

# 近傍関係を考慮するネットワーク構造型粒子群最適化法

## Network-Structured Particle Swarm Optimizer

松下春奈<sup>†</sup>

西尾芳文<sup>†</sup>

<sup>†</sup>徳島大学 工学部 電気電子工学科

Haruna MATSUSHITA<sup>†</sup>

Yoshifumi NISHIO<sup>†</sup>

<sup>†</sup>Department of Electrical and Electronic Engineering, Tokushima University

### 1 はじめに

粒子群最適化 (Particle Swarm Optimization: PSO) は、鳥の群れの動きをモデル化した進化アルゴリズムの一種である。概念が単純であり実現が容易な上、収束が早いという特徴を持つことから、近年、異なる分野の様々な応用に用いられている。しかしながら、パラメータに大きく依存することや、多数の局所解を持つような複雑な問題を解く場合においての早すぎる収束などが、PSOの問題点として挙げられている。

PSOは複数の解(粒子)群で構成され、各粒子は位置と速度の情報を持っている。通常PSOは、これらの情報を群内で交換しながら良好な位置を探す。粒子間に特別な繋がりはない。つまり、各粒子の情報は固体が最も良かった位置と群全体で最も良かった位置に従って更新するが、どちらを重要視するかの割合は毎更新ごとにランダムで決められる。一方、実社会においては、上下関係、信頼関係、親子関係などの様々な人間関係が存在している。

PSOにおける様々なトポロジー近傍については、以前から議論されてきており[2]–[6]、リング型近傍、ノイマン近傍、ランダム近傍などのあらゆるトポロジー近傍が適用されてきた。しかしながら、粒子情報を更新する際の近傍を考慮するための新しい項、及びそれに伴うパラメータの増加などが懸念されている。また、“近傍”の概念についてはいくつかの考え方があり、複数の研究においては、似た評価値を持つ粒子や空間で近くに位置する粒子が近傍粒子として定義されてきた。しかし、粒子間に結合関係がない場合、更新の度に近傍粒子を見つけ出す必要があり、計算量が増大するといった問題がある。

本研究では、近傍関係を考慮するネットワーク構造型PSO(NS-PSO)を提案する。NS-PSOの全粒子

は2次元ネットワーク上での近傍に従い結合されており、直接結合された粒子間、つまりネットワーク上の近傍粒子間で、それらが最も良かった位置を情報交換する。また、ステップ毎にその時最も良い位置を持つ粒子を勝者とし、ネットワーク上の勝者との近傍距離に従い各粒子は更新される。

提案手法の有効性を確認するために、単峰性と多峰性それぞれ4つの評価関数に対してシミュレーションを行う。従来のPSOと比較することにより、正確性、ロバスト性、パラメータ依存性の観点から評価し、NS-PSOが探索能力を効果的に高めることを確認する。

### 2 Particle Swarm Optimization (PSO)

PSOには粒子と呼ばれる複数の解が存在しており、各更新では、固体の最良の位置と群全体での最良の位置に向かうように動く。各粒子は、位置  $\mathbf{X}_i = (x_{i1}, \dots, x_{id}, \dots, x_{iD})$  と速度  $\mathbf{V}_i = (v_{i1}, \dots, v_{id}, \dots, v_{iD})$  の情報を持っている。ここで、 $(d = 1, 2, \dots, D)$ ,  $(i = 1, 2, \dots, M)$ ,  $x_{id} \in [x_{\min}, x_{\max}]$  とする。

**(PSO1)** (初期化)  $t = 0$  とし、粒子  $i$  の位置  $\mathbf{X}_i$  をランダムに、速度  $\mathbf{V}_i$  を0に初期化する。 $\mathbf{P}_i = (p_{i1}, p_{i2}, \dots, p_{iD})$  を現在位置  $\mathbf{X}_i$  に初期化する。各粒子  $i$  に対し目的関数  $f(\mathbf{X}_i)$  を計算し、 $\mathbf{P}_g$  を群全体で最も良い評価値を持つ粒子の位置に初期化する。

**(PSO2)** 適応度  $f(\mathbf{X}_i)$  を計算する。各粒子に対して、 $f(\mathbf{X}_i) < f(\mathbf{P}_i)$  であれば、自己最高の位置 ( $pbest$ ) を置き換える ( $\mathbf{P}_i = \mathbf{X}_i$ )。  $\mathbf{P}_g$  を群全体において今までで最良の位置 ( $gbest$ ) とし、必要があれば更新する。

**(PSO3)**  $pbest$  と  $gbest$  に応じて、粒子  $i$  の  $\mathbf{V}_i$  と  $\mathbf{X}_i$

を更新する.

$$v_{id}(t+1) = wv_{id}(t) + c_1 \text{rand}(\cdot) (p_{id} - x_{id}(t)) + c_2 \text{Rand}(\cdot) (p_g - x_{id}(t)) \quad (1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1)$$

ここで,  $w$  は慣性重みであり, 以前の速度をどれだけ保持するかを決める.  $c_1, c_2$  は正の加速度係数であり, 通常  $c_1 = c_2$  とする.  $\text{rand}(\cdot), \text{Rand}(\cdot)$  は  $U(0, 1)$  における一様ランダム値である.

(PSO4)  $t = t + 1$  とし, (PSO2) に戻る.

### 3 近傍関係を考慮するネットワーク構造型 PSO (NS-PSO)

NS-PSO は通常の PSO と同様にシンプルなアルゴリズムで成り立っている. NS-PSO の最も重要な特徴は, 全粒子が図 1 のように二次元グリッド上で編成されており, 粒子は隣接した粒子と結合されていることから, 近傍関係を持っていることである. また, 粒子は直接結合された近傍粒子間で, ローカルな最良の位置を交換する. ここで, 本研究における“近傍”の定義について記述する. 複数の研究においては, 似た評価値を持つ粒子や空間で近くに位置する粒子が, 近傍粒子として定義されてきた. しかし本研究では, たとえ評価値や位置が離れていたとしても, ネットワーク上で直接結合されている粒子を“近傍”とする.

(NS-PSO1) (初期化) ステップを  $t = 0$  とし, (PSO1) に従い粒子情報を初期化する.  $gbest$  の粒子  $g$  を勝者粒子  $c$  とする.  $\mathbf{L}_i = (l_{i1}, l_{i2}, \dots, l_{iD})$  を, 直接結合された粒子間, つまり近傍粒子間で, 最良評価値を持つ粒子の位置に初期化する.

(NS-PSO2) 適応度  $f(\mathbf{X}_i)$  を計算し, 現ステップ  $t$  において最良の評価値を持つ粒子を勝者  $c$  とする.

$$c = \arg \min_i \{f(\mathbf{X}_i(t))\} \quad (2)$$

各粒子  $i$  に対して,  $f(\mathbf{X}_i) < f(\mathbf{P}_i)$  であれば  $pbest$  を置き換える ( $\mathbf{P}_i = \mathbf{X}_i$ ).

もし  $f(\mathbf{X}_c) < f(\mathbf{P}_g)$  であれば,  $gbest$  を置き換える ( $\mathbf{P}_g = \mathbf{X}_c$ ). ここで,  $\mathbf{X}_c = (x_{c1}, x_{c2}, \dots, x_{cD})$  は勝者  $c$  の位置である.

(NS-PSO3)  $\mathbf{L}_i$  を, 粒子  $i$  とネットワーク上で直接結合された近傍粒子内の今までで最良の位置 ( $lbest$ ) とし, 必要であれば更新する (図 1 参照).

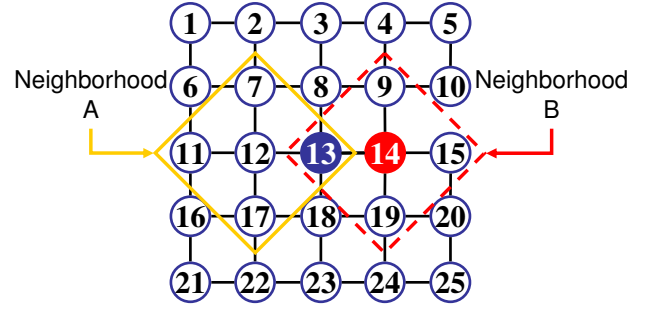


図 1: ネットワーク構造と近傍関係. 粒子 12 の近傍は粒子 7, 11, 12, 13, 17 を含む Neighborhood A である. 粒子 14 の近傍は粒子 9, 13, 14, 15, 19 を含む Neighborhood B である. もし  $f(\mathbf{P}_{13})$  が Neighborhood A 内で最小の値なら, 粒子 12 の  $lbest$  は  $\mathbf{L}_{12} = \mathbf{P}_{13}$  となる. もし  $f(\mathbf{P}_{14})$  が Neighborhood B 内で最小の値, 粒子 14 の  $lbest$  は  $\mathbf{L}_{14} = \mathbf{P}_{14}$  となる.

(NS-PSO4) 各粒子  $i$  の  $\mathbf{V}_i$  と  $\mathbf{X}_i$  を,  $lbest$ , 勝者の位置  $\mathbf{X}_c$  及び勝者  $c$  とのネットワーク上の近傍距離に従い更新する.

$$v_{id}(t+1) = wv_{id}(t) + c_1 \text{rand}(\cdot) (l_{id} - x_{id}(t)) + c_2 h_{c,i} (x_{cd} - x_{id}(t)) \quad (3)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1)$$

ここで,  $h_{c,i}$  は近傍関数と呼ばれるガウス関数である.

$$h_{c,i} = \exp\left(-\frac{\|\mathbf{r}_i - \mathbf{r}_c\|^2}{2\sigma^2}\right) \quad (4)$$

$\|\mathbf{r}_i - \mathbf{r}_c\|$  は勝者  $c$  と粒子  $i$  のネットワーク上の距離,  $\sigma$  は近傍関数の幅に対応する固定パラメータである.  $\sigma$  が大きいと粒子群が空間全体に広がる力を強め,  $\sigma$  が小さいと粒子群が勝者に近づく収束力を強める.

(NS-PSO5)  $t = t + 1$  とし, (NS-PSO2) に戻る.

## 4 シミュレーション結果

NS-PSO の性能を評価するために, 従来型 PSO との比較実験を行った. 用いたのは表 1 に示される 8 つのベンチマーク最適化問題で,  $f_1, f_2, f_3$  及び  $f_4$  は単峰性関数 (図 2(a)–(d) 参照),  $f_5, f_6, f_7$  及び  $f_8$  は多くの局所解を持つ多峰性関数 (図 2(e)–(h) 参照) である. 図 2(h) において, Griewank 関数は単峰性関数である Sphere 関数 (図 2(a)) と類似しているように見えるが, 中心付近を拡大すると多くの小さな山と谷が存在する. Rosenbrock 関数  $f_2$  の最適

表 1: 8 つの最適化関数.

関数名	関数式	初期空間
Sphere;	$f_1(x) = \sum_{d=1}^{D-1} x_d^2,$	$x \in [-2.048, 2.047]^D$
Rosenbrock;	$f_2(x) = \sum_{d=1}^{D-1} \left( 100(x_d^2 - x_{d+1})^2 + (1 - x_d)^2 \right),$	$x \in [-2.048, 2.047]^D$
3 <sup>rd</sup> De Jong;	$f_3(x) = \sum_{d=1}^D  x_d ,$	$x \in [-2.048, 2.047]^D$
4 <sup>th</sup> De Jong;	$f_4(x) = \sum_{d=1}^D dx_d^4,$	$x \in [-1.28, 1.27]^D$
Rastrigin;	$f_5(x) = \sum_{d=1}^D (x_d^2 - 10 \cos(2\pi x_d) + 10),$	$x \in [-5.12, 5.12]^D$
Ackley;	$f_6(x) = \sum_{d=1}^{D-1} \left( 20 + e - 20e^{-0.2\sqrt{0.5(x_d^2 + x_{d+1}^2)}} - e^{0.5(\cos(2\pi x_d) + \cos(2\pi x_{d+1}))} \right),$	$x \in [-30, 30]^D$
Stretched V sine wave;	$f_7(x) = \sum_{d=1}^{D-1} (x_d^2 + x_{d+1}^2)^{0.25} (1 + \sin^2(50(x_d^2 + x_{d+1}^2)^{0.1})),$	$x \in [-10, 10]^D$
Griewank;	$f_8(x) = \sum_{d=1}^D \frac{x_d^2}{4000} - \prod_{d=1}^D \cos\left(\frac{x_d}{\sqrt{d}}\right) + 1,$	$x \in [-600, 600]^D$

解  $x^*$  は  $[1, 1, \dots, 1]$  で、それ以外の関数の  $x^*$  は全て  $[0, 0, \dots, 0]$  である. 全関数において最適評価値  $f(x^*)$  は 0 である. 全関数は  $D$  個の変数を持っており、本研究では  $D = 50$  に設定する.

PSO の全粒子数  $M$  は 36 に、MS-PSO のネットワークサイズは  $6 \times 6$  に設定した. PSO および MS-PSO においてパラメータは  $w = 0.7$ ,  $c_1 = c_2 = 1.6$  とし、MS-PSO の近傍幅  $\sigma$  は 1.5 とする.

3000 generations を一回のシミュレーションとし、30 回試行したときの  $gbest$  の最小値、平均値及び標準偏差を表 2 に示す. これを見ると、全ベンチマーク関数において MS-PSO が良い値を得られていることがわかる. 特に  $f_1$ ,  $f_3$ ,  $f_4$  及び  $f_8$  に関しては、MS-PSO は PSO の結果を大きく改善しており、さらに、7 つの関数においてより良い最小値に到達できている.

### 5 パラメータ依存性

さらに、慣性重み  $w$  の性能評価への効果と、そのパラメータ鋭敏性について考察する.

図 4 は異なる  $w$  を用い、それぞれ 30 回試行した  $gbest$  の平均を示している. 固定パラメータ  $c_1 = c_2$

表 2: PSO と MS-PSO の比較結果.

$f$	Method	Mean	Minimum	Std
$f_1$	PSO	2.31e-18	1.25e-25	1.26e-17
	MS-PSO	<b>8.34e-26</b>	2.16e-30	2.74e-25
$f_2$	PSO	55.88	16.33	27.55
	MS-PSO	<b>41.60</b>	37.22	9.63
$f_3$	PSO	1.70e-06	1.15e-09	6.81e-06
	MS-PSO	<b>1.59e-07</b>	3.16e-12	7.46e-07
$f_4$	PSO	9.83e-36	1.34e-41	4.15e-35
	MS-PSO	<b>3.27e-42</b>	9.94e-48	1.13e-41
$f_5$	PSO	149.71	99.50	34.84
	MS-PSO	<b>87.69</b>	54.72	21.91
$f_6$	PSO	235.98	123.85	59.03
	MS-PSO	<b>180.79</b>	10.32	239.37
$f_7$	PSO	69.88	49.71	9.51
	MS-PSO	<b>37.08</b>	17.72	7.91
$f_8$	PSO	9.10e-02	0	0.2788
	MS-PSO	<b>2.61e-02</b>	0	4.11e-02

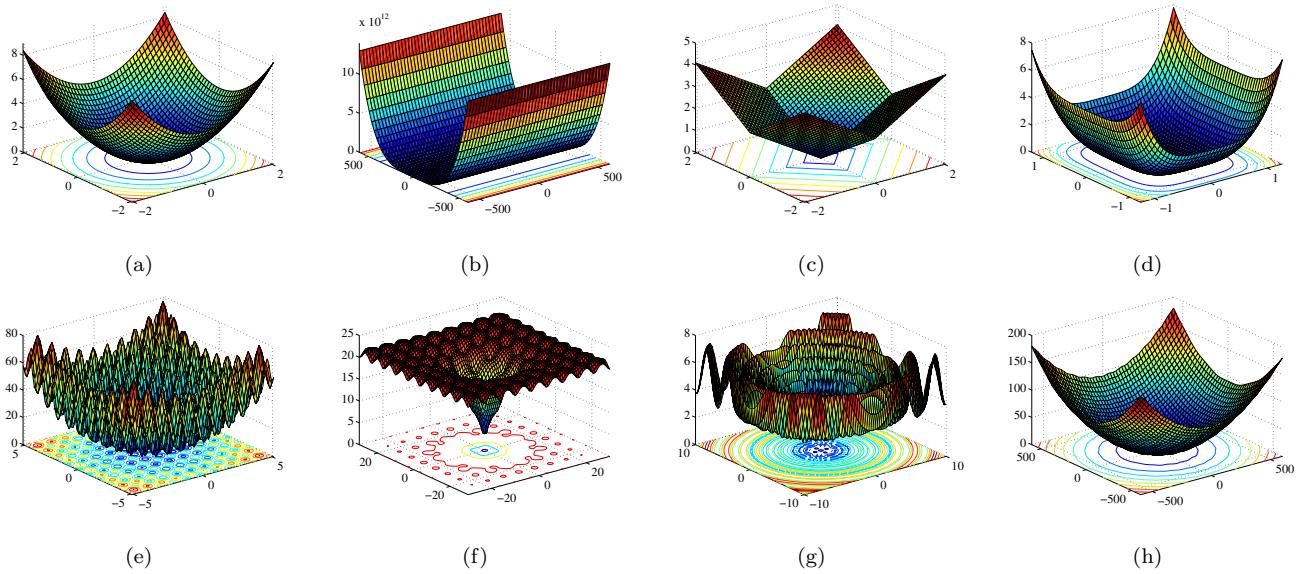


図 2: 8つのベンチマーク最適化関数. 1番目と2番目の変数値をそれぞれX軸, Y軸とし, Z軸は対応する関数値を示している. (a) Sphere. (b) Rosenbrock. (c) 3<sup>rd</sup> De Jong. (d) 4<sup>th</sup> De Jong. (e) Rastrigin. (f) Ackley. (g) Stretched V sine wave. (h) Griewank.

及び $\sigma$ は, 4章で用いたものと同じとした. この図から, たとえ慣性重み $w$ を変化させたとしても, MS-PSOはPSOよりも良い性能を示していることがわかる. さらに, たとえ $w$ の値を0.1増減させただけでも, PSOの性能は大きく悪くなってしまっており, 特に図4(a)–(d), (g)及び(h)においてその様子が顕著に見られる. 一方, MS-PSOは慣性重み $w$ が小さな値( $w \in [0.1, 0.6]$ )だとしてもPSOよりも良い値を得られており [7], 特に図4(e)–(h)の様な多峰性関数において安定した結果を示している. これらの結果から, MS-PSOはより高性能であり, そのパラメータ依存性は従来のPSOの比べてそれほど強くないことがわかる.

## 6 おわりに

本研究では, トポロジー近傍を持った新しい粒子群最適化法 (Particle Swarm Optimization: PSO) である, 近傍関係を考慮するネットワーク構造型 PSO (NS-PSO) を提案した. NS-PSOの全粒子は, 2次元格子ネットワークにおける近傍関係に従って隣接する粒子と結合されており, 直接結合されている粒子間で各自の最良の位置を情報交換する. また, ステップ毎にその時最も良い位置を持つ粒子を勝者粒子とし, 各粒子は勝者とのネットワーク上の近傍距離に従い更新される.

単峰性と多峰性それぞれ4つの評価関数に対してシミュレーションを行った. それらの結果より, PSOとNS-PSOの正確性, ロバスト性及びパラメータ依存性の観点から比較し, NS-PSOの総合的性能がPSOよりも有効でロバストであることを明らかにした.

今後の課題として, 様々なネットワークトポロジーを持ったNS-PSOの振る舞いを調査するとともに, ネットワーク構造と性能の関係性をさらに詳しく調査する必要がある.

## 参考文献

- [1] J. Kennedy and R. C. Eberhart, “Particle swarm optimization,” in *Proc. IEEE. Int. Conf. Neural Netw.*, pp. 1942–1948, 1995.
- [2] J. Kennedy, “Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance,” in *Proc. of Cong. on Evolut. Comput.*, pp. 1931–1938, 1999.
- [3] J. Kennedy and R. Mendes, “Population structure and particle swarm performance,” in *Proc. of Cong. on Evolut. Comput.*, pp. 1671–1676, 2002.
- [4] R. Mendes, J. Kennedy and J. Neves, “The Fully Informed Particle Swarm: Simpler,

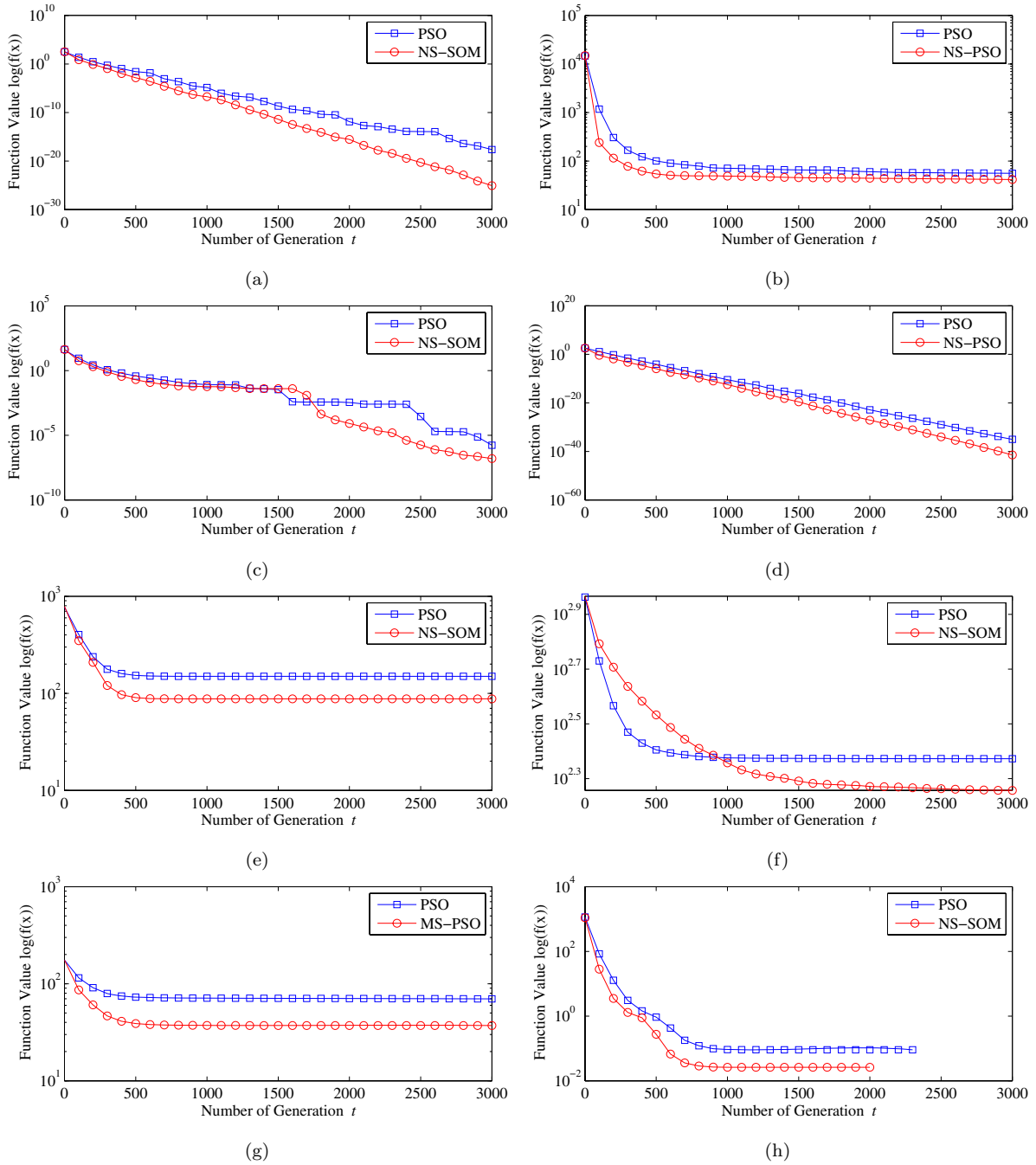


図 3: 毎 generation の  $gbest$  の平均値.  $D = 50$ . (a) Sphere. (b) Rosenbrock. (c) 3<sup>rd</sup> De Jong. (d) 4<sup>th</sup> De Jong. (e) Rastrigin. (f) Ackley. (g) Stretched V sine wave. (h) Griewank.

Maybe Better,” in *IEEE Trans. Evolut. Comput.*, vol. 8, no.3, pp. 204–210, June 2004.

[5] J. Lane, A. Engelbrecht and J. Gain, “Particle Swarm Optimization with Spatially Meaningful Neighbours” in *Proc. of IEEE Swarm Intelligence Symposium*, 2008.

[6] S. B. Akat and V. Gazi, “Particle Swarm Opti-

mization with Dynamic Neighborhood Topology: Three Neighborhood Strategies and Preliminary Results” in *Proc. of IEEE Swarm Intelligence Symposium*, 2008.

[7] X. Xie, W. Zang and Z. Yang, “A dissipative swarm optimization” in *Proc. of Cong. on Evolut. Comput.*, pp. 1456–1461, 2002.

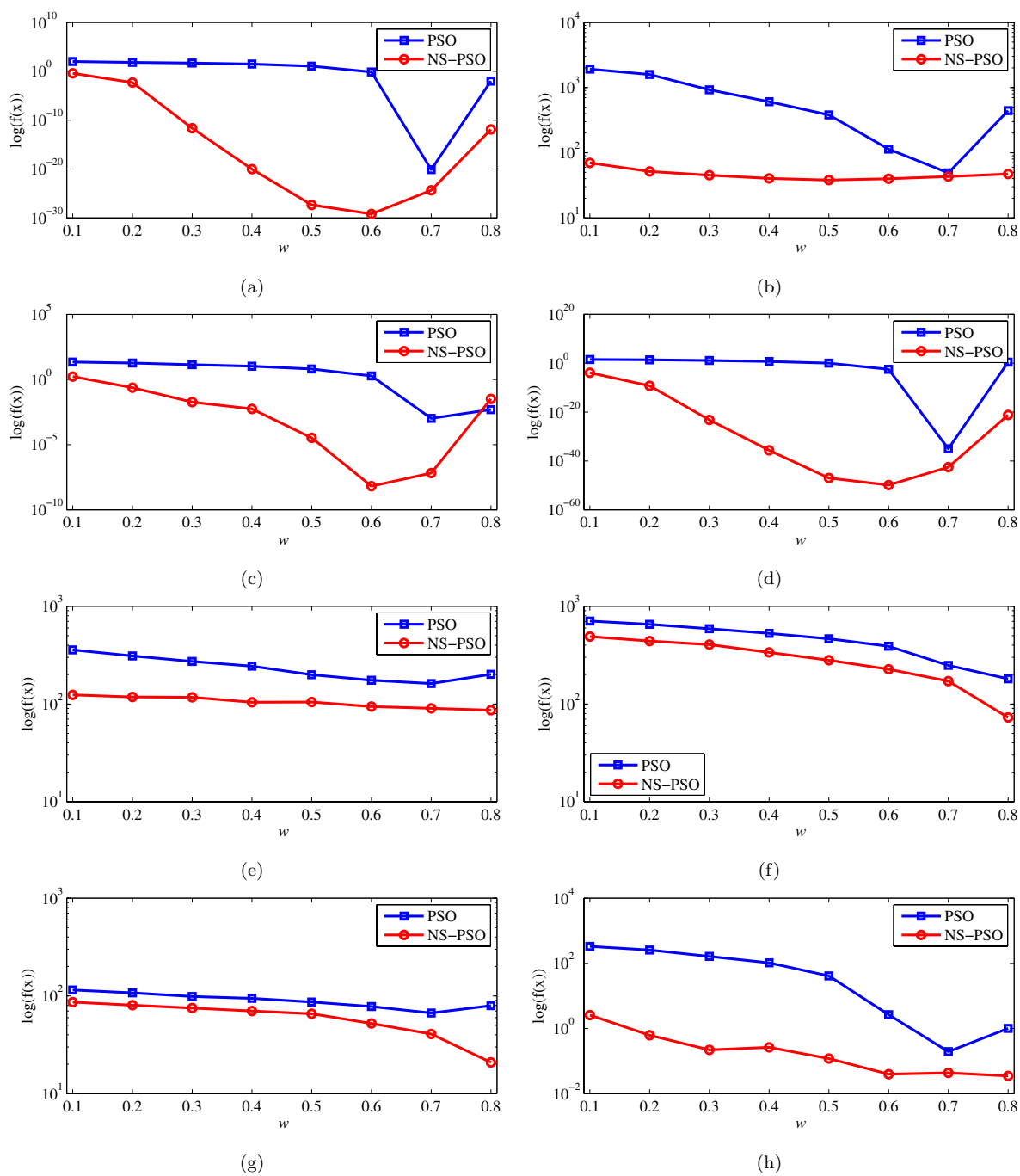


図 4: 異なる  $w$  の値に対する  $g_{best}$  の平均値.  $D = 50$ . 固定パラメータは  $c_1 = c_2 = 1.6$  及び  $\sigma = 1.5$  とする. (a) Sphere. (b) Rosenbrock. (c) 3<sup>rd</sup> De Jong. (d) 4<sup>th</sup> De Jong. (e) Rastrigin. (f) Ackley. (g) Stretched V sine wave. (h) Griewank.