

勾配情報にもとづく局所探索を組み込んだ

ハイブリッド粒子群最適化

富山県立大学工学部電子情報工学科
1515050 山本聖也

指導教員：奥原浩之

1 はじめに

本研究の目標は PSO の応用手法である提案手法を実装しその有効性を示すとともに、提案手法を IoT センサの技術と結び付け、データの意思決定に用いることである。

今回はその前段階として、定式化したプログラムを実装し、制約条件付の PSO のモデルの有効性を示す。

2 PSO の概要

2.1 PSO アルゴリズム

PSO は群をなして移動する生物の行動を模範したアルゴリズムである(図 1)。群をなす生物をモデル化し、粒子は最適化問題における候補解を示している。PSO は群の中の粒子がもつ最良の情報 (pbest) とその集団の最適値 (gbest) から過去の探索を考慮し、さらにその集団の各粒子の位置および速度を更新することによって計算される。(1),(2) 式にそれを示す。 x_d^k は位置、 v_d^k は速度、 w は運動量、 x_{db}^k は p-best、 x_{gb}^k は g-best、 r_1, r_2 はそれぞれ 0 から 1 の乱数、 c_1, c_2 は任意で与えるパラメータである。

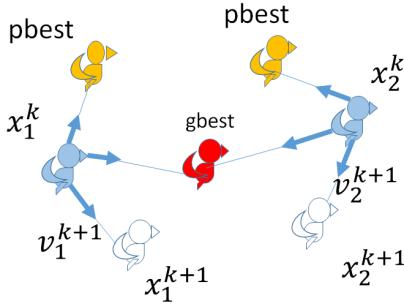


図 1 PSO の探索模式図

$$x_d^{k+1} = x_d^k + v_d^{k+1} \quad (1)$$

$$v_d^{k+1} = wv_d^k + c_1r_1(x_{db}^k - x_d^k) + c_2r_2(x_{gb}^k - x_d^k) \quad (2)$$

ここで、PSO の探索模式図及び速度と位置の更新式より、pbest に向かう $c_1r_1(x_{db}^k - x_d^k)$ 、gbest に向かう $c_2r_2(x_{gb}^k - x_d^k)$ 、これまでの進行方向へ向かう wv_d^k の 3 つのベクトルを合成して速度ベクトル v_i^{k+1} を決定し、それを元に次に移動する位置 x_d^{k+1} 決定する。

PSO の探索式はランダム要素を含み、同時に最良解情報である pbest と gbest が探索に伴い変化するという時変性を有している [3]。このままの形では理論解析が困難であるので、一つの Particle に着目し、一次元の位置 x と速度 v について考え、さらに pbest と gbest を一つの点に縮約した簡略モデルが提案されている [2]。この簡略モデルは、確定的な線形時不変システムとして表現されており、その安定性を示す。

Particlei に注目すると速度ベクトル v^{k+1} は以下の式のように変形できる。ステップ幅 ϕ は二つの一様乱数を足し合わせたものであり、最小値 0、最大値 $C_1 + C_2$ 、平均 $\frac{C_1 + C_2}{2}$ の分布に従う。

$$v_d^{k+1} = wv^k + \phi(P - x^k) \quad (3)$$

$$P = \frac{\phi_1 pbest^k + \phi_2 gbest^k}{\phi_1 + \phi_2} \quad (4)$$

ここで、 $\phi = \phi_1 + \phi_2$ 、 $\phi_1 = C_1 \text{rand}$ 、 $\phi_2 = C_2 \text{rand}$
さらに $y^k = p - x^k$ とおくと以下のように表せる。

$$\begin{bmatrix} v^{k+1} \\ y^{k+1} \end{bmatrix} = \begin{bmatrix} w & \phi \\ -w & 1 - \phi \end{bmatrix} \begin{bmatrix} v^k \\ y^k \end{bmatrix} \quad (5)$$

また $\phi = (C_1 + C_2)/2$ と見なすと固有値 λ は、

$$\lambda = \frac{w + 1 - \phi \pm \sqrt{(w + 1 - \phi)^2 - 4w}}{2} \quad (6)$$

よって λ が 1 を境にシステムの特性が安定・不安定(収束・発散)に変化することが分かる。

2.2 連続時間 PSO アルゴリズム

連続時間 PSO アルゴリズムについて述べる。

PSO の更新式を力学系モデルとみなし、その連続化を試みると、

$$\frac{dx^p(t)}{dt} = c \int_0^t e^{-a(t-\tau)} [F^p(x^p(\tau), \tau) + C(x^p(\tau), \tau)] d\tau \quad (7)$$

$$\frac{d^2x^p(t)}{dt^2} + a \frac{dx^p(t)}{dt} = c [F^p(x^p(t), t) + C(x^p(t), t)] \quad (8)$$

またそれぞれの関数 F^p, C は以下のようになる。

$$F^p(x^p, t) = c_1(x^p(T^p(t) - x^p)) \quad (9)$$

$$C^p(x^p, t) = c_2(x^{Q(t)}(T^o(t) - x^p)) \quad (10)$$

また、2 階微分方程式で表される連続時間系モデルの状態変数表現を、 $u^p(t) = x^p(t)$ 、 $v^p(t) = du^p(t)/dt + au^p(t)$ とおいて導入すると、離散時間系に対応した連続系の内部状態表現モデル、

$$du^p(t)/dt = -au^p(t) + v^p(t) \quad (11)$$

$$dv^p(t)/dt = c [F^p(u^p(t), t) + C(u^p(t), t)] \quad (12)$$

を得ることができる。

3 ハイブリッド PSO の提案

本節では提案手法であるハイブリッド PSO について解説する。PSO の応用法である連続時間 PSO アルゴリズムの応用法であり、勾配情報を加えることにより、精密な探索を行うことを狙いとしている。

(7),(8) 式に勾配情報を加えるとそのモデルは、

$$\frac{dx^p(t)}{dt} = c \int_0^t e^{-a(t-\tau)} [F^p(x^p(\tau), \tau) + C(x^p(\tau), \tau) - \nabla E(x^p(\tau), \tau)] \tau \quad (13)$$

$$\frac{d^2x^p(t)}{dt^2} + a \frac{dx^p(t)}{dt} = c [F^p(x^p(t), t) + C(x^p(t), t) - \nabla E(x^p(t), t)] \quad (14)$$

またそれぞれの関数は以下のようになる。

$$F^p(x^p, t) = c_1(x^p(T^p(t) - x^p)) \quad (15)$$

$$C^p(x^p, t) = c_2(x^{Q(t)}(T^o(t) - x^p)) \quad (16)$$

$$\nabla E(x^p, t) = c_3 \frac{\partial E(x^p, t)}{\partial x^p} \quad (17)$$

解説したままのモデルでは無制約なので、制約条件に対応したモデルである上下制約連続時間 PSO モデルを以下の式に示す。上下制約付最適化問題

$$\min E(x) \quad (18)$$

$$\text{subj.} \text{top}_i \leq x_i \leq q_i, i = 1, \dots, n \quad (19)$$

を直接解くために、この上下制約領域内に問題 (18),(19) の変数を変換して無制約化した新たな変数空間に無制約 PSO モデルを適用した「変数変換モデル」を導入する。非線形変数変換モデルを作成するために、

$$x_i = f_i(y_i) = \frac{q_i + p_i \exp(-y_i)}{1 + \exp(-y_i)} \quad (20)$$

とおく。この変換式を制約条件付き問題に代入して変数 x を消去すると、

$$\min E(f(y)) \quad (21)$$

を得ることができる。よって (14) 式に対応させると、

$$\frac{d^2 y^p(t)}{dt^2} + a \frac{dy^p(t)}{dt} = c[F^p(y^p(t), t) + C(y^p(t), t) - \nabla E(y^p(t), t)] \quad (22)$$

またそれぞれの関数は以下のようになる。

$$F^p(y^p, t) = c_1(y^p(T^p(t) - y^p)) \quad (23)$$

$$C^p(y^p, t) = c_2(y^{Q(t)}(T^o(t) - y^p)) \quad (24)$$

$$\nabla E(y^p, t) = c_3 \frac{\partial E(y^p, t)}{\partial y^p} \quad (25)$$

次にプログラムへの実装を考えた時に、連続式のままではプログラムに実装することが難しいので、オイラー法を用いて連続式を離散化し非線形変数変換モデルの離散化 PSO を作成。それぞれに対応する式を以下に示す。

$$u^p(k+1) = (1 - a\Delta T)u^p(k) + \Delta T v^p(k) \quad (26)$$

$$v^p(k+1) = v^p(k) + c\Delta T[F^p(u^p(k), k) + C(u^p(k), k) - \nabla E(u^p(k), k)] \quad (27)$$

$$F^p(k, k) = c_1(u^p(l^p(k)) - u^p(k)) \quad (28)$$

$$C^p(k, k) = c_2(u^{Q(k)}(l^o(k)) - u^p(k)) \quad (29)$$

$$\nabla E(k, k) = c_3 \frac{\partial E(k, t)}{\partial k} \quad (30)$$

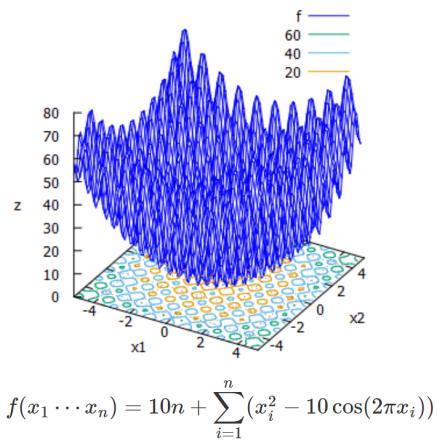
$$l^p(k) = \operatorname{argmin}(E(x^p(l)) | l = 0, \dots, k) \quad (31)$$

$$(Q(k), l^o(k)) = \operatorname{argmin}(E(x^q(l)) | q = 1, 2, \dots, P, l = 0, 1, \dots, k) \quad (32)$$

$$x^p_i(k) = f_i(u^p_i(k)) = \frac{q_i + p_i \exp(-u^p_i(k))}{1 + \exp(-u^p_i(k))}, i = i, \dots, n \quad (33)$$

4 数値実験ならびに考察

提案手法の有効性を示すため評価関数として Rastrigin function(図 2) と Griewank function(図 3) を用いて数値実験を行う。これらの評価関数は多峰性ならびに、非常に多くの局所解を持つ。



$$f(x_1 \cdots x_n) = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i))$$

$$-5.12 \leq x_i \leq 5.12$$

$$f_{min}(0, \dots, 0) = 0$$

図 2 Rastrigin function

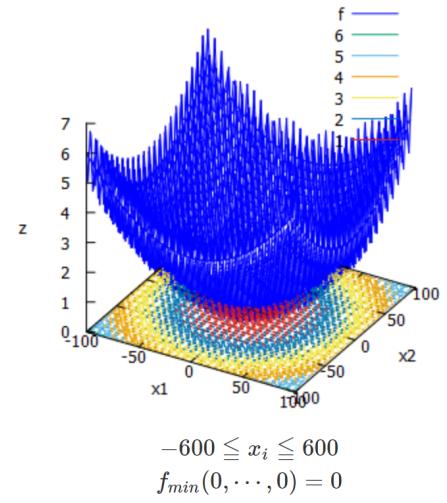


図 3 Griewank function

以下に従来法と提案手法を比較した物を示す(図 4,5)。各パラメータの値は N (粒子の数) = 10, $C = 1.0, c_1 = 1.4, c_2 = 0.1, c_3 = 0.1, a = 1.0, \Delta T = 0.9$ というように与えた。また試行回数は 100 とした。

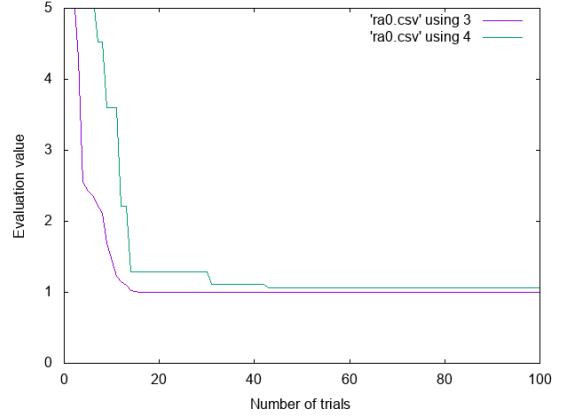


図 4 Rastrigin function の実験結果

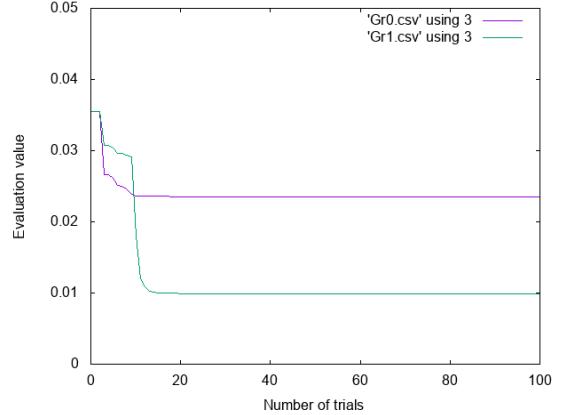
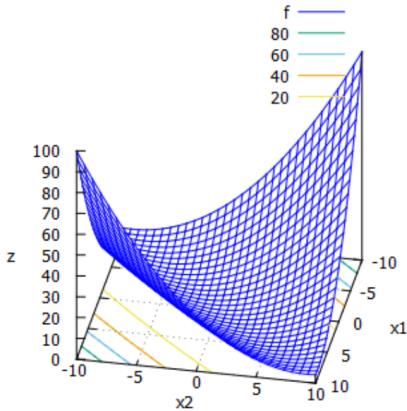


図 5 Griewank function の実験結果

加えて評価関数 Matyas function(図 6) と Booth function(図 7) を用いて数値実験を行う。この関数は单峰性関数である。上記の実験と同じ条件で行いその結果を示す(図 8, 9)。また Booth function の実行時に粒子の数を $N=100$ に変更した場合の結果(図 10)も示す。

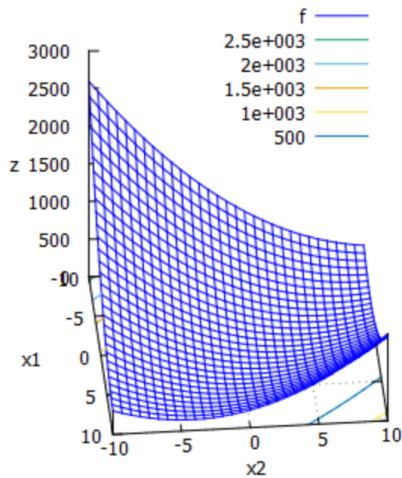


$$f(x_1, x_2) = 0.26(x_1^2 + x_2^2) - 0.48x_1x_2$$

$$-10 \leq x_1, x_2 \leq 10$$

$$f_{min}(0,0) = 0$$

図 6 Matyas function



$$-15 \leq x_1 \leq -5$$

$$-3 \leq x_2 \leq 3$$

$$f_{min}(-10, 1) = 0$$

図 7 Booth function

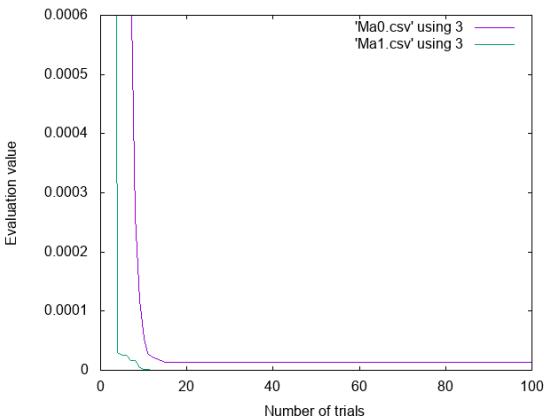


図 8 Matyas function の実験結果

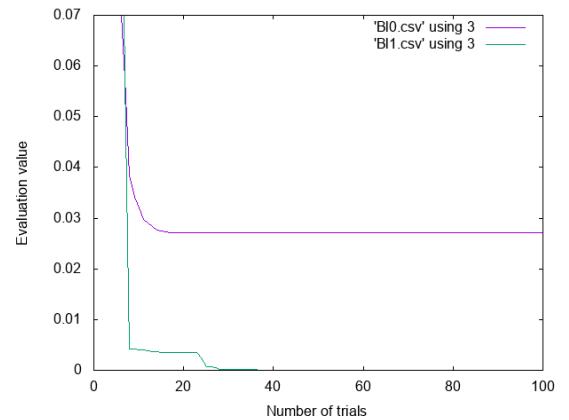


図 9 Booth function の実験結果

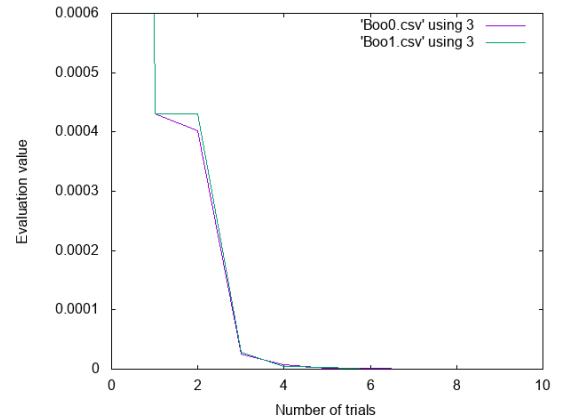


図 10 Booth function (N=100) の実験結果

提案手法は多峰性をもつ関数の場合においてあまり良い結果が得られなかった。これは提案手法が勾配法を用いているため多峰性をもつ関数には適していないと思われる。一方、単峰性をもつ関数の場合においては従来法よりも良い結果を得ることができた。

また発生させる粒子の数を 100 から 10 を比較した場合、従来法では最適値を導出できなかったが、提案手法では最適値を導出することができた。これにより計算コストの削減が可能となった。

5 おわりに

今回は定式化したプログラムを実装した。その結果提案手法は粒子の少ない状態かつ単峰性の関数の場合において、従来法より良い結果を示すことができ、計算コストを軽減することができた。今後は提案手法を IoT センサの技術と結び付けることを目標としている。

参考文献

- [1] J. Kennedy, R. C. Eberhart, "Particle swarm optimization", *IEEE Conf. On Neural Networks, IV, Piscataway, NJ*, pp. 1942-1948 (1995).
- [2] J. Kennedy, R.C. Eberhart, Y. Shi, "Swarm intelligence", Morgan Kaumann Publishers, San Fran-cisco, CA, pp. 1942-1948, 2001.
- [3] 石亀敦司, 安田恵一郎, "群れの知能: Particle Swarm Optimization", 知能と情報 (日本知能情報ファジィ学会誌) , Vol. 20, No. 6 , pp. 829-839 (2008) .
- [4] H. M. Emara and H. A. Abdel Fattah, "Continuous swarm optimization technique with stability analysis", *Proceeding of the 2004 American Control Conference*, 2811-2817 (2004).
- [5] Ryuzaburo SUGINO, Anan National College of Technology: "Numerical Performance of PSO Algorithm Using by Gradient Method",
- [6] M. Jiang, Y. P. Luo and S. Y. Yang, "Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm", *Information Processing Letters* vol. 102, No. 1, pp. 8-16 (2007).