3J-06

多目的最適化における遺伝的アルゴリズムの初期個体群に関する評価

藤本 大輝[†] 久保田 光一[‡]

中央大学大学院 理工学研究科[†] 中央大学 理工学部[‡]

概要: 遺伝的アルゴリズムは生物の進化の過程をモデルにした多点探索である．優良な解を同時に保持するため, 最適解が複数存在する多目的最適化問題に適応されることがある．多目的最適化問題は適当な重み付けによる単一目的最適化問題への変換が可能であるが, 全ての最適解を列挙することは難しいとされている．本研究では多目的最適化問題の 1 つとして, 多目的 0-1 ナップサック問題を例に取り上げ, 遺伝的アルゴリズムの初期化時に生成された個体が最終結果にどう影響するのかを評価した．

キーワード: 多目的最適化, 遺伝的アルゴリズム

1 はじめに

最適化とは, ある目的を良くするための解を見出すことをいう．現実世界では企業の戦略の意思決定や工学設計などで利用される．通常の最適化は目的が 1 つだけの単一目的最適化である．しかし, 現実には複数の目的を同時に良くしたい場合があり, その場合は多目的最適化という．一般に多目的最適化の解は複数あり, 最終的な意思決定はそれらのいずれかを選ぶ．

多目的最適化を解く 1 つの手法は遺伝的アルゴリズムを利用したものである．本研究では多目的最適化問題の 1 つである多目的 0-1 ナップサック問題を例に取り上げ, 初期化時に生成された個体が最終結果にどう影響するのかを評価する．

2 多目的最適化

通常の最適化問題は次の数理計画問題として定式化される．

$$\begin{aligned} \min \quad & f(\boldsymbol{x}) \\ \text{s.t.} \quad & \boldsymbol{x} \in X \end{aligned}$$

関数 f は目的関数, $\boldsymbol{x}$ は実行可能解, X は実行可能集合という．最適化問題では目的関数 f を最小にするような実行可能解 $\boldsymbol{x}$ を求めることが目標である．なお, 目的関数を最大化する場合は $-f(\boldsymbol{x})$ を最小化することを考えればよい．

多目的最適化問題は次の数理計画問題として定式化される．

$$\begin{aligned} \min \quad & \boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \ldots, f_p(\boldsymbol{x})) \\ \text{s.t.} \quad & \boldsymbol{x} \in X \end{aligned}$$

多目的最適化では目的関数が p 個あり, この p 次元の目的関数ベクトル $\boldsymbol{f}$ を最小にする実行可能解 $\boldsymbol{x}$ を求めることが目標である．目的関数ベクトルの大小関係は次項の優越関係で定義される．

2.1 パレート最適

一般に多目的最適化問題は目的関数同士が競合することが多く, 最適解は一意に定まらない．そこで次のパレート解を導入する．パレート解とはこれ以上すべての目的関数を同時に改善できない解のことを言う．

優越 $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^*)(\forall i = 1, \ldots, p)$ かつ $f_j(\boldsymbol{x}) < f_j(\boldsymbol{x}^*)(\exists j \in \{1, \ldots, p\})$ となる $\boldsymbol{x} \in X$ が存在するとき, $\boldsymbol{x}$ は $\boldsymbol{x}^*$ を (弱い意味で) 優越するという．

パレート解 $\boldsymbol{x}^*$ を優越する $\boldsymbol{x} \in X$ が存在しないとき, $\boldsymbol{x}^*$ をパレート解という．

非優越解 $\boldsymbol{x}$ がパレート解であるとき, $\boldsymbol{f}(\boldsymbol{x})$ を非優越解 (nondominated solution) という．

2.2 重み付け線形和によるスカラー化法

重み付け線形和によるスカラー化法は多目的最適化問題に対する 1 つの解法である．任意の正の重み $w_1, w_2, \ldots, w_p$ に対し,

$$F(\boldsymbol{x}) = w_1 f_1(\boldsymbol{x}) + w_2 f_2(\boldsymbol{x}) + \cdots + w_p f_p(\boldsymbol{x})$$

を定義することで, 目的関数を F とした単一目的最適化問題に帰着できる．この手法で得られる $\boldsymbol{x}$ はパレート解であることが保証される．

この手法の問題点として, 重みのつけ方が難しいこと, 解空間が非凸な場合にすべてのパレート解を導出できないことが挙げられる．

3 遺伝的アルゴリズム

遺伝的アルゴリズム (Genetic Algorithm : 以下 GA) は 1975 年に Holland によって提案された, 生物の進化の過程をモデルにしたメタヒューリスティクスアルゴリズムである [1]．候補解を生物に見立て, “選択・淘汰”, “交叉”, “突然変異” といった操作を繰り返すことで探索を進める．

GA では解を文字列として表現する．また, この解を持った構造を GA では個体と呼ぶ．図 1 は後に説明する交叉の例であるが, 左上の個体は “10101100” という文字列で解を表している．この “0” や “1” といった要素は遺伝子と呼ばれる．

3.1 選択・淘汰

選択・淘汰は個体群から次世代に残す個体を選択し, それ以外を個体群から削除する操作である．通常は個体の持つ適応度を参照して, 個体を選択する．適応度とはその個体 (解) が問題にどれだけ適しているかを表す値である．適応度を計算する関数を適応度関数といい, ユーザーは適応度関数を適切に定義しなければならない．

3.2 交叉

交叉は親個体から新たな子個体を生成する操作である．交叉方法には m 点交叉や一様交叉が提案されている．図 1 は $m = 1$ とした一点交叉の例である．切断点の後ろを入れ替えることで新たな個体を生成している．

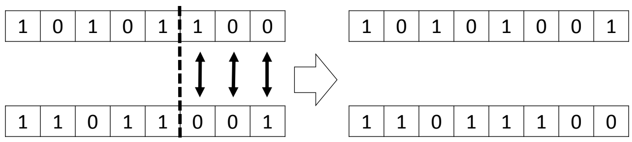

図 1 一点交叉の例．点線は切断点を表す．

3.3 突然変異

突然変異はある個体の遺伝子を書き換え, 別の個体にする操作である．GA ではランダムに選ばれた遺伝子を別の遺伝子に書き換える方法が知られている．

Evaluation of Genetic Algorithm with respect to Initial Population in Multiobjective Optimization

† Taiki FUJIMOTO, Graduate School of Science and Engineering, Chuo University

‡ Koichi KUBOTA, Faculty of Science and Engineering, Chuo University

4　遺伝的アルゴリズムの多目的最適化への応用

GA を多目的最適化問題に適用する場合, 適応度の定義が重要となる．単一目的最適化問題を対象にした GA では目的関数の値を適応度とみなすことが多い．しかし, 多目的最適化問題を対象にした GA では目的関数の値が複数あるため, スカラーである適応度に目的関数そのものを定義することは難しい．そのため, 目的関数を何らかの形でスカラー化し適応度とするか, 別の方法で個体の選択を行わなければならない．

4.1　パレートランキング法

1989 年に Goldberg は "ランク" という概念を提案した．ランクとは個体同士の重要度を表す指標で, ランクの小さい個体程重要となる．1993 年には Fonseca と Fleming によってランクを用いた Multi-objective genetic algorithm (MOGA) が提案された [2]. Goldberg が提案したランク付けの方法と, Fonseca と Fleming が提案したランク付けの方法は異なる．ここでは Fonseca と Fleming が提案したランク付けの方法を説明する．

個体 i を優越する個体の数を s_i とする．個体 i のランク r_i は $r_i = s_i + 1$ で定義される．図 2 は 2 目的最適化におけるランク付けの例である．f_1, f_2 ともに最小化するとき, 図中の点で表した個体が見つかったとするなら, 図で表されるようなランク付けがなされる．

このランクをもとに個体の優劣を決定し, 次世代に残す個体を選択する．本来の MOGA では解の広がりや間隔の均等さを考慮し, ニッチングやシェアリングという操作の後に適応度を決定している．本研究ではニッチング, シェアリングを行わずに, 単純にランクの小さい順に個体を選択している．

5　計算機実験

計算機実験では多目的 0-1 ナップサック問題を対象に, 通常の MOGA と初期個体群に重み付け線形和によるスカラー化法により導出された個体を混ぜた MOGA との比較を行った．

5.1　多目的 0-1 ナップサック問題

多目的 0-1 ナップサック問題は, p 種類の価値を持つ品物 i について, ナップサックに入れた品物の合計価値を p 種類同時に最大にする問題である．品物の個数を n, 品物 i の大きさを w_i, 品物 i の j 番目の価値を c_i^j, ナップサックの合計容量を W, 品物 i をナップサックに入れるか否かを x_i で表すと き, 次のように定式化される．ただし $x_i = 0$ は品物 i をナップサックに入れない, $x_i = 1$ は入れることを表す．

$$\begin{aligned} \max \quad & f_j(\boldsymbol{x}) = \sum_{i=1}^{n} c_i^j x_i \quad (j = 1, 2, \ldots, p) \\ \text{s.t.} \quad & \sum_{i=1}^{n} w_i x_i \leq W \\ & x_i \in \{0, 1\} \quad (i = 1, 2, \ldots, n) \end{aligned}$$

5.2　実験内容

テストケースに対して, 全探索によるパレート解数, 通常の MOGA のパレート解数, 重み付き線形和によるスカラー化法によるパレート解数, 初期個体群操作による MOGA のパレート解数を調べた．テストケースは Kirlik[4] が実験に用いたデータと同じものを使用した．重み付き線形和によるスカラー化法では, 各重みを 0 から 10 の整数に変えて計算した．MOGA では個体群数 50, 世代交代数 50 とし, 各世代でランクが小さい順に 50 個体が選択されるようにした．また, 最終結果の rank1 の個体と全探索の結果を比較し, 実際にパレート解である個体数を調査した．

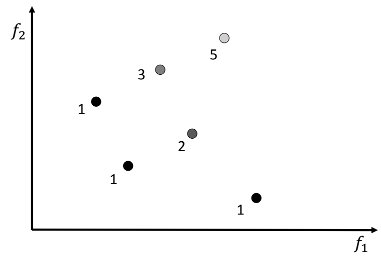

図 2　ランク付けの例．図中の数字はその個体のランクを表す．

5.3　実験結果

表 1, 表 2 はテストケース 1, 2 に対するそれぞれの手法の解数を記した表である．表中の p は目的関数, n はナップサック問題の商品数, S は全探索によって得られたパレート解数, MOGA-α は初期個体群操作を加えた MOGA を示している．商品数が 10 の場合はパレート解も少ないため, 初期個体群を操作してもあまり変化は見られない．しかし, 商品数が増えるにつれ, 初期個体群操作を加えた MOGA が良い性能を示した．他のテストケースにおいても同様の傾向を確認した．

表 1　テストケース 1 に対する実験結果.

p	n	S	MOGA	スカラー化法	MOGA-α
3	10	7	7	4	7
3	20	35	6	12	21
4	10	10	10	5	10
4	20	60	21	18	35

表 2　テストケース 2 に対する実験結果.

p	n	S	MOGA	スカラー化法	MOGA-α
3	10	19	19	5	18
3	20	35	16	15	24
4	10	13	12	7	12
4	20	143	22	32	41

6　今後の課題

実行可能解 $\boldsymbol{x}$ が実数値ベクトルである問題に対しても, 今回の初期個体群操作が有効であるか否かを調査する．

参考文献

[1] S. Forrest, and M. Mitchell. Adaptive Computation: The Multidisciplinary Legacy of John H. Holland, Communications of the ACM, Vol. 59, No. 8, pp. 58–63.

[2] C. M. Fonseca, and P. J. Fleming. Genetic Algorithms for Multiobjective Optimization: FormulationDiscussion and Generalization, ICGA, Vol. 93, 1993.

[3] 中山弘隆, 岡部達哉, 荒川雅生, 尹禮分. 多目的最適化と工学設計—しなやかシステム工学アプローチ—, 現代図書, 2008.

[4] G. Kirlik, and S. Sayın. A new algorithm for generating all nondominated solutions of multiobjective discrete optimization problems, European Journal of Operational Research, Vol. 232, No. 3, pp. 479–488.