

Adaptive PBI を用いた非マイグレーション並列化 MOEA/D

平山友駿† 佐藤裕二†

法政大学情報科学部†

1. まえがき

多目的最適化問題 (MOP) とは複数の目的関数の最小値 (または最大値) を求める問題である。各目的関数はトレードオフの関係にあるため、非劣解は複数存在する。進化型多目的最適化アルゴリズム (MOEA) は、遺伝的操作による探索を繰り返し、その複数存在する解集合 (パレート解集合) を求めるための進化計算アルゴリズムである。MOEA の性能評価には解の散らばりを示す多様性と、MOEA で解探索に用いる探索点 (個体) がパレート解集合に近づく速度が示す収束性を用いられる。代表的な MOEA である MOEA/D は MOP を部分問題に分解し、各部分問題を解くことで MOP を解くため、個体の評価を行うスカラー化関数、問題の分割を行う重みベクトルなどへの様々な改良が行われている[1]。

MOEA/D を複数の CPU コアで並列化することで、高速化を図る研究が進められている[2]。MOEA/D を並列化する場合、目的関数空間をコア数分だけ分割し、分割した領域を各ローカルメモリ (LM) に割り当てる。各 LM の領域は他の LM 内の領域の情報を保持していないため、目的関数空間のコアの境目に個体が広がらず、多様性が低下する。一般的に、LM 同士の通信を行うマイグレーションにより各 LM の個体情報を共有することで並列化 MOEA/D の多様性低下を抑制する。マイグレーションを行うには各コアの世代数の同期を行う必要があるため、実行時間が低下する。本研究ではマイグレーションを行わずに多様性を向上させる方法を提案する。MOEA/D の個体の評価にはスカラー化関数を用いる。スカラー化関数の改良は解集合に影響を与えることが分かっており、MOEA/D の研究における一つの議題となっている。MOEA/D のスカラー化関数には一般的にチェビシェフ関数が用いられるが、ペナルティ値を利用するため、

Non-Migration Parallel MOEA/D Using Adaptive PBI

†Tomoya Hirayama †Yuji Sato

†Faculty of Computer and Information Science, Hosei University

Penalty Boundary Intersection(PBI)を用いる。本論

文では、PBIのペナルティ値を用いて MOEA/D の並列化による多様性低下を緩和する方法を提案する。

2. Adaptive PBI (APBI) の提案

PBI の個体評価は以下の数式で行われる。

$$\text{minimize } g(x | \lambda, z^*) = d_1 + \theta d_2 \quad (1)$$

$$d_1 = \frac{\|(z^* - F(x))^T \lambda\|}{\|\lambda\|}, d_2 = \|F(x) - (z^* - d_1 \lambda)\| \quad (2)$$

図 1 に PBI の数式の説明図と個体の更新範囲を示す。λ は重みベクトル、z は目的関数の最大値を各成分とした理想点、d₂ は個体とそれに対応する重みベクトルまでの距離、d₁ は個体から重みベクトルに引いた垂線の足と理想点の距離を示している。θ はペナルティ値と呼ばれ、ペナルティ値を大きくすれば解の収束性、ペナルティ値を小さくすれば多様性を促すことができる。

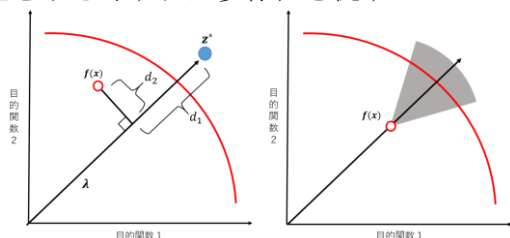


図 1 PBI の式の図と個体更新範囲

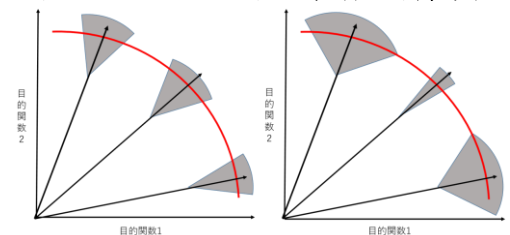


図 2 PBI, APBI における個体の更新範囲

図 2 にシングルコアでの PBI と APBI の個体の更新範囲を示す。従来の PBI におけるペナルティ値は定数である。解の多様性を向上させるためには解集合の両端の範囲を広げることが有効と考える。APBI では目的関数空間の領域中央では収束性を促し、領域の両端では多様性を促すことで、コア内の多様性を向上する方法を提案す

る。隣同士の個体のペナルティ値の差が大きい場合、解集合に偏りが生じ、多様性が低下するため、ペナルティ値の割り当ては正規分布に基づいて行う。任意のペナルティ値と正規分布の確率密度関数の積を実際のペナルティ値とする。

3. 評価実験

PBI, APBI, マイグレーションを行うチェビシェフ関数に関する3つの比較実験を行う。ベンチマーク問題にはナップサック問題を用いる。なお、すべての実験は4コアで並列化を行う。1) APBIの正規分布の分散の値による解分布の変化に関する実験、2) 世代数による多様性の変化に関する実験、3) APBIと一般的に使われるチェビシェフ関数との比較。1) では世代数を10kとして、ペナルティ値0.1としたPBIと分散が1, 10, 100, ペナルティ値を0.1の場合のAPBIを比較する。2) ではペナルティ値0.1の5k世代と500k世代でのPBIと分散100のAPBIとを比較する。3) では世代数500kにおけるチェビシェフ関数と分散100のAPBIで比較する。

1) の実験結果を図3に示す。APBIはPBIと比較してどの分散においても多様性が向上している。2) の実験結果を図4に示す。MOEAでは世代数を増やすことで、解の多様性が向上する。従来PBIも同様に、多様性が向上する。さらに、どの世代においてもAPBIはPBIより多様性に優れている。3) の実験結果を図5に示す。一般的に利用されているAPBIはPBIと比較すると多様性に優れている。チェビシェフ関数を用いて4分割した場合、解分布がスパースとなる。

4. 考察

PBIと比較してAPBIは分散の値、世代数に寄らず多様性が向上している。分散と多様性の関係は、対象とする問題や生成される個体に依存していると考ええる。APBIではコア内の領域で両端の多様性を向上した。この要因として、両端の個体評価のペナルティ値が小さいことに加え、中央と両端とでペナルティ値が違うことがあると考える。ペナルティ値がすべて同じ場合、個体の評価は常に一定であるが、一つの個体に対しても異なる評価を行えることで個体が集中することを防いでいると考える。チェビシェフ関数における目的関数空間の中央の解集合は両端に比べてパレートフロントの形がより凸になっているため、理想点から離れた多様性のある個体の評価値がより低くなり、多様性が低下したと考える。非マイグレーションではチェビシェフ関数よりAPBIは有効であると考ええる。

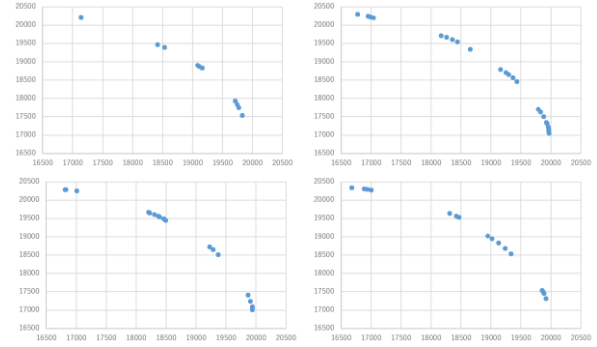


図3

左上はペナルティ値が一定のPBI,
右上が分散1, 左下が分散10,
右下が分散100のAPBI

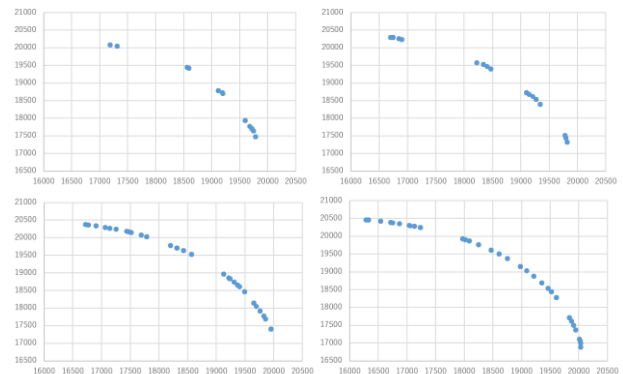


図4 左がPBI, 右がAPBI,
上段が5k世代で下段が500k世代

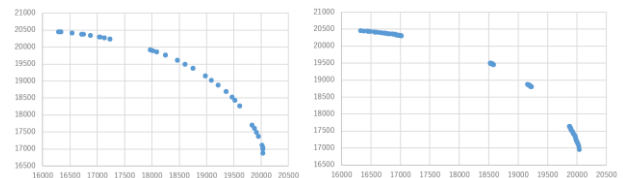


図5 左がAPBI, 右がチェビシェフ関数

5. むすび

本研究ではPBIのペナルティ値を適応的に割り当てることで非マイグレーション並列化MOEA/Dの多様性向上を図る提案を用いて実験を行い、ペナルティ値を領域によって適応的に変更した提案手法はペナルティ値が一定である従来のPBIと比較してどの世代においても多様性が向上することを示した。

参考文献

- [1] Zhang, Qingfu, and Hui Li. "MOEA/D: A multiobjective evolutionary algorithm based on decomposition." *IEEE Transactions on evolutionary computation* 11.6 (2007): 712-731.
- [2] NEBRO, Antonio J.; DURILLO, Juan J. A study of the parallelization of the multi-objective metaheuristic MOEA/D. In: *International Conference on Learning and Intelligent Optimization*. Springer, Berlin, Heidelberg, 2010. p. 303-317.