチプロセッサにより、Genetic Algorithmを並列処理することを試みた。

The diagram illustrates a ring network topology. At the top, a box labeled "IBM PC/AT" is connected to a central node labeled "1800". This central node is part of a ring of eight nodes, each labeled "1800". The nodes are connected in a circular fashion, forming a closed loop.

今回Genetic Algorithmを適用した問題は、Knapsack Problemと呼ばれる全0-1整数計画問題であり、以下ので示される。

目的関数	$\sum V_i x_i$	最大化	V: 価値
拘束条件	$\sum w_i x_i \leq C$		C: Capacity
	$x_i = \{0, 1\}$		w_i : weight

親集合の数を N_p 、世代数を N_g 、用いるプロセッサ数を M とすると、基本的には 1 台のプロセッサが N_p / M の親集合を分割して持ち、全親集合からのコストに応じた確率で親抽出、子孫生成、子供の評価を繰り返し、受け持った親の数と同数の子供が生成されるまで処理を行う。ここで親として選ばれる確率は、（各自の解のコスト／親集合全体のコスト）であるため、各プロセッサは各自の子供生成が終了すると、自分で生成した子供（次世代の親）の部分集合と子供のコストの部分総和を他のプロセッサへ知らせ、かつ同時に他のプロセッサから子供の部分集合と部分総和を受け取って、親集合全体

4. 处理性能

SUN 3	153.45
VAX 8200	64.21
T414	6.40
T800	1.63

アロセッサ台数 (Number of Aroclor 1248)	相対処理時間 (Relative Processing Time)
1	1.0
2	0.4
4	0.25
8	0.2

の種を変えて処理をした平均である。扱った問題は、親が80人、世代数が20世代の処理をする。T800はVAX8200の約40倍の性能が得られた。

以上トランスペュータを用いたリング構成マルチプロセッサにより、ジェネティックアルゴリズムを台数に応じて高速化できた。プログラム構造としても、台数を増やすことによる変更は殆どなく、使用環境に応じて台数を用意することによって、コスト性能比が高い処理が可能である。

[1]Edit by Lawrence Davis: Pitman Publishing
"Genetic Algorithm and Simulated Annealing"