

多数制約付き最適化問題への 遺伝的アルゴリズムの適用法に関する検討

A Study on Application Method of Genetic Algorithm
for Many Constrained Optimization Problem.

大江良^{*1} 吉川大弘^{*1}
Ooe Ryo Yoshikawa Tomohiro

^{*1}名古屋大学工学研究科
Graduate School of Engineering Nagoya University

Recently, Genetic Algorithm (GA) is actively applied to engineering problems. In the most of actual engineering problems, they have many constraints. Penalty methods have been applied to these constrained problems so far. However, it is difficult to find the solutions which satisfy all constraints because penalty methods try to satisfy all constraints at the same time and it is often easier to optimize objective functions than satisfying difficult constraints especially in later generations. In this paper, the stepwise satisfaction method of constraints is proposed. The features of this method are to define the difficulty of each constraint by the initial population and to give the priority to be satisfied on more difficult constraints. In this method, the individual which satisfies more difficult constraints is given priority for the parents selection. Thus constraints are satisfied in order of their difficulties. The proposed method and one of the penalty methods are applied to MOPTA08, which is a multi-constrained optimization benchmark problem, to compare their performances. While penalty method cannot satisfy all constraints, the proposed method can satisfy them.

1. はじめに

計算機の性能向上に伴い、進化計算手法の一つである遺伝的アルゴリズム (Genetic Algorithm: GA) の工学的応用が盛んである [1][2]。工学的応用をはじめとする実問題の多くは、ある制約の下で目的関数を最適化する制約付き最適化問題である。制約には、目的関数や設計変数に対する等式および不等式制約、また目的関数とは別の制約関数などによるものがある。特に多数の制約を持つ問題では、目的関数の最適化以前に、これらの制約を全て充足する解を探索すること自体が困難である場合がある。従来では、制約付き最適化問題に対して、制約違反量や制約違反数に応じて評価値を悪化させるペナルティ法 [3] が多く用いられてきた。ここで、制約違反量とは、各制約において制約を違反している量 (の総和) であり、制約違反数とは、充足できていない制約の数である。探索においては、ペナルティ係数を一定とする静的ペナルティ法 [4]、探索の進行に合わせて、ペナルティ係数を変化させる動的ペナルティ法 [5]、解集団の状態に応じてペナルティ係数を変化させる適応的ペナルティ法 [6] がある。一般的に制約には、充足することが難しい制約と易しい制約がある。しかし、ペナルティ法では複数の制約を同時に満たそうとするため、GA の性質上、先に易しい制約を充足する方向に探索が進む。その結果、探索の後半にはいわば局所解に陥ってしまい、難しい制約を充足することが困難となる。そうなってしまうと、もはや難しい制約を充足するのではなく、目的関数の最適化を行う方向のみに探索が進んでしまう。こうして、難しい制約を充足することなく収束、すなわち局所解に陥ってしまう。そのため、難しい制約を充足するためには、探索の方向、すなわち充足していく制約に優先順位を設ける必要があると考えられる。

そこで本稿では、制約の難易度に着目する。個体選択の際に、より難易度の高い制約を充足している個体を優先して選択することで、難易度の高い制約から順に充足する、段階的制約

充足手法を提案する。多数制約付き最適化のベンチマーク問題である MOPTA08 [7] に、提案手法と従来手法である静的ペナルティ法を適用し、提案手法の有効性を示す。

2. 関連研究

ここでは、多数制約付き問題に対するアプローチとして、先行研究である 2 例を紹介する。

2.1 二段階の非支配ソーティングと支配性交配による制約付き多目的最適化

宮川らは、評価値と制約違反量を別々に扱い、満たされていない制約が存在する実行不可能解から、可能解へ進化させる方法 [8] に着目し、制約違反量と評価値による二段階の非支配ソーティングを用いた親集団選択と、探索方向への解の収束を促す支配性交配を用いるアルゴリズムを提案した [9]。

宮川らの手法では、解集団を、制約違反量に関して支配されない順に、フロントと呼ばれる解集合に分類する。さらにそれぞれのフロントを、評価値に関して支配されない順に細分化し、上位フロントから順に、解集団の半分を下記の交叉における片方の親集団にする。これにより、制約違反量による優劣が同一となる解について、評価値により優劣関係を決定できるため、評価値の良好な実行可能解を獲得しやすくなる。さらに交配においては、親個体集団から片方の親を選択した後、その親を評価値で支配する解を、実行不可能解を含む全ての解集団から選出し、その中からもう片方の親を選択する。これにより、実行不可解の遺伝子も活用して探索の収束を促進させる。

2.2 看護師勤務表作成問題に対する進化型多目的最適化に基づくアプローチ

看護師勤務表作成問題は、看護師の勤務表を様々な制約条件の下で作成する問題である。また特徴として、制約の強さには、その制約を充足しないと勤務表として成り立たないハード制約と、出来る限り満たしてほしいソフト制約の 2 種類がある。全てのハード制約を満たす勤務表を作成することが第一の目標である。この問題に対して、渡邊・奥寺は、以下の方針に

連絡先: 大江良, 名古屋大学大学院工学研究科, 名古屋市中種
区不老町, TEL:052-789-2793, ooe@cplx.cse.nagoya-
u.ac.jp

基づいたアルゴリズム設計を行い、効率的な解探索の実現を試みている [10].

初めに、初期個体生成および各遺伝的操作を行う際に、特定の制約を必ず充足するように解操作をする。これにより、探索領域を削減する。次に、現時点で最良と思われる個体を、必ず各遺伝的操作の適用対象とする。これは、優良個体に対して、徹底した局所探索を行い、より短い期間でハード制約を満たす解を導出することを目的としている。最後に、探索が停滞していると判断した場合には、現在の最良個体を種に、いくつかの新規個体を生成し、それらを用いて探索を行う。その結果、一時的に解は悪化するが、局所解からの早期脱出を実現する。

3. 提案手法

本稿で提案する手法では、各制約について、制約を充足している個体の割合により各制約の難易度を定義する。親個体選択において、難易度の高い制約を充足している個体を優先して親個体として選択することで、難易度の高い制約から易しい制約へ、段階的に制約を充足する。全ての制約を充足した後、目的関数を最適化する。また、次世代個体群選択においては、制約違反数の小さい個体を優先して次世代に残すことで、探索全体では、全ての制約を充足する方向に圧力をかける。

3.1 制約の難易度の定義

初期個体をランダムに生成し、各制約を充足している個体の割合を求める。各制約について、充足している個体の割合が低い制約ほど、難易度の高い制約として順位付けする。

3.2 親個体選択方法

交叉における親個体選択において、より難易度の高い制約を充足している個体を優先するトーナメント選択を行う。親個体の選択手順を以下に示す。

Step 1: 個体群からランダムに N_t 個の個体を選ぶ。ここで、 N_t はトーナメントサイズである。

Step 2: 選ばれた N_t 個の個体のうち、最も難易度の高い制約（対象制約）を充足している個体を親個体として選択する。

- 対象制約を充足している個体がない場合
図1のように、その制約に対する制約違反量が最も小さい個体を親個体として選択する。
- 対象制約を充足している個体が1個体だけある場合
図2のように、その個体を親個体として選択する。
- 対象制約を充足している個体が複数ある場合
図3のように、次に難易度の高い制約を充足している個体を親個体として選択する。
- 全ての制約を充足している個体が複数ある場合
図4のように、最も評価値の良い個体を親個体として選択する。

以上の手順を2度行い、親個体として2個体選択する。

3.3 次世代個体群選択方法

提案手法では、次世代に残す個体を選ぶ基準として、制約違反数、制約違反量、評価値をこの優先順位で用いる。次世代個体群の選択手順を以下に示す。

Step1: 図5のように、親個体群と子個体群を合わせ、各個体を制約違反数の小さい順に並べる。

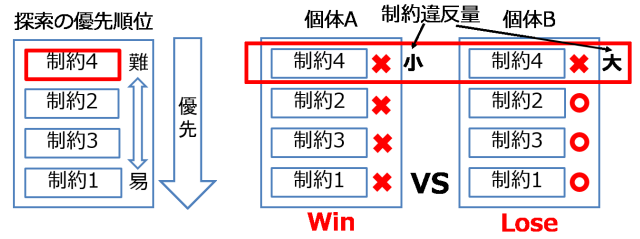


図 1: 制約を充足している個体がない場合

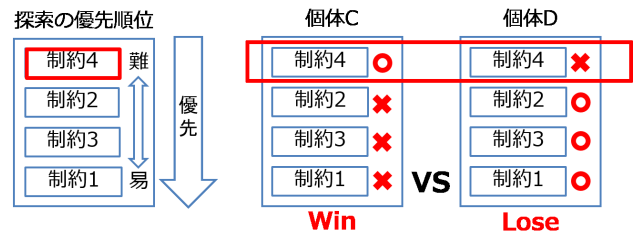


図 2: 制約を充足している個体が1個体だけある場合



図 3: 制約を充足している個体が複数ある場合

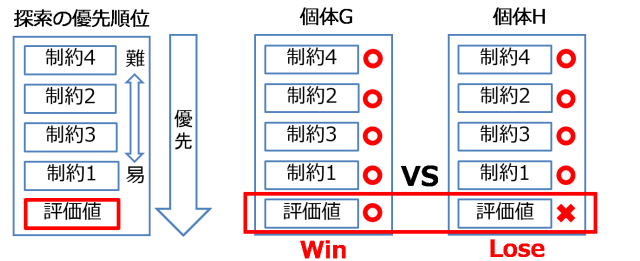


図 4: 全ての制約を充足している個体が複数ある場合

- 制約違反数が等しい個体が複数ある場合
制約違反量の小さい個体を上位とする。
- 全ての制約を充足する個体が複数ある場合
評価値の良い個体を上位とする。

Step2: 並べられた個体のうち、上位から次世代個体群の個体数だけ次世代個体として選択する。

4. 実験

4.1 実験条件

本実験では、多数制約付き最適化問題のベンチマーク問題であり、ウェブ上で公開されている MOPTA08[7] を用いた。MOPTA08 は、次式で表される 68 個の制約を持つ、1 目的最小化問題である。

	制約違反数	制約違反量 (総和)	目的関数 (最小化)
個体S	0	0	250
個体L	0	0	274
⋮			
個体N	6	2.35	312
個体A	6	5.19	281
⋮			
個体F	13	8.56	241

図 5: 個体のソート方法

$$\text{Minimize } f(\mathbf{x}) \quad (1)$$

$$\text{subject to } g_i(\mathbf{x}) \leq 0 \quad (i = 1, 2, \dots, 68) \quad (2)$$

解 \mathbf{x} の制約違反量ベクトル $\mathbf{v}(\mathbf{x})$, 制約違反量 $\Omega(\mathbf{x})$ は次式で定義される.

$$\mathbf{v}(\mathbf{x}) = (v_1(\mathbf{x}), v_2(\mathbf{x}), \dots, v_{68}(\mathbf{x})) \quad (3)$$

$$v_i(\mathbf{x}) = \begin{cases} g_i(\mathbf{x}) & (g_i(\mathbf{x}) \geq 0) \\ 0 & (\text{otherwise}) \end{cases} \quad (4)$$

$$\Omega(\mathbf{x}) = \sum_{i=1}^{68} v_i(\mathbf{x}) \quad (5)$$

提案手法の有用性を示すため、従来手法との比較を行う。2章で紹介した宮川らの手法 [9] では、制約違反量による支配関係を用いるが、MOPTA08 には 68 個もの制約が存在する。そのため、制約違反量による支配関係が成り立たないと考えられる。また、渡邊らの手法 [10] では、制約を必ず充足しなければならないハード制約と、出来る限り充足したいソフト制約に区別して扱う。MOPTA08 では、全ての制約をハード制約とし、評価値をソフト制約をみなすことで、この手法を適用できる。しかし、MOPTA08 に関する事前知識がないことを前提とするため、この手法の最大の特徴である解操作を行うことができない。

そこでここでは、静的ペナルティ法を従来手法とし、提案手法との比較を行った。従来手法では、ペナルティとして、制約違反数の 10 倍、制約違反量の 10 倍を評価値に加えた。提案手法と従来手法のどちらも、探索個体数を 100 個体とし、100 世代で探索打ち切りとした。また、トーナメントサイズ N_t は 10 とした。従来手法と提案手法、それぞれ 10 試行の平均値により比較を行った。手法の評価指標として、制約違反数、制約違反量、評価値を用いた。これらの評価指標は、いずれも値が小さいほど高い性能を表している。

4.2 実験結果

実験の結果を図 6 から図 8 に示す。図 6 および図 7 より、従来手法では、100 世代探索を行っても、充足できない制約が存在することがわかる。また図 8 より、探索が進むにつれて、易しい制約を充足した後、難しい制約を充足しないまま、評価値のみを最小化する探索が行われていることが確認できる。こ

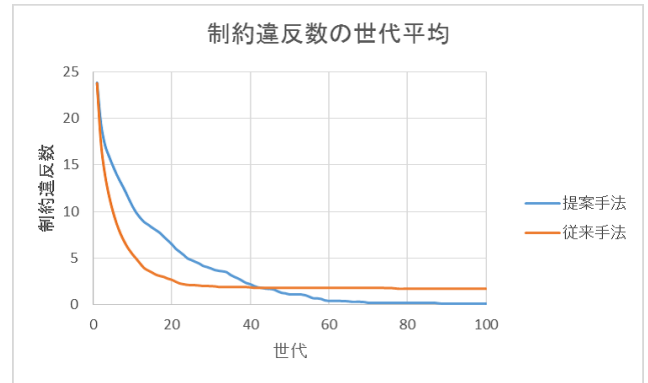


図 6: 制約違反数の世代平均

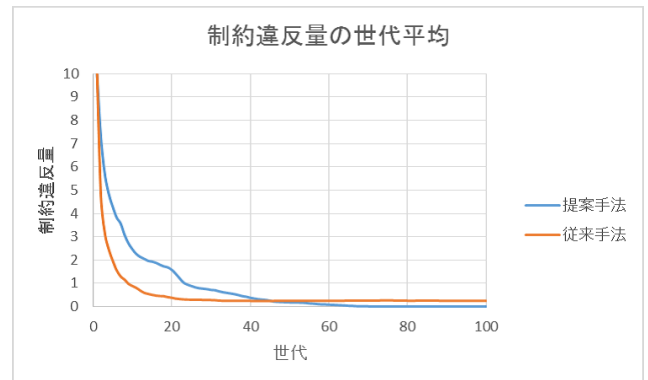


図 7: 制約違反量の世代平均

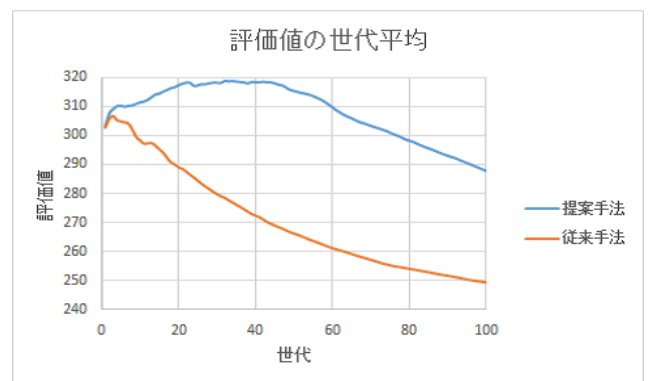


図 8: 評価値の世代平均

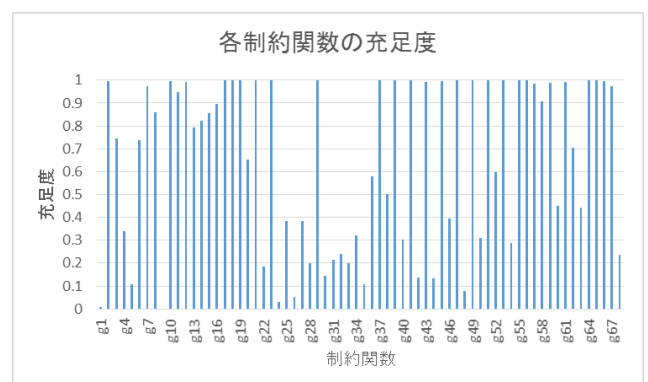


図 9: 各制約の充足度

ここで、従来手法において、ほぼ全ての試行で充足することのできなかった制約関数は、 $g_1(\mathbf{x})$ および $g_9(\mathbf{x})$ であった。図9より、これらの制約関数は、提案手法で定義した難易度によると、最も難易度の高い制約関数であることが確認できる。

一方、図6および図7より、提案手法では、約60世代探索を行った時点で、全ての制約を充足できたことがわかる。図8より、提案手法では、探索の序盤では、制約を充足することを優先しているため、評価値は悪化しているが、約60世代で全ての制約を充足した後は、評価値が最小化されていっていることが確認できる。提案手法の特徴である、制約を段階的に充足することで、全ての制約を充足すること、および、全ての制約を充足した後に、評価値の最適化が行われたことがわかる。

5. まとめ

本稿では、制約の難易度を定義し、より難易度の高い制約から段階的に充足する段階的制約充足法を提案した。提案手法では、ランダムに生成された初期個体の制約充足度により、制約の難易度を定義する。また、親個体選択の際には、より難易度の高い制約を充足している個体を優先し、選択することで、段階的に制約を充足していくという特徴を持つ。さらに次世代個体群選択においては、制約違反数と制約違反量の小さい個体を優先して次世代に残すことで、全ての制約を充足する方向に圧力をかける。従来手法である静的ペナルティ法と提案手法を、多数制約付き最適化問題のベンチマーク問題である MOPTA08 に適用し、従来手法では充足することのできない全ての制約を、提案手法では充足することができることを示した。今後は、初期個体以外の探索個体を用いた、制約の難易度の決定方法の検討や、各制約間の相関を考慮した制約充足法に対する検討、人手で作られた設計案や現行のモデルなどの見本解を種とした場合の有効な探索方法に対する検討を行っていく予定である。

6. 謝辞

本研究は、文部科学省科学研究費（基盤研究（C）, No15K00336）の補助を得て遂行された。

参考文献

- [1] Dipankar Dasgupta and Zbigniew Michalewicz. *Evolutionary algorithms in engineering applications*. Springer Science & Business Media, 2013.
- [2] Carlos A Coello Coello and Gary B Lamont. *Applications of multi-objective evolutionary algorithms*, Vol. 1. World Scientific, 2004.
- [3] Efrén Mezura-Montes and Carlos A Coello Coello. Constrained optimization via multiobjective evolutionary algorithms. In *Multiobjective problem solving from nature*, pp. 53–75. Springer, 2008.
- [4] Abdollah Homaifar, Charlene X Qi, and Steven H Lai. Constrained optimization via genetic algorithms. *Simulation*, Vol. 62, No. 4, pp. 242–253, 1994.
- [5] Jeffrey A Joines and Christopher R Houck. On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GA's.

In *Evolutionary Computation, 1994. Proceedings of the First IEEE Conference on IEEE World Congress on Computational Intelligence.*, pp. 579–584. IEEE, 1994.

- [6] Raziye Farmani and Jonathan A Wright. Self-adaptive fitness formulation for constrained optimization. *IEEE Transactions on Evolutionary Computation*, Vol. 7, No. 5, pp. 445–455, 2003.
- [7] F. Anjos. Mopta 2008 benchmark. <http://www.miguelanjos.com/jones-benchmark>.
- [8] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMI Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 2, pp. 182–197, 2002.
- [9] 宮川みなみ, 佐藤寛之. 二段階の非支配ソーティングと指向性交配による制約付き多目的最適化. *進化計算学会論文誌*, Vol. 3, No. 3, pp. 185–196, 2013.
- [10] 渡邊真也, 奥寺将至. 看護師勤務表作成問題に対する進化型多目的最適化に基づくアプローチの提案. *進化計算学会論文誌*, Vol. 5, No. 3, pp. 32–44, 2014.