

遺伝的アルゴリズム

花 田 良 子*

Genetic Algorithm

Yoshiko HANADA

1. はじめに

私たちの身の回りには多くの工学的问题が存在する。産業諸分野においては、回路設計、プラントの制御、スケジューリング、マネージメントやプランニング、建物の構造設計などが挙げられる。これらの多くは限られた資源、制約のもとで最善の解を決定することが望まれ、最適化問題として定式化される。最適化とは、ある制約条件のもと、所与の目的関数の最小値、あるいは最大値を与える設計変数を求めるということをいう。身近な例を挙げると、カーナビにおける目的地ルート検索でも最適化が行われている。ユーザが目的地を入力すると最小走行距離のルートや最小料金ルートなどが表示される。ここでは、ルートが設計変数、走行距離や料金が目的関数である。一方で、通過地点の指定などユーザが与える制約や工事中で利用できない経路などがあった場合には、それが制約条件になる。

目的関数および設計変数を設定しモデルが決定すれば、次に最適解を得るために解法が必要となる。最適化理論に基づく種々の最適化手法は、システムの設計、計画、制御、意思決定論などの工学諸分野における広範囲な問題を解決するのに必要な基本的なアプローチの1つである。最適化手法には、従来の数理計画法に加え、近似解を効率的に求めるヒューリスティックな手法があり、これらの多くは膨大な反復計算を必要とするためコンピュータ上で実現されている。近年、コンピュータ、ネットワークの諸技術の目覚しい進歩に伴い、私たちが「十分速い」あるいは「許容できる／

現実的」と認識する時間内に計算できる量が飛躍的に大きくなっている。このような技術革新によって、領域におけるエキスパートに頼るしかなかった大規模複雑な設計や制御などの分野においても、それらを最適化問題として定式化することで、コンピュータシミュレーションによる最適設計、最適制御が実現可能となってきた。

本稿では、数ある近似解法の中で最も期待されている手法の1つである遺伝的アルゴリズム (Genetic Algorithm ; GA) [1] を紹介する。GA は目的関数の勾配に関する情報などを用いず、目的関数値のみを用いる直接探索法であるため、複雑な非線形関数や任意の構造体を最適化することができ、非常に適用範囲が広く、近年、実問題への適用が多く見られる手法である。

2. 最適化と解法

2.1 最適化問題の定義

最適化問題は一般に次のように記述される。 F を実行可能領域、目的関数 $f(x)$ は実数値 R あるいは整数值 Z をとる関数 $f: F \rightarrow R$ (あるいは Z) である。

$$\text{minimize}(\text{maximize}) f(x), \text{subject to } x \in F \quad (1)$$

F が連続的な構造を持つ場合、式(1)は連続最適化問題、離散的な構造を持つ場合は離散最適化問題あるいは組合せ最適化問題と呼ばれる。ここで連続的であるとは、実数の連続性を満たすことを指す。一方、離散的についてはどのような構造を離散的とみなすかについては様々な意見があるが、一般的に、 n 次元0-1ベクトル、 n 次元整数ベクトル、 n 要素の順列の集合な

原稿受付 平成20年9月17日

*システム理工学部 電気電子情報工学科 助教

ど、“有限個あるいは可算無限個の要素を持つ離散集合”を指すことが多い。

2.2 問題の特徴と最適化手法

連続最適化は目的関数 $f(x)$ 、制約条件 $g(x)$ の特長により大きく 2 つに分けられ、適用する解法も異なる。 $f(x)$ 、 $g(x)$ が x の 1 次式で表される問題は線形計画問題、それ以外は非線形計画問題とよばれる。最適解を求めるにあたり、着目すべき両者の違いは「極値を持つか持たないか」という点である。図 1 に示すように、線形計画問題(a)において、最適解は実行可能領域の端、すなわち制約条件上に必ず存在する。一方、非線形計画問題(b)の場合は、実行可能領域内に極値が存在する可能性があるため、制約条件上に最適解があるとは限らない。

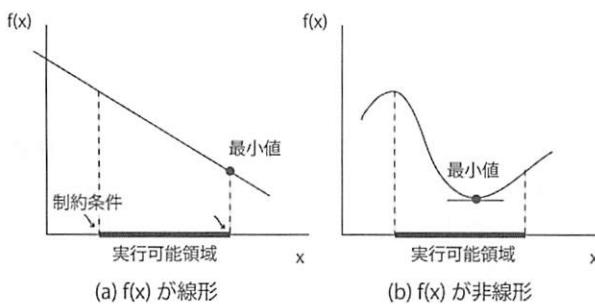


図 1 線形計画・非線形計画の特徴

線形計画問題、非線形計画問題の代表的な解法は以下のとおりである。前者は制約条件上の点に着目した解法、後者は微分を用いた解法である。

- { 線形計画—シンプレックス法 など
- 非線形計画
 - 最急降下法（制約なし）
 - ニュートン法（制約なし）
 - 準ニュートン法（制約なし）
 - 逐次 2 次計画法（制約あり） など

離散最適化問題については、大きく分けて、(1)厳密解法、(2)近似解法あるいは発見的解法、の 2 種のアプローチがある。前者は最終的に得られた解が大域的最適性を有する解法であり、分枝限定法 (branch and bound ; BAB) [2] や動的計画法 (dynamic programming ; DP) [3] などが挙げられる。一方、後者は、最適性の保証はなくとも、現実的な時間で満足解を得ることを目的としたアプローチであり、その基本戦略として欲張り法や局所探索がある。

また、基本的なヒューリスティック手法を組み合わ

せる、あるいは発展させることで、より強力な探索を実現するメタヒューリスティックが注目されている。その代表的なものが、遺伝的アルゴリズム (Genetic Algorithm ; GA)、シミュレーテッドアニーリング (Simulated Annealing ; SA) [4] やタブサーチ (Tabu Search ; TS) [5] などである。これらは目的関数の勾配に関する情報などを用いず、目的関数值のみを用いる直接探索法であり、個々の問題に対して適した操作を設計することで連続最適化、離散最適化の両方を分け隔てなく取り扱うことが可能である。また、目的関数の自由度も高く、少なくとも解同士の順序付けさえできればよいため、多くの制約条件が複雑に絡み合った多重制約条件下での最適化にも対応できる。中でも、GA は他の有力な近似解法の特徴をすべて取り込んだといってもよいほど様々な特徴を持ち合わせており、その高い適用性、潜在能力から、多くの研究、実問題への応用がなされている。本稿では、GA の基本的なモデルと GA の応用について紹介する。

3. 遺伝的アルゴリズムとは

3.1 概 要

GA は自然界における生物の適応進化のメカニズムを工学的にモデル化したアルゴリズムである。自然界では、図 2 に示すように、生活環境に適応できない個体は死滅してゆき、環境に適応した個体は生き残り子孫を増やしていく。このメカニズムをモデル化し、問題に対して最もよく適応する個体、すなわち最適解を計算機上で生成しようというのが GA の概念である。

GA の研究は 1960 年代の後半に Michigan 大学の Holland らによって始められ、その研究成果は 1975 年に “Adaptation in Natural and Artificial Systems” という題目で出版された [6]。また、1989 年に出版さ

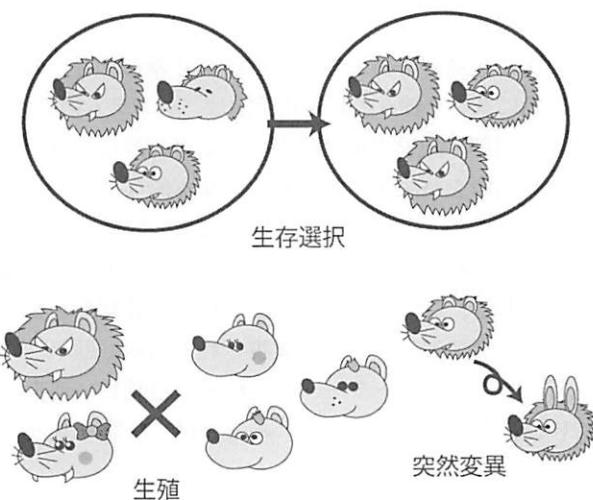


図 2 適応進化のメカニズムのイメージ

れた Goldberg の “Genetic Algorithms in Search, Optimization and Machine Learning” 以降 [1]、日本でも盛んに研究が行われるようになり、多方面に応用されるようになった。

GA による解探索は、個体 (individual) とよばれる解の集合である母集団に対して、選択 (selection)・交叉 (crossover)・突然変異 (mutation) と呼ばれる遺伝的オペレータを繰り返し適用することによって行われる。ある世代を形成している個体間で交叉を行い新しい個体を生成する。また、一定の確率で遺伝子を反転させる、突然変異という操作を加える。こうして生成された個体を評価し、その中で環境への適合度が高い個体ほど高い確率で生き残り、次の世代を形成する。このような世代の更新を繰り返すことで、次第により良い個体が生成され、最適解に近づいていくというのが GA の基本的な概念である。GA は従来の最適化手法と比較して次の 3 つの特徴を持っている：

- ・目的関数値のみを用いる直接探索法である
- ・一点探索ではなく、多点探索である
- ・決定論的規則ではなく、確率的オペレータを用いる探索である

3.2 GA の基本操作

GA における処理の流れを図 3 に示す。以下で、最も基本的なモデルである Simple GA (SGA) を例に GA の個々の処理について説明する。

§ 1 GA の構成要素

ビットを用いる GA の最小構成要素は、0 または

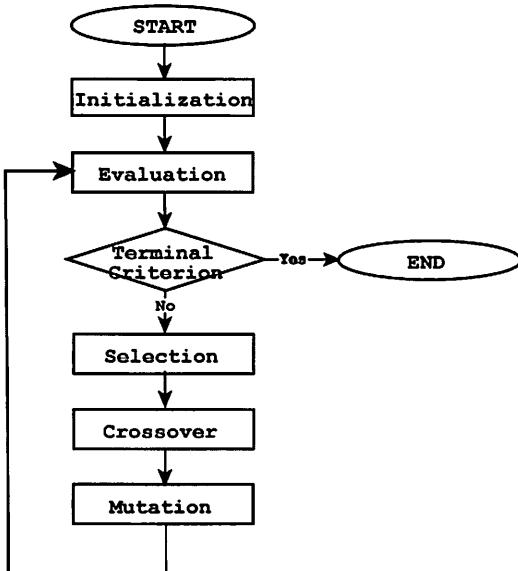


図 3 GA における処理の流れ

1 で表現される遺伝子 (gene) であり、遺伝子の配列を染色体 (chromosome) と呼ぶ。生物においては、染色体の情報より個体が生成されるが、GA では 1 つの染色体が 1 つの個体を表現することが多い。染色体上のある位置を遺伝子座 (locus) と呼び、同じ位置に配置可能な異なる遺伝子を対立遺伝子 (allele) と呼ぶ。GA においては、個体は解候補であり、複数の個体を用いて設定された解空間内における解の探索を行う。図 4 に染色体の例を示す。GA では、その個体の集まりを母集団 (population) と呼ぶ。

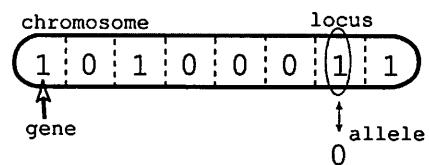


図 4 染色体の例

§ 2 遺伝子型と表現型

生物の染色体の中には、その特徴を表す遺伝子の組み合わせが存在している。GA においても、遺伝子の組み合わせによって、各個体の特徴が決定される。染色体上の各遺伝子座にはそれぞれに特徴があり、目的関数を計算する際にはそれらを設計変数の値に変換する必要がある。設計変数に変換する前のビット列を遺伝子型 (genotype) と呼び、変換後の値を表現型 (phenotype) と呼ぶ。遺伝子型から表現型への変換をデコーディング (decoding) と呼び、表現型から遺伝子型への変換をコーディング (coding) という。2 設計変数の問題におけるコーディング/デコーディングの例を図 5 に示す。コーディング/デコーディングの手法にはさまざまなものがあるが、実数値に符号化する代表的なものとしては、バイナリコーディング (binary coding) とグレイコーディング (gray coding) が挙げられる。

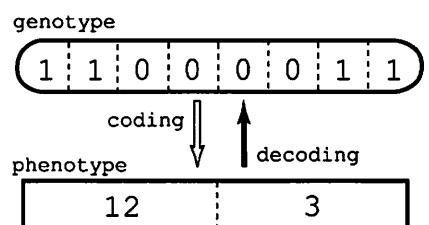


図 5 コーディングの例

§ 3 母集団の初期化

母集団の初期化 (initialization) では、設定された個体数 (population size) に応じた数の個体を生成する。個体は通常、解空間内にランダムに生成される。GA における個体数の設定は重要である。解空間に対し十分な個体数を用いなかった場合には、解空間をくまなく探索することができず、最適解を発見することができない。一方、対象問題に対して過度の個体数を適用した場合には、最適解に収束しにくくなる。そのため、GA を用いた最適化では、対象問題に対する適切な個体数の設定が必要になる。

§ 4 個体の評価

自然界においては、環境に対して適応する度合いの高い個体が生き残り繁殖する。逆に適応力が低い個体は死滅する。GA においてもそれは同様であり、求める解に近い個体ほど、次世代へ残る確率が高くなる。その程度を数値化して表したもののが GA では適合度 (fitness) という値で表す。評価 (evaluation) では、閾数の評価値に基づいて適合度の計算を行う。適合度は一般に非負で、高いほど最適解に近いように設定される。そのため最小値探索などでは、各個体の目的閾数と適合度との間に何らかの変換が必要である。この変換操作をスケーリング (scaling) と呼ぶ。

§ 5 選択

選択 (selection) は評価された個体の適合度によって、次世代の母集団を生成する遺伝的オペレータである。以下に SGA で用いられるルーレット選択の説明を行う。ルーレット選択では、各個体が適合度に比例した形で次世代に生き残る。母集団内の個体数が N の場合において、適合度 f_i の個体が親として選択される確率 P_{select_i} は、式(2)で表現される。

$$P_{select_i} = \frac{f_i}{\sum_{i=1}^N f_i} \quad (2)$$

このほかにも、ランダムに選択した複数の個体の中から、最も適合度の高い個体を選択するトーナメント選択や、適合度の順に個体をソートし、順位に応じた生存確率を設定するランク方式などがある。また、適合度が上位の個体を無条件に次世代に残すエリート保存戦略も選択の一部として考えられる。エリート保存戦略を用いると局所探索能力が向上する反面、局所解に陥る確率も高くなる。

§ 6 交叉

交叉 (crossover) は生物の有性生殖を模倣した操

作であり、親として選ばれた個体間で染色体の一部の組み替えを行う遺伝的オペレータである。交叉は交叉率 (crossover rate) で定められた確率で適用される。ここでは代表的な交叉法である一点交叉の説明を行う。まず、親となる個体を 2 つ選ぶ。次に交叉点と呼ばれる点を染色体上に発生させ、交叉点を境に両者の遺伝子を交換する。図 6 に 1 点交叉の例を示す。

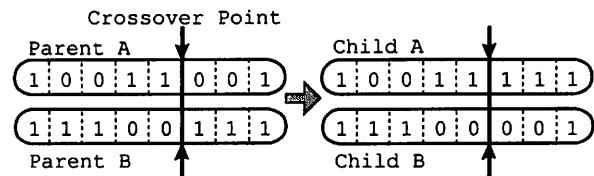


図 6 1 点交叉の例

§ 7 突然変異

選択と交叉のみでは、初期母集団内に存在する遺伝子の組み合わせでしか探索が行われないため、局所解に陥ると脱出が不可能になる。GA における突然変異 (mutation) は、突然変異率 (mutation rate) で定められた一定確率で、染色体上のある遺伝子を他の対立遺伝子に置き換えることを指す。図 7 に突然変異の例を示す。突然変異を行うことにより、母集団内に存在しない遺伝子を発生させ、探索空間全域の探索が可能となる。しかしながら、突然変異率が高すぎると染色体中の部分解が破壊され、ランダムサーチの要素が高くなる。

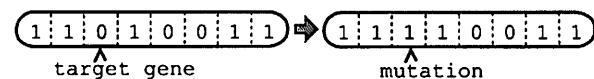


図 7 突然変異の例

§ 8 終了判定

GA の実行は、あらかじめ定めた終了条件を満たした時点で終了され、その時点で最高適合度を示した個体を解として採用する。一般的な終了条件として以下のものがあげられる。

- ・母集団内の最高適合度あるいは平均適合度がある値を超えた場合
- ・最高適合度がある世代経っても更新されない場合
- ・世代交代の数があらかじめ設定された回数を超えた場合

3.3 GA の研究課題

3.2で挙げた SGA は最も基本的な GA の構成である。これまでに非常に多くの手法が考案されており、近年では実問題への応用も多く見られるようになった。

実問題へ GA を適用するにあたって望まれることは、その探索性能、汎用性、実用性の向上である。高性能化を図る鍵は、GA の主探索オペレータである交叉の設計であり、問題固有の構造、性質を十分熟知した上で、交叉の設計を行う必要がある。交叉において、親個体が持つ良好な形質（部分解）が結合され、より優れた子個体が生成される操作を設計できたなら、GA は他の近似解法を凌駕する強力な最適化手法となりうる。

また、並列処理との親和性は GA の高い実用性を保証する大きな特徴の 1 つであり、負荷の高い実問題に GA を適用する際には、並列処理することで高速化が図られてきた。最近では、広域ネットワークの通信速度、計算機性能の向上により超並列 PC クラスタ、グリッドなどの大規模な計算環境が整いつつあり、GA の実アプリケーションへの期待が高まっている。

4. 遺伝的アルゴリズムの応用

4.1 実数値関数最適化へのアプローチ

連続最適化の 1 つである実数値関数の最適化においては、コーディングに実数ベクトルを採用する実数値 GA が用いられることが多くなっている。前節で説明したビットを用いる従来型の GA を関数最適化に適用する場合、遺伝子型と表現型の位相構造が大きく異なることから形質遺伝に優れた交叉を設計することは難しい。また、精度の良い解を得ようとした場合、染色体長が非常に長くなるといった問題がある。実数値 GA では、コード化に実数ベクトルを採用することでこれらの問題を解決している。ここでは、設計変数の依存関係を考慮した单峰性正規分布交叉 (Unimodal Normal Distribution Crossover ; UNDX) [7] を紹介する。

UNDX は、図 8 に示すように、両親と第 3 の親を用いて両親を囲む正規分布に従い子個体を生成する。子個体は次式に基づき生成される。

$$C_1 = \frac{P_1 + P_2}{2} + z_1 e_1 + \sum_{k=2}^n z_k e_k \quad (3)$$

$$C_2 = \frac{P_1 + P_2}{2} - z_1 e_1 - \sum_{k=2}^n z_k e_k \quad (4)$$

ただし、

$$z_1 \sim N(0, \sigma_1^2), \quad z_k \sim N(0, \sigma_2^2) \quad (k = 2, \dots, n)$$

$$\sigma_1 = \alpha d_1, \quad \sigma_2 = \beta d_2 / \sqrt{n}$$

$$e_1 = (P_2 - P_1) / |P_2 - P_1|,$$

$$e_i \perp e_j \quad (i \neq j = 1, \dots, n)$$

ここで、 n は対象問題の次元数、 P_1 と P_2 は両親、 P_3 は第 3 の親、 C_1 と C_2 は子個体、 d_1 は両親間の距離、 d_2 は第 3 の親と両親を結ぶ直線との距離、 e_1 は両親を結ぶ軸方向の単位ベクトル、 e_i ($i = 2, \dots, n$) は e_1 に垂直かつ線形独立な単位ベクトル、 $N(\mu, \sigma^2)$ は平均 μ 、標準偏差 σ の正規分布をあらわす。

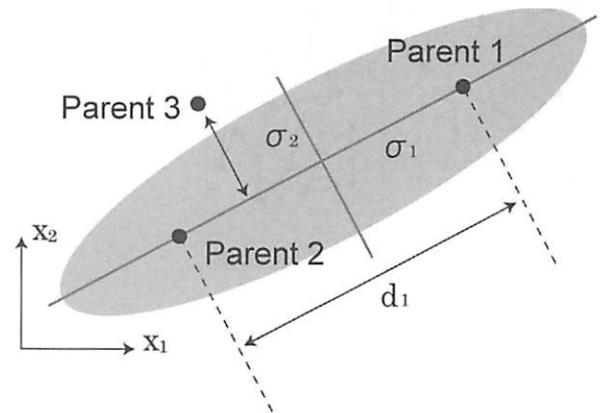


図 8 単峰性正規分布交叉

UNDX では、両親を結ぶ直線付近に子個体を生成することで、設計変数間に依存関係を有する問題に対しても良好な解を得ることができるという特徴を持つ。

4.2 巡回セールスマン問題

巡回セールスマン問題 (Traveling Salesman Problem ; TSP) は都市の集合が与えられたとき、すべての都市を 1 回ずつ訪問して、出発点の都市に戻ってきて来る Hamilton cycle (tour) の中で最短の経路長を有するものを求める問題であり、代表的な NP-困難である組合せ最適化問題の 1 つである。単純な記述の問題であるが、X 線結晶解析のアプリケーションや回路設計、タンパク質の構造解析などの基礎となるモデルの 1 つである。

TSP によらず、問題固有の構造、性質を持つ組合せ最適化問題において、交叉の設計は困難であり、設計に失敗するとランダムサーチ程度の性能しか得られないことが経験的に知られている。そのため、GA の探索性能の向上を目的とした研究では、交叉の設計が大きな課題となっている。TSP は単純な定義と軽い制約を持つ問題であるが、制約を満たしつつ親の良い点を十分に受け継ぐことができる交叉が開発されるまでには、十年以上の時間と何十もの交叉を積み重ねる必要があった。現在、計算コストと精度の面から最も優れた交叉は永田による枝組み立て交叉 (Edge

Assembly Crossover ; EAX) [8] であるといわれている。図9にEAXの概要を示す。図9では、親A、BからABサイクルという部分的なツアーアップを作り、中間的な解を作っている。これは親の良い部分を子個体にうまく遺伝させるためである。

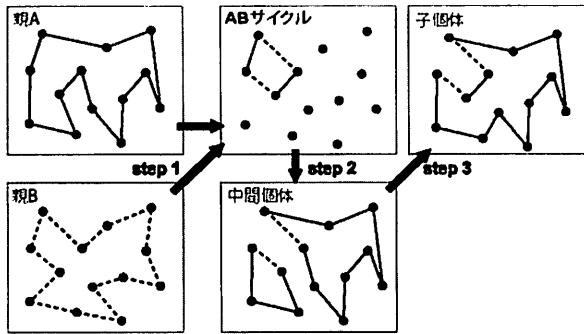


図9 EAX の例

なお、TSPLIB [9] や World TSP [10] のサイトに、数十都市～数百万都市の規模の例題が用意されている。

4.3 GA の並列化

GAは多点探索を用いた反復計算により計算コストが高い。従来から、実時間でより精度の高い探索を実現するため、並列計算機に実装を行うことで解決されている。代表的な並列GAモデルとしてマスター・スレーブモデル、セルラモデル、および島モデル [11] がある。中でも島モデルは並列化効率が高く、解探索性能も優れており、探索アルゴリズムとして様々な拡張が行われている。

島モデルでは、GAの母集団は使用する計算ノード分のサブ母集団(subpopulation)に分割される。サブ母集団は並列計算機上の各プロセッサに分配され、各サブ母集団においてGAが行われる。なお、広義では、母集団をプロセッサ分のサブ母集団に分割するモデルの総称を島モデルと呼ぶが、分散GAと呼ばれる島モデルの1つを意味する場合にも用いられる。

各サブ母集団が孤立した環境で探索を進めるモデルと、移住(migration)と呼ばれる、一定間隔ごとにサブ母集団間で個体の交換を行うモデルがある。移住の概念図を図10に示す。

島モデルでは、移住を行う世代周期を移住間隔(migration interval)、サブ母集団内全体に対する移住個体の割合を移住率(migration rate)といったパラメータが必要となる。上記のパラメータに加え、移住個体の選択法や移住先の選択方法であるトポロジーの設計が探索能力に大きく影響し、論理的なGAの

モデルとの繋がりが非常に大きい。複数の母集団で探索を進めるため、多様性の維持が可能となり、上記のパラメータをうまく設定すれば、1つの母集団で行うGAと比較して高い探索性能を示すことが知られている。

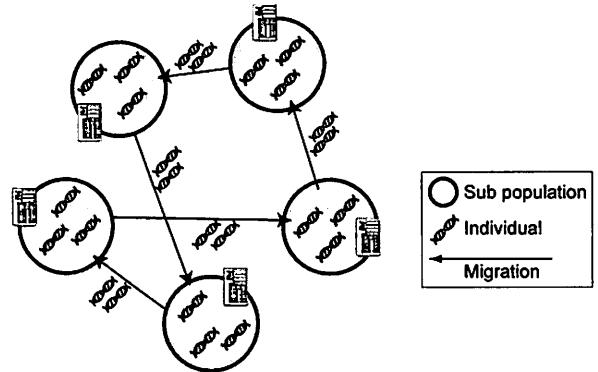


図10 島モデル

5. おわりに

コンピュータシミュレーションによる問題解決のプロセスは数理モデル化と最適化の2つのステップからなる。本稿ではモデル化後の最適化に絞り、その手法の1つであるGAの基礎と応用について述べた。GAが提案され50年近く経つが、高い計算コストから、産業の分野では幅広く用いられるようになったのは最近である。現在のGAの主要な研究課題の1つは実用化を高めることである。実問題においては、最適化自体も困難を極めるが、解空間を決定するモデル化が非常に重要である。実現性を加味した数理モデリングのもとでは、単に最適解を求めるだけでなく、外部環境の変化など何らかの振動に対する解のロバスト性や、多数の目的関数において得られた候補解集合からの選好方法なども大きな課題となっている。

参考文献

- [1] D. E. Goldberg: *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Publishing Company, 1989.
- [2] A. H. Land and A. G. Doig: *An Automatic Method for Solving Discrete Programming Problems*. *Econometrica*, Vol. 28, pp. 497-520, 1960.
- [3] Dimitri P. Bertsekas: *Dynamic Programming and optimal Control*. 2nd ed. Athena Scientific, Vols. 1 and 2, 2000.

- [4] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi: Optimization by Simulated Annealing. *Science*, Vol. 220, No. 4598, pp. 671-680, 1983.
- [5] F. Glover and M. Laguna: *Tabu Search*. Kluwer Academic Publishers, 1997.
- [6] J. H. Holland: *Adaptation In natural and Artificial Systems*, University of Michigan Press, 1975.
- [7] 小野功、佐藤浩、小林重信。単峰性正規分布交叉 UNDX を用いた実数値 GA による関数最適化。人工知能学会誌、Vol. 14, No. 6, pp. 1146-1155, 1999.
- [8] 永田裕一。局所的な交叉 EAX を用いた GA の高速化と TSP への適用、人工知能学会論文誌、Vol. 22, No. 5, pp. 542-552, 2007.
- [9] TSPLIB: <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>
- [10] World TSP:<http://www.tsp.gatech.edu/world/>
- [11] R. Tanese: Distributed genetic algorithms, Proc. of 3rd International Conference on Genetic Algorithms, pp. 434-439, 1989.