

# Parallerization methods of Evolutionary Computation

Mitsunori MIKI\*, and Tomoyuki HIROYASU\*

(Received November 4, 1999)

Recently, several heuristic computational methods have been developed. Especially, many researchers are focused on the evolutionary computation methods that derives its behavior from a metaphor of some of the mechanisms of evolution in nature. As the result, it is found that the evolutionary computation can be derived good solutions in real world problems. However, the evolutionary computation needs a lot of iterations of calculation and it takes much time to derive the solutions. One of the solutions of this problem is to perform evolutionary computation in parallel. This paper explains the parallerization methods of evolutionary computation. There are also several methods in evolutionary computation and the genetic algorithms and the simulated annealing are focused.

**Key words** : Evolutionary Computation, Parallerization, Distributed Processing, Genetic Algorithms, Simulated Annealing

キーワード : 進化的計算, 並列処理, 分散処理, 遺伝的アルゴリズム, シミュレーティッドアニーリング

## 進化的計算手法の並列処理

三木光範・廣安知之

### 1. はじめに

近年, 自然界にアナログを持つ計算手法が注目を集めている. これらの手法は進化的手法と呼ばれ, コンピュータ上でなんらかの生物の進化シミュレーションを行い, 解析や解探索を行うことを特徴としている. これらの中で代表的な手法が遺伝的アルゴリズム (Genetic Algorithm: GA) や遺伝的プログラミング (Genetic Programming: GP) であろう. これらの進化的手法はこれまでの手法では求めることができなかったような複雑な問題や多様な問題に対しても対応できることが明かとなってきている.

また, タブサーチ (Tabu Search) などといったヒューリスティックな手法やインテリジェント手法 (Intelligent Method), 創発的計算手法 (Emergent Computation

Method) なども注目されるようになってきた. これらのいくつかの手法は自然界にアナログを持つことを特色としているため進化的計算手法とは関連が深い. アントコロニシステム (Ant Colony System) などはその例の一つである.

一方, 確率を使用した確率的戦略手法 (Probabilistic Strategy) も近年, 実用的に使用されている計算手法である. シミュレーティッドアニーリング (Simulated Annealing: SA) はその中でも代表的な手法である. SA などは狭義では進化計算的手法とは言えないが, 多くの進化的計算手法が確率を使用しているため, その類似性は高い. また, 単点探索ではなく, 多点探索を行った場合には, 進化的手法を取り入れることが有効である.

\* Department of Knowledge Engineering and Computer Sciences, Doshisha University, Kyoto  
Telephone: +81-774-65-6434, Fax: +82-774-65-6796, E-mail: mmiki@mail.doshisha.ac.jp, tomo@is.doshisha.ac.jp

これら近年注目されている手法は、複雑な問題や多様な問題に対して有効であるが、一方で、繰り返し計算を多く必要とするといった問題が存在する。これらの問題の解決の一つは、これらの手法を並列に処理することである。そもそも進化的計算手法は多点探索を行う場合が多いため、並列計算には適していると言われており、いくつかの並列モデルが考えられる。

そこで、本研究では、進化的計算手法うち最も広く使用されている遺伝的アルゴリズム (GA) とシミュレーテッドアニーリング (SA) に着目し、その並列モデルについて検討する。進化的計算手法を並列に処理する際に最も重要なことは、並列処理により処理速度が向上するだけでなく、モデルの変更により、必要計算数や求まる解の精度が変化することである。

## 2. 進化的計算手法

これまでにさまざまな進化的計算手法が提案されている。その中でも有力な手法が遺伝的アルゴリズム (Genetic Algorithm: GA)、遺伝的プログラミング (Genetic Programming: GP)、進化的戦略 (Evolutionary Strategies: GS)、進化的プログラミング (Evolutionary Programming: EP)、クラシファイアシステム (Classifier System: CS) などであろう。これらの手法の共通的な特徴は、ある構造を持った個体が多数で解析や探索を行うことにある。さらに、解析や探索の過程では、これら複数の個体は、交叉、突然変異といった操作によって構造を変化させる。各個体はそれぞれ適合度と呼ばれる値を持ち、この値によって確率的に選択されたりされなかったりする。これを繰り返し計算することによって解析や探索が行われるのである。

遺伝的アルゴリズムの典型的なアルゴリズムを Fig. 1 に示す。

先にも述べたように、進化的計算手法では多点により繰り返し計算を行うために計算コストが莫大となる。そのため、続く 2 章ではこれらの進化的手法のうち遺伝的アルゴリズムとシミュレーテッドアニーリングに着目し、その並列化について説明する。

## 3. 並列遺伝的アルゴリズムのモデル

### 3.1 遺伝的アルゴリズムの並列化

GA は自然界の進化と淘汰の仕組みを模倣したアルゴリズムである。GA は評価・選択・交叉・突然変異と呼ばれる遺伝的操作を繰り返すことで最適解を見つけ出す最適化アルゴリズムである。本論文におけ

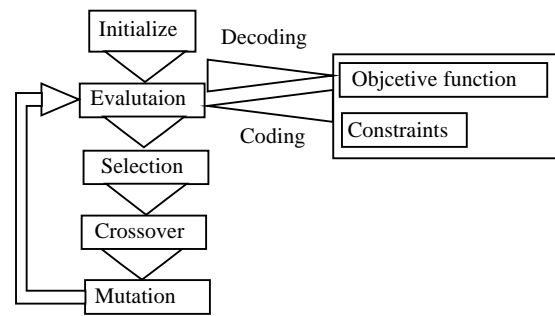


Fig. 1 Genetic Algorithm.

る GA の遺伝的操作は、交叉には一点交叉、選択にはルーレット選択を用いている。実用的な GA の並列モデルを考えると、通信によるオーバーヘッドやモデルの単純さの点から、粗粒度モデルと細粒度モデルを考えることができる。次節では、これらのモデルについて詳細を述べる。

### 3.2 粗粒度並列モデル

粗粒度モデルは一般に島モデルとも呼ばれる。このモデルでは、母集団を複数のサブ母集団に分割し、各母集団ごとに遺伝的操作を実行する。そして数世代の反復計算ごとに、あるサブ母集団のいくつかの個体を他のサブ母集団に移動させる。この遺伝的操作は移住と呼ばれる。本論文での移住は、全てのサブ母集団において同一の世代で適用し、ランダムリング型の移住を行う。これは移住世代ごとにランダムにリングを形成し、そのリングに基づいて移住先が決定される。つまり母集団全体でみるとある世代における移住は、Fig. 2 に示すようにリング型の個体交換モデルとなる。移住個体数は初期パラメータで与えるが、移住個体や移住先などは、移住世代ごとにランダムに選ばれることになる。粗粒度モデル GA の処理の流れを Fig. 3 に示す。これからもわかるようにこのモデルは通信量が少なく並列処理に適している。また、モデルの特性から単一母集団モデルの際と比較して必要な計算量が減少することが知られている。

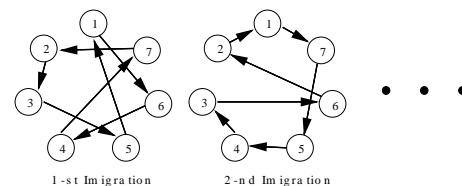


Fig. 2 Migration method.

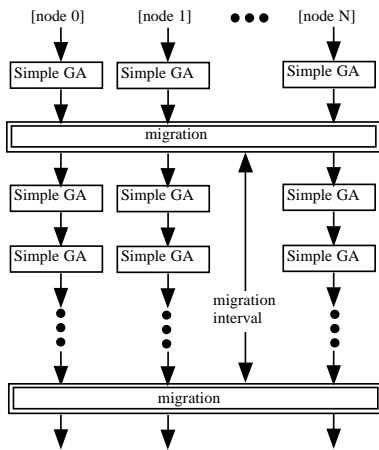


Fig. 3 Flow of coarse-grained model.

### 3.3 細粒度並列モデル

GA においては評価の操作に要する時間が、全体の計算時間の大部分を占めることが多く、対象とする問題が複雑になるほどその傾向が強くなる。そこで非常に単純な並列モデルとして評価計算の部分を分散し、並列に計算を行う細粒度並列モデルが考えられる。これはマスター・スレーブ型をとり、評価を除く全ての遺伝的操作はマスターとなる1つのプロセスにおいて行う。評価の操作は、マスターから複数のスレーブに評価すべき個体のデータを送信し、スレーブにおいて実際の計算を行い、結果をマスターに返すという手順となる。ロードバランスを考慮して、マスターからスレーブへの個体データの送信は1個体ずつとするため、実際には Fig. 4 に示すようなモデルとなる。

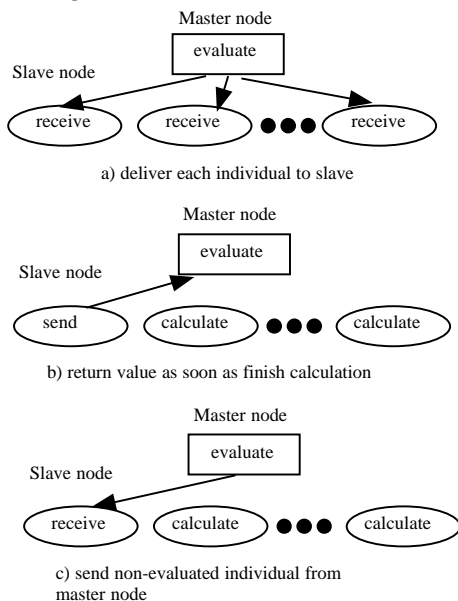


Fig. 4 Flow of micro-grained model.

このモデルの長所は逐次アルゴリズムから並列アルゴリズムに変更することが極めて簡単である点である。しかしながら、基本的なモデルは逐次処理の際と同一のため必要な総計算量は変化せず、かつ、通信量は比較的多く、1CPU がマスターとして必要なため、小規模な並列計算機では不利であるという特徴を持つ。

## 4. シミュレーテッドアニーリングの並列化

### 4.1 シミュレーテッドアニーリング

シミュレーテッドアニーリング (SA) は、過熱炉内の個体を徐々に冷却し低エネルギーの状態を得るといった焼きなましの過程をシミュレートした手法である。近傍探索をランダムに行いながら、解が改悪する場合でも次の状態に遷移する可能性を持つことで局所解に陥ることを防いでいる。SA は最適化問題を解く有効な手段であるが、マルコフ連鎖をたどる処理であるため、本来強い逐次性があり並列化は容易ではない。しかし計算の効率化と解品質の向上を図るために、SA を並列化する研究が盛んに行われている。次節では提案されているいくつかの並列 SA モデルについて詳細を述べる。

### 4.2 マルコフ連鎖モデル

マルコフ連鎖とは、現時点の状態と出力が定まれば次の状態は一意に定義されるという状態遷移のことである。SA ではこのマルコフ連鎖を次々にたどることで、最適解に到達することが数学的に保証されている。そのため SA を並列化する際も、マルコフ連鎖を分断しないことで理論上は最適解が求められる。マルコフ連鎖モデルには以下のようなものがある。

温度並列 SA (TPSA) は、複数のプロセッサに異なる温度を与え、各プロセッサでは一定温度でアニーリングを行い、一定の間隔で隣接する温度プロセッサ間で解を交換するという手法である。Fig. 6 は逐次 SA と比較した TPSA の概念図である。上側に示した逐次 SA ではクーリングスケジュールを実験的に決定しなければならないが、TPSA では各プロセッサが一定の温度を保つためクーリングスケジュールが不要となる。また時間的に一様であるため任意の時点で探索を終了することができ、解の劣化を防ぐことができる。

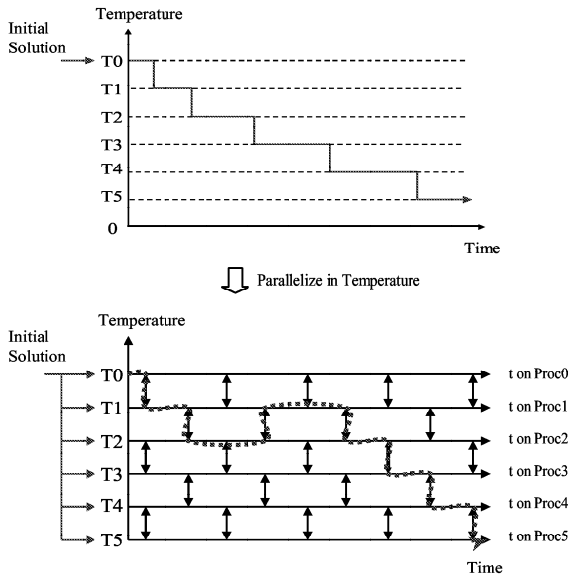


Fig. 5 SA and TPSA.

並列 SA をシストリックにする方法もマルコフ連鎖が保たれたモデルである。P 個のプロセッサを用いて実行するとき、長さ N の連鎖を長さ  $L = N/P$  のサブ連鎖に分割する。図 4 はシストリックのアルゴリズムの概念図であるが、まずプロセッサ 0 で長さ L のアニーリングをした後、一定温度でアニーリングを続けるプロセッサ 1 と、クーリングを行ってアニーリングを続けるプロセッサ 0 に探索を二分する。PICK 部では競合した 2 組の解と温度を選択する。この作業を繰り返し、P 個の異なったプロセッサで実行する。

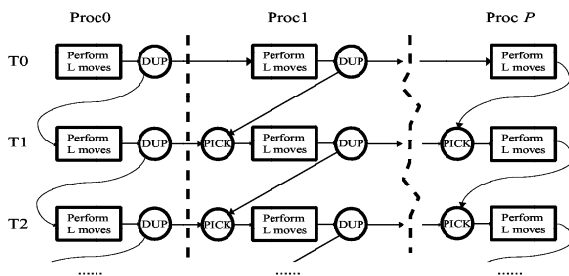


Fig. 6 Systoric SA.

### 4.3 マルコフ連鎖分断モデル

マルコフ連鎖をたどり無限に冷却を行うことで、理論上は探索が最適解に到達することを前節で述べた。しかし限られた時間の中では理論を満たす探索は不可能に近い。そこで我々は、複数のプロセッサで SA を実行し、一定間隔ごとに同期をとって解を他のプロセッサに伝達するといったマルコフ連鎖を分断した並列 SA の研究を行っている。伝達の方法としては GA オペレータである選択と交叉が挙げられる。

解の伝達時に選択オペレータを適用したモデルは、すべてのプロセッサの中で最も良い解を選択し、他のすべてのプロセッサはこの同一解から探索を続けるという手法である。一定間隔ごとにすべてのプロセッサに良い解が与えられることでアニーリングの収束が速く、良い解の近傍を重点的に探索することができる。

交叉オペレータを適用したモデルは、解の伝達時にすべてのプロセッサからランダムに親として 2 個体を選択し、設計変数間交叉を行う。もとの親と生成した子との 4 個体間のうち上位 2 個体を選択して、この 2 個体から次の探索を続けるという方法である。ある設計変数の最適値が求まっている場合、交叉によってその最適値を他のプロセッサに伝達することができるため、アニーリングの収束を早めることができる。

### 5. おわりに

これまでにいくつもの進化的計算手法が提案されており、様々な実問題でその有効性が示されている。しかしながら、進化的計算手法の各手法は繰り返し計算を多く必要とするという欠点を有する。

その一つの解決方法がそれらの各手法の並列処理であると考えられる。本論文では遺伝的アルゴリズムとシミュレーテッドアニーリングに着目し、それぞれのいくつかの並列モデルについて説明を行った。遺伝的アルゴリズムにおいては、分散母集団モデルが、シミュレーテッドアニーリングにおいては温度並列モデルや交叉モデルが有効であることを示唆した。

今後、多くのアプリケーションが並列処理により行われるようになるものと思われるが、進化的計算手法の各手法は極めて並列処理との親和性が高く、かつ、様々な問題への適応が可能である有効な手法である。