

4L-5

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

大庭 良忠 青柳 雄大 緑川 博子 飯塚 肇
成蹊大学 工学部

1. はじめに

0-1問題などに対し、ヒューリスティック解法であるGenetic Algorithm^[1]が効果をあげている。このアルゴリズムはランダムに生成した複数の解の集合（第1世代親集合）から個々のコストに応じた確率で2人の親を選び、この2つの親の解からランダムな方法で子孫生成とも言うべき操作を行い、生成された子供のうち、優秀なものを残して次の親集合へ、そうでないものは捨てるという操作を何世代にも渡って繰り返すものである。予め決められた親の数、世代数の処理を行い、その中で最良解を解とする。

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。今回トランジistor用いたリング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。

2. システム構成と問題

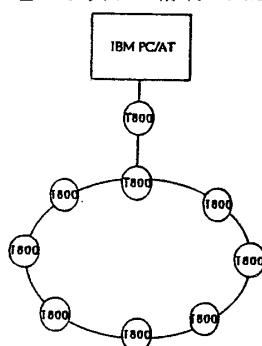


図1 システム構成

本実験で用いたシステムは図1のような構成をとる。用いたのはT800、或はT414である。
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、以下のとおり示される。

$$\begin{array}{ll} \text{目的関数} & \sum V_i x_i \quad \text{最大化} \\ \text{拘束条件} & \sum w_i x_i \leq C \\ x_i = & \{0,1\} \end{array} \quad \begin{array}{l} V: \text{価値} \\ C: \text{Capacity} \\ w_i: \text{weight} \end{array}$$

3. 並列アルゴリズム

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とするとき、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の親抽出、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/N_p）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も子供の部分集合と部分総和を受け取って、親集合全体

のコストから、次世代の親の抽出に必要な確率計算を行う。通信は各プロセッサからのメッセージをリング上に流す事により行う。

4. 処理性能

使用したトランジistor台数と処理速度の関係を図2に示す。測定は、乱数の種類、親の数、世代数などを変えて行い、相対処理時間の平均を示した。子供の生成は表1 他のマシンとの性能比較

SUN 3	153.45
VAX 8200	64.21
T414	6.40
T800	1.63

単位:sec

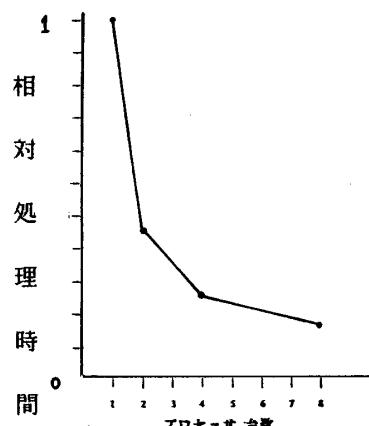


図2 プロセッサ台数と処理時間

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VAX 8200の約40倍の性能が得られた。

5. おわりに

以上トランジistor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。

参考文献

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