

関数値最適化のための Particle Swarm Optimization, Differential Evolution, 実数値遺伝的アルゴリズムの 探索性能に関する検討

伊藤 稔 知的情報通信研究所
田中雅博 理工学部

(受理日 2005年4月15日)

1 はじめに

近年、最適化問題の大規模化と複雑化に伴い厳密な最適解を求めるのが難しくなっている。このようなことから、必要十分な最適性を持つ近似解を実用時間内に求めることが可能なヒューリスティックな最適化手法が多く注目を集めている。代表的な手法の一つとして遺伝的アルゴリズム (Genetic Algorithm, GA) がある[1,2]。GAは生物の進化を工学的最適化手法に応用した確率的な手法であり、複数の探索点 (個体) の間において選択、交叉、突然変異などの遺伝的操作を行う多点探索法である。また、GAは目的関数の勾配情報を用いず、各探索点の評価値のみを用いて解探索を行う直接探索法である。したがって、連続値問題や離散値問題にも適用できる利点がある[2]。特に、近年では、遺伝子表現として実数値ベクトルを用いた実数値GAが非線形関数の最適化手法として注目されている[3,4,5,6,7,8,9]。

一方、近年では、連続関数の最適化手法としてGAに代わる新しい最適化手法が提案されている。J.KennedyとR.EberhartによるParticle Swarm Optimization (PSO) [10]やK.PriceとR.StornによるDifferential Evolution (DE) [11]などが代表的な手法である。これらの手法は基本的なアイデアはGAとは異なるが、多点探索法であることと直接探索法であることからGAと同様に進化的計算に分類できる。PSOとDEはGAと比較してアルゴリズムがシンプルであることと収束性の良さ、そして、様々な問題への実装の容易性などから多くの注目を集めている。

これまで、連続関数の最適化手法におけるPSOやDEのアルゴリズムの改良などの研究では比較対象としてオリジナルのアルゴリズム、あるいは、GAとの比較がしばしば行われている[12,13,14,15]。しかしながら、比較対象として用いられているGAは連続関数の最適化に適している実数値GAではなくビットGAであり、必ずしもGAと比較して性能が優れていることが示されていない。このようなことから、本論文では、比較対象として実数値GAを用いて関数値最小化問題に適用して性能の比較を行う。

本論文は「関数値最適化のための Particle Swarm Optimization, Differential Evolution, 実数値遺伝的アルゴリズムの探索性能に関する検討」と題し6章からなる。1章は序論である。2章でPSO、3章でDE、4章では実数値GAについてそれぞれ述べる。5章では関数値最小化問題をベンチマークとする比較実験を行い考察する。6章は結論で、本論文の研究成果

をまとめ、今後の課題について述べる。

2 Particle Swarm Optimization

2.1 PSOの概要

Particle Swarm Optimization (PSO) は Swarm Intelligence の1分野で GA に代わる最適化手法として 1995 年に J.Kennedy と R.Eberhart により紹介された確率的な最適化手法の1つである [10]。PSO は、それ以前の Reynolds の魚や鳥の群の中での個々の生物の振舞いと群としてのまとまりについての研究 [16] などを工学的最適化に応用したものである。PSO の基本概念は、群 (swarm) となった粒子 (particle) がお互いの情報を共有しながら解空間を探索していくもので、個々の粒子の持つ最良の情報と群で共有する最良の情報を利用して解を探索する手法である。

PSO は、従来法で必要とされる設計変数に関する関数の微分計算など複雑で時間を必要とする計算を一切必要としない非常にシンプルなアルゴリズムである。このため、PSO は非常に短い時間で最適解を発見する事が可能になる。

2.2 位置と速度の更新式

時刻 t において、 n 次元の探索空間と N 個の粒子により構成される群を仮定すると、 i 番目の粒子は n 次元の位置ベクトル $\mathbf{X}_i(t)$ で表現される。PSO による解探索は、各個体が持つ自分自身の最良解の位置ベクトル $\mathbf{P}_i(t)$ 、群で共有している最良解の位置ベクトル $\mathbf{P}_g(t)$ 、および各個体の速度ベクトル $\mathbf{V}_i(t)$ の3つのベクトルの線形結合として、新しい探索点 $\mathbf{X}_i(t+1)$ が生成される。以下に新しい探索点の生成式を示す。図 1 に粒子の移動の概略図もあわせて示す。

$$\mathbf{V}_i(t+1) = w\mathbf{V}_i(t) + c_1r_1\{\mathbf{P}_i(t) - \{\mathbf{X}_i(t)\}\} + c_2r_2\{\mathbf{P}_g(t) - \{\mathbf{X}_i(t)\}\} \quad (1)$$

$$\mathbf{X}_i(t+1) = \mathbf{X}_i(t) + \mathbf{V}_i(t+1) \quad (2)$$

ここで、 $i(=1, 2, \dots, N)$ は各粒子のインデックス、 w は Inertia Weight で速度に対する重み係数で正の値である。 c_1 と c_2 は各項に対する重み係数で正の値である。 r_1 と r_2 は $[0, 1]$ の一様乱数を表している。式(1)において、新しい速度ベクトル $\mathbf{V}_i(t+1)$ の各成分があらかじめ設定した範囲 $[-V_{max}, V_{max}]$ 内かどうかチェックし、範囲を超えていた場合は $\pm V_{max}$ に設定する。

2.3 PSOの基本アルゴリズム

PSO の基本アルゴリズムを以下に示す。

1. 初期粒子の位置ベクトル \mathbf{X}_i と速度ベクトル \mathbf{V}_i をランダムに初期化。
2. 各粒子の評価。
3. 探索開始から現在までの各粒子の最良点 $\mathbf{P}_i(t)$ の保存。

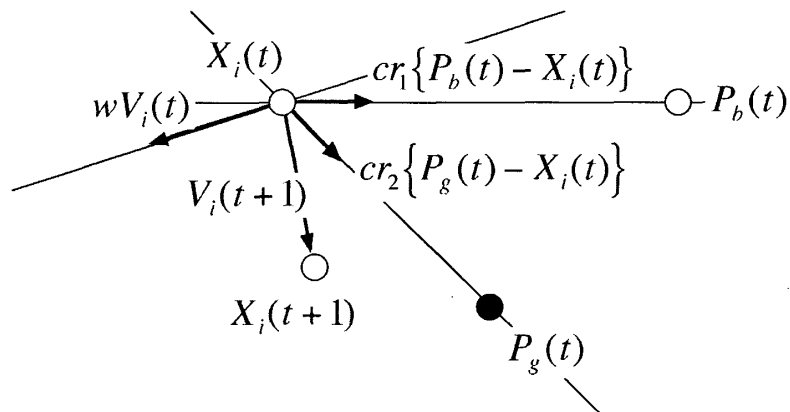


図1: PSOにおける粒子の動き

4. 現在の群の最良点 $P_g(t)$ の保存。
5. 式(1)と(2)を用いて新しい粒子の生成。
6. 2.~5.までを終了条件を満たすまで繰り返す。

3 Differential Evolution

Differential Evolution (DE) は1995年にK.PriceとR.Stornにより導入された確率的な最適化手法の1つであり、進化的計算手法に分類される[11]。DEでは従来法で必要とされる設計変数に関する関数の微分計算など複雑で時間を必要とする計算を必要としないシンプルなアルゴリズムであり、問題への実装も比較的容易に行うことができる。DEのアルゴリズムは改良手法を含め様々な手法が提案されているが、本論文では代表的なDE/rand/1/expを用いる。以下にアルゴリズムを示す。図2に新しいベクトルの生成の概略図をあわせて示す。

1. 各ベクトル $X_i (i=0, 1, \dots, n-1)$ の初期化と評価。
2. ランダムに目標ベクトル X_t を選択。
3. ランダムに3つのベクトル X_a, X_b, X_c 。ここで、 X_t と X_a, X_b, X_c はすべて異なるベクトルである。
4. 以下の式を用いて突然変異を行う。Fは突然変異率で正の定数である。

$$V = X_a + F \times (X_b - X_c) \quad (3)$$

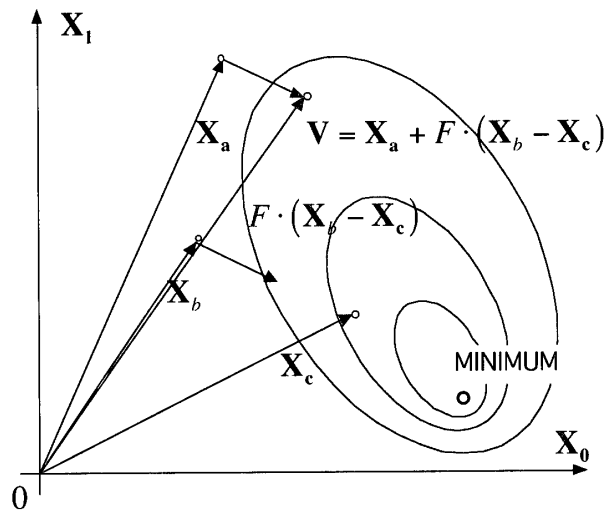


図2: DEにおける新しいベクトルの生成

5. X_t と V を交叉率 CR で交叉し新しいベクトル X_o を生成する。 CR は正の定数である。
6. X_t と X_o の適応度を比較して、適応度の高いベクトルを次世代に残す。
7. 終了条件を満たすまで2.~6.を繰り返す。

4 実数値遺伝的アルゴリズム

4.1 実数値GAの概要

実数値GAは、個体の遺伝子表現として従来の2値表現（グレイコード・バイナリーコード）ではなく実数値ベクトルを用い、変数の連続性を考慮した交叉手法を導入したGAである[3,4]。実数値GAは、2値表現を用いたGAと比較し優れた探索性能を持っている。連続性を考慮した交叉手法として、小野らによるUNDX[5]やUNDXを多親拡張した喜多らによるUNDX-m[6]、および、樋口らによるSPX[7]などが提案され実数値GAの交叉手法として広く用いられている。

一方、優れた遺伝子表現や交叉手法を用いたとしても、個体集団の多様性維持に劣った世代交代モデルを用いては、優れた解の探索を行うことができない。最適化が難しい問題においてGAの性能を十分に引き出すためには、個体集団の慎重な絞り込み、つまり、淘汰圧力の低い選択・世代交代を行う必要がある。これらの点を考慮した世代交代モデルとして佐藤らによるMGGモデルが提案されている[8]。MGGモデルは、個体の淘汰圧力を可能な限り小さくすることにより、探索の前半における初期収束や探索後半における解探索の停滞を効果的に抑制している。

本論文では、実数値GAの計算モデルとしてMGGモデルとUNDXを組合わせたモデルを用いる。

4.2 Minimal Generation Gap, MGG

佐藤らにより提案されたMGGモデル[8]は、初期収束や解探索の停滞を防ぐのに効果的な世代交代モデルである。このMGGモデルでは、まず、個体集団中からランダムに一組の親個体を選択する。つぎに、この親個体に対して複数回の交叉の操作を行い複数個の子個体を生成する。生成された子個体とその親個体からなる家族集団から個体の評価値（適応度）の最も優れたエリート個体と、家族集団からルーレット選択により選択された個体の合計2個体を次世代に生き残る個体として選択し、個体集団に戻す。

以下にMGGモデルのアルゴリズムを示す。

1. 個体集団 P から親個体 $\mathbf{x}^1, \mathbf{x}^2$ をランダムに選択する。
2. 親個体 $\mathbf{x}^1, \mathbf{x}^2$ に c 回交叉の操作を適用し子個体 $\mathbf{y}^1, \dots, \mathbf{y}^c$ を生成する。
3. 親個体と子個体を合わせた集団を家族集団 F とする。
4. F から最良の評価値を持つ個体を選択し \mathbf{s}^1 とする。
5. F から個体の評価値に基づいたルーレット選択によりもう1個体選択し \mathbf{s}^2 とする。
6. 個体群の親個体 $\mathbf{x}^1, \mathbf{x}^2$ を選択した $\mathbf{s}^1, \mathbf{s}^2$ と入れ替える。

4.3 単峰性正規分布交叉, UNDX

小野らにより提案されたUNDX[5]は、設計変数間に依存関係のある関数においても優れた探索性能を持つことが示されている。このUNDXは3つの親個体により決定される正規分布乱数を用いて子個体の生成を行っている。子個体は2つの親個体を結ぶ直線の周辺に正規分布に従い生成され、第3の親個体は正規分布の標準偏差を決定するために補助的に用いられている。正規分布の標準偏差は、両親を結ぶ方向の成分については両親間の距離に比例し、それ以外の軸の成分については、第3の親と両親を結ぶ直線との距離に比例する。

以下に、UNDXのアルゴリズムを示す。

1. 3つの親個体 $\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3$ を選択する。
2. 親個体 $\mathbf{x}^1, \mathbf{x}^2$ の中点を $\mathbf{x}^p = (\mathbf{x}^1 + \mathbf{x}^2)/2$ とする。
3. 親個体 $\mathbf{x}^1, \mathbf{x}^2$ の差ベクトルを $\mathbf{d} = \mathbf{x}^1 - \mathbf{x}^2$ とする。
4. 親個体 $\mathbf{x}^1, \mathbf{x}^2$ を結ぶ直線を主探索直線と呼び、親個体 \mathbf{x}^3 から主探索直線までの距離を D とする。
5. 子個体 \mathbf{x}^c を以下の式に従い生成する。

$$\mathbf{x}^c = \mathbf{x}^p + \xi \mathbf{d} + D \sum_{i=1}^{n-1} \eta_i \mathbf{e}_i \quad (4)$$

$$\xi \sim N(0, \sigma_\xi^2), \eta_i \sim N(0, \sigma_\eta^2)$$

この式で n は探索空間の次元数、 $N(0, \sigma^2)$ は平均 0、分散 σ^2 の正規分布を示し、 e_i は主探索直線に直交する部分空間の正規直交基底ベクトルを示している。

この UNDX により生成される子個体の成分については、式 1 の右辺、第 1、第 2 項の和を主探索成分、第 3 項を副探索成分と呼ぶ。また、喜多らは σ_ξ と σ_η の推奨値として $\sigma_\xi = 1/2$ 、 $\sigma_\eta = 0.35/\sqrt{n}$ を推奨している [9]。

5 数値実験と考察

本論文では、関数最適化のための代表的な手法である実数値 GA と新しい最適化手法である PSO と DE の比較実験を行う。ベンチマーク問題としては代表的な 4 つのテスト関数を用いて最適解発見回数と評価計算回数について比較を行う。

5.1 ベンチマーク問題

表 1 に示す 4 種類のテスト関数 [17, 18] を用いて数値実験を行う。 F_1 は単峰性の Sphere 関数、 F_2 は単峰性かつ設計変数間に依存関係のある Rosenbrock 関数。 F_3 は多峰性の Rastrigin 関数、 F_4 は多峰性かつ設計変数間の依存関係のある Griewank 関数である。表 2 は各関数の次元数と定義域を示してある。一般に、多峰性関数の方が単峰性関数より最適化が困難である。また、関数中に非線形成分を含む場合を設計変数間に依存関係があると呼び、変数間に依存関係がある関数は最適化が困難である。本論文では、関数値を最小化する関数値最小化問題を対象としている。

5.2 数値実験の設定

表 3 に数値実験の設定を示す。各手法とも、個体数を 100 個体、最大探索回数を 500 万回とする。終了条件として関数の評価値が $1.0e^{-10}$ 以下になった場合、最適解を発見したと見なし探索を終了する。試行回数は初期値を変えて 30 試行行う。また、各アルゴリズムの固有パラメータは PSO については予備実験結果、DE と実数値 GA については推奨値を用いる。PSO における速度制限 V_{max} は各関数の定義域とする。

5.3 数値実験の結果と考察

表 4～7 に各テスト関数における各手法の最適解発見回数と最適解発見までに必要とする評価計算回数を示す。表 4 のテスト関数 F_1 の結果より、各手法とも全ての試行で最適解を発見することができている。DE が最も優れた性能を示しているが、30 次元の場合、評価計算回数が 20 次元の場合の約 3 倍増加しており、他の手法より評価計算回数の増加率が大きい。つづいて、表 5 のテスト関数 F_2 の結果より、DE だけが全ての試行で最適解を発見できている。PSO は 1 度も最適解を発見することができおらず、実数値 GA も次元数の増加にともない最適解を発見できない試行が増えている。これは、 F_2 は単峰性の関数であるが設計変数間に依存関係のある関数のためと考えられる。

つぎに、表 6 のテスト関数 F_3 の結果より、DE だけが全ての試行で最適解を発見できている。PSO と実数値 GA については次元数の増加に伴い最適解を発見する試行が少なくなっている。DE についても、30 次元の場合、評価計算回数が 20 次元の場合の約 3 倍増加

表 1: テスト関数

$$F_1(x_i) = \sum_{i=1}^n x_i^2$$

$$F_2(x_i) = \sum_{i=1}^n \left[100(x_1^2 - x_i)^2 + (x_i - 1)^2 \right]$$

$$F_3(x_i) = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i))$$

$$F_4(x_i) = 1 + \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \left(\cos\left(\frac{x_i}{\sqrt{i}}\right) \right)$$

表 2: 各テスト関数の次元数と定義域

テスト関数	次元数	定義域
Sphere 関数 F_1	$dim. = 10, 20, 30$	$-5.12 < x_i < 5.12$
Rosenbrock 関数 F_2	$dim. = 10, 20, 30$	$-2.048 < x_i < 2.048$
Rastrigin 関数 F_3	$dim. = 10, 20, 30$	$-5.12 < x_i < 5.12$
Griewank 関数 F_4	$dim. = 10, 20, 30$	$-512 < x_i < 512$

表 3: DE、PSO、実数値 GA のパラメータ設定

	DE	PSO	実数値 GA
個体数	100	100	100
最大探索回数	5000000	5000000	5000000
試行回数	30	30	30
終了条件	$1.0e^{-10}$	$1.0e^{-10}$	$1.0e^{-10}$
アルゴリズムの 固有パラメータ	突然変異率 $F=0.5$ 交叉率 $CR=0.9$	Inertia Weight $w=0.6$ $C_1=C_2=2.0$	$\sigma_\xi=1/2$ $\sigma_\eta=0.35/\sqrt{n}$

表4: Sphere関数 F_1 における性能比較 (評価計算回数、最適解発見回数)

	DE	PSO	実数値 GA
10 dim.	29028 (30/30)	316773 (30/30)	230473 (30/30)
20 dim.	94207 (30/30)	414156 (30/30)	340533 (30/30)
30 dim.	270165 (30/30)	497146 (30/30)	450233 (30/30)

表5: Rosenbrock関数 F_2 における性能比較 (評価計算回数、最適解発見回数)

	DE	PSO	実数値 GA
10 dim.	63327 (30/30)	5000000 (0/30)	1362046 (30/30)
20 dim.	238162 (30/30)	5000000 (0/30)	4985260 (8/30)
30 dim.	706847 (30/30)	5000000 (0/30)	5000000 (0/30)

表6: Rastrigin関数 F_3 における性能比較 (評価計算回数、最適解発見回数)

	DE	PSO	実数値 GA
10 dim.	84578 (30/30)	445306 (15/30)	1580503 (23/30)
20 dim.	324128 (30/30)	4848726 (2/30)	4024483 (7/30)
30 dim.	1016356 (30/30)	5000000 (0/30)	5000000 (0/30)

表7: Griewank関数 F_4 における性能比較 (評価計算回数、最適解発見回数)

	DE	PSO	実数値 GA
10 dim.	137426 (30/30)	4944203 (1/30)	2308656 (18/30)
20 dim.	115836 (30/30)	2816680 (15/30)	642153 (30/30)
30 dim.	330761 (30/30)	2392926 (18/30)	1097540 (29/30)

しており、探索性能の低下率が大きい。最後に、表7のテスト関数 F_4 の結果より、DEだけが全ての試行で最適解を発見することができている。

数値実験の結果をまとめると、DEのみが全ての試行で最適解を発見することができた。しかし、高次元の問題において最適解を発見するまでに必要とする評価計算回数が大きく増加した。PSOについては、単峰性でも設計変数間に依存関係のある関数や多峰性の関数では最適解を発見できない試行が多く見られた。実数値GAでは、PSOと比較して設計変数間に依存関係のある関数や多峰性関数で最適解を発見できているが、高次元の問題においては最適解を発見できない試行が多く見られた。

本論文では、個体数やアルゴリズムのパラメータについて詳しく検討を行っていない。実数値GAは、個体数を増やし十分な探索を行えば最適解を発見することが可能である。しかし、多くの時間を必要とする。PSOについても、パラメータを調整すれば、ある程度性能は向上する。しかし、初期パラメータの設計に多くの労力を必要とする。これに対して、DEでは高次元の問題における性能の低下が見られたが、比較的ルーズに設定したパラメータでも十分な探索を行うことができた。このことから、DEはPSOや実数値GAと比較して、設計者やユーザーにとって扱いやすい手法と考えられる。

6 おわりに

本論文では、進化的計算手法として代表的な実数値GAと、近年、GAに代わる新しい最適化手法として提案され注目されているPSOとDEの性能比較を関数値最小化問題に適用して行った。数値実験の結果、DEが他の手法と比較して優れた性能を示した。

今後の課題として、DEについては、高次元の問題における性能評価、アルゴリズムのパラメータに関する解析。PSOについては、アルゴリズムのパラメータに関する解析が上げられる。

謝辞

本研究の一部は日本文部科学省のオープン・リサーチ・センター整備事業による私学助成を得て行われた（平成16年度-20年度）。

参考文献

- [1] D.E.Goldberg : Genetic Algorithm in Search, Optimization and Machine Learning, Addison-Wesley, 1989
- [2] 山村雅幸, 小林重信 : 遺伝的アルゴリズムの工学的応用, 工知能学会誌, Vol.9, 506/511, 1994
- [3] L.J.Eshelman and J.D.Schaffer : Real Coded Genetic Algorithms and Interval Schemata, Foundations of Genetic Algorithms 2, 187/202, 1993
- [4] 小野 功, 山村雅幸, 喜多 一 : 実数値GAとその応用, 人工知能学会誌, Vol.15, 259/266, 2000

- [5] 小野 功, 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, 1146/1155, 1999
- [6] 喜多 一, 小野 功, 小林重信 : 実数値 GA のための正規分布交叉の多数の親を用いた拡張法の提案, 計測自動制御学会論文集, Vol.36, 875/883, 2000
- [7] 樋口隆英, 筒井茂義, 山村雅幸 : 実数値 GA におけるシンプレックス交叉の提案, 人工知能学会誌, Vol.16, No.1, 146/155, 2001
- [8] 佐藤 浩, 小野 功, 小林重信 : 遺伝的アルゴリズムにおける世代交代モデルの提案と評価, 人工知能学会誌, Vol.12, No.5, 734/744, 1997
- [9] 喜多 一, 小野 功, 小林重信 : 実数値 GA のための正規分布交叉に関する理論的考察, 計測自動制御学会論文集, Vol.35, No.11, 1333/1339, 1999
- [10] J.Kennedy and R.Eberhart : Particle Swarm Optimization, Proceedings of the 1995 IEEE International Conference on Neural Networks, 1942/1948, 1995
- [11] R.Storn and K.Price : Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces, Technical report, International Computer Science Institute, Berkley, 1995
- [12] J.Kennedy and R.Eberhart : Swarm Intelligence, Morgan Kaufman, San Francisco, 2001
- [13] Y.Shi and R.Eberhart : Empirical Study of Particle Swarm Optimization, Proceedings of the 1999 Congress on Evolutionary Computation, 1945/1950, 1999
- [14] Y.Shi, R.Eberhart : Parameter selection in particle swarm optimization, The 7th Annual Conference on Evolutionary Programming, 1998
- [15] 伊藤 稔, 田中雅博 : Particle Swarm Optimization の解探索性能に関する基礎的検討, 第 32 回知能システムシンポジウム資料, 217/220, 2005
- [16] C.W.Reynolds: Flocks, herds and schools: a distributed behavioral model, Computer Graphics, 21(4), 25/34, 1987
- [17] D.Whitley, K.Mathias, S.Rana, and J.Dzubera : Building better test functions, Proceedings of the 6th International Conference on Genetic Algorithms, 239/246, 1995
- [18] J.Digalakis, K.Margaritis : An Experimental study of benchmarking functions for Genetic Algorithms, In Proceedings of IEEE Conference on Transactions, Systems, Man and Cybernetics, 3810/3815, 2000

A comparative study of particle swarm optimization, differential evolution, real-coded genetic algorithm on function optimization

Minoru ITO Institute of Intelligent Information and Technology
Masahiro TANAKA Faculty of Science and Engineering

(Received April 15, 2005)

Abstract:

In this paper, we evaluate the performance of Differential Evolution (DE), Particle Swarm Optimization (PSO), and Real-Coded Genetic Algorithm (Real-Coded GA) on a function optimization problem. The comparative study is performed on 4 benchmark problems. The results from our study show that DE outperforms the other methods.

Keywords: Real-Coded Genetic Algorithm, Particle Swarm Optimization, Differential Evolution, Function Optimization