

区域分割的自适应变异粒子群算法

陈侃松^{1,2}, 阮玉龙¹, 戴磊¹, 兰智高², 邵建设²

(1. 湖北大学计算机与信息工程学院物联网工程研究所, 湖北武汉 430062; 2. 黄冈师范学院电子信息学院, 湖北黄冈 438000)

摘要: 为了提高粒子群算法(PSO)的收敛性及多样性,提出一种基于区域分割的自适应变异粒子群算法(RS-VPSO). 算法采用区域分割的思想,利用粒子间信息交叉,使粒子搜索区间快速缩小;同时在迭代后期与自适应变异策略相结合,提高粒子跳出局部最优陷阱的能力和增强粒子多样性,达到寻优的目的. 将所提出的算法应用于8个测试函数,并与精英免疫克隆选择的协同进化粒子群等算法进行比较,结果表明,新算法在收敛速度、搜索精度及寻优效率等方面有较大提高.

关键词: 区域分割; 信息交叉; 自适应变异; 多样性

中图分类号: TP18 **文献标识码:** A **文章编号:** 0372-2112 (2017)08-1849-07

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2017.08.007

Regional-Segmentation Self-Adapting Variation Particle Swarm Optimization

CHEN Kan-song^{1,2}, RUAN Yu-long¹, DAI Lei¹, LAN Zhi-gao², SHAO Jian-she²

(1. Institute of Internet of Things, School of Computer Science and Information Engineering, Hubei University, Wuhan, Hubei 430062, China; 2. School of Electronic Information, Huanggang Normal University, Huanggang, Hubei 438000, China)

Abstract: To improve convergence and diversity of particle swarm optimization(PSO), an improved PSO which called regional-segmentation self-adapting variation particle swarm optimization (RSVPSO) algorithm is introduced. Regional-segmentation is adopted in the algorithm, using information cross between particles, narrow search region quickly; combining with self-adapting variation strategy in late iterations at the same time, improved capacity of jumping out local optimum trap and enhanced the diversity of particles, reach the goal of optimization. The proposed algorithm is applied to eight test functions and compared with the elite immune clonal selection co-evolutionary particle swarm optimization and so on. The results show that the proposed algorithm has considerable improvement in the convergence speed, search accuracy, optimum efficiency and so on.

Key words: regional-segmentation; information cross; self-adapting variation; diversity

1 引言

PSO算法是由Kennedy^[1]等人于1995年提出的,是一种典型的群体智能寻优算法.因其具有概念简单、参数少和收敛速度相对较快的优点,已被广泛应用于工程实践和函数优化等领域.但其在进化后期存在收敛速度慢以及易陷入局部最优而导致早熟的不足,特别是对于高维多峰等复杂函数的寻优.为了改进这些不足,国内外研究学者从不同角度与机理对算法进行了改进.主要可分为以下两类:

第1类是对算法本身的改进.如Shi^[2]等人提出基于线性惯性权重的改进粒子群算法,在速度项前引入线性递减惯性权重,以平衡算法的全局探测和局部搜

索能力. Lovbjerg^[3]提出多子群策略,将群体划分成多个子群分别进行优化搜索,避免所有粒子收敛于同一局部极值点.文献^[4]将动态语境协作的思想引入粒子群算法中,有效地提高了算法收敛精度. Li^[5]用量子行为来引导粒子运动,使算法开销得以大大减少.

第2类是与其他算法的结合. Angeline^[6]将遗传算法的选择操作引入到PSO中,有效提高了算法的收敛性. Xin^[7]等人将差分进化算法与PSO结合,更新两种算法被选中的概率,提高了算法的执行效率.周剑波^[8]等将虚拟力算法引入PSO中,加快了粒子运动,提高了算法收敛速度.

2 标准 PSO 算法

标准 PSO 算法数学描述为:在一个 D 维搜索区域

(解空间)内存在 N 个粒子,每个粒子被视为 D 维搜索空间的一个搜索个体,描述粒子状态的参数有两个,分别为速度与位置.其中速度可表示为: v_{ij} ,其中 i 和 j 分别表示粒子 i 的速度在 j 维上的分量;位置表示为: $X_i = (x_{i1}, x_{i2}, \dots, x_{ij}, \dots, x_{iD})$,粒子根据自身的经验和群体经验来指导自己的“飞行”,个体历史最优表示为 $pbest$,群体历史最优记为 $gbest$. 粒子速度与位置分别按下式更新:

$$v_{ij}(t+1) = \omega v_{ij}(t) + c_1 r_1 (pbest_{ij}(t) - x_{ij}(t)) + c_2 r_2 (gbest_g(t) - x_{ij}(t)) \quad (1)$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \quad (2)$$

其中, t 是迭代次数, c_1 为自身加速常数, c_2 为全局加速常数,均在 $[0, 2]$ 间; r_1 和 r_2 是 $[0, 1]$ 之间的随机数, ω 为惯性权重,一般取值在 $[0.1, 0.9]$ 之间. 文献[9]论证了算法满足收敛性的条件为:

$$\begin{cases} 1 - \omega > 0 \\ 2\omega + 2 > c_1 r_1 + c_2 r_2 \end{cases} \quad (3)$$

3 区域分割的自适应变异粒子群算法

3.1 惯性权重的设计

在每次迭代中,定义每个粒子历史最优位置与全局最优位置的欧氏距离为 L_{ij} ,其中 i 和 j 分别为第 i 次迭代中的第 j 个粒子,则 L_{ij} 的表达式为:

$$L_{ij} = \sqrt{\sum_{k=1}^D (g_k - p_{jk})^2} \quad (4)$$

其中, D 表示搜索空间维度, g_k 和 p_{jk} 分别表示全局最优位置在 k 维上的分量和第 j 个粒子历史最优位置在 k 维上的分量. 改进惯性权重表示如下:

$$\omega_{ij} = (\omega_{\max} - \omega_{\min}) \cdot \frac{L_i}{L_{\max}} \cdot e^{\frac{L_i - L_{\min}}{L_{\max}}} + \omega_{\min} \quad (5)$$

其中, ω_{\max} 和 ω_{\min} 为 0.9 和 0.4, L_{\max} 表示搜索区间的所有粒子欧氏距离的最大值. 显而易见: $-1 \leq \frac{L_{ij} - L_{\max}}{L_{\max}} \leq 0$,

从而推得 $\frac{1}{e} \leq e^{\frac{L_i - L_{\min}}{L_{\max}}} \leq 1$,也即保证了 ω_{ij} 在 $(0.4, 0.9)$ 理想区间,同时也将 Shi 线性递减权重优化为指数的非线性递减权重,使粒子速度变化呈现出前期快后期慢的特征,更好地平衡全局搜索与局部挖掘;更为重要的是在同一次迭代中,不同的粒子根据其其与全局最优位置的欧氏距离不同而被赋予不同的惯性权重,从而在大大提高收敛速度的同时也很好地兼顾了发现新的最优解的概率,即充分利用粒子的差异性提高搜索效率.

3.2 区域分割

区域分割思想是通过不断分割搜索区域,以达到快速缩小收敛区域的目的. 同时用随机选择来提高粒子的多样性,具体操作为以下 5 步:

(1) 在整个搜索区间内,按粒子到全局最优位置的

欧氏距离进行排序,以某一欧氏距离 ($e \cdot f_{ave}$, 其中 f_{ave} 为粒子适应值的平均值) 为界,将粒子区域分成两部分:界限内区域与界限外区域. 界限内区域为一群较优粒子,界限外为一群较劣粒子;

(2) 从较优粒子中随机选择两个粒子,进行交叉操作,即将两粒子的速度和位置按一定权重比例进行交叉,生成一个新粒子的速度及位置信息;

(3) 用新粒子的信息取代一个较劣粒子的信息;

(4) 执行操作(2)和(3)直到所有较劣粒子被取代,从而形成新的种群;

(5) 反复执行操作(1)~(4),通过不断随机交叉,多次分割搜索区域.

其数学描述如下:

$$V_i = aV_j + (1-a)V_k \quad (6)$$

$$X_i = aX_j + (1-a)X_k \quad (7)$$

其中, V_j 和 V_k 为随机选择的两个较优粒子 j 和 k 的速度, V_i 为新粒子 i 的速度, a 为常数,在 $[0, 1]$ 之间. X 表示粒子位置,意义与 V 对应类似. 以一维搜索区域为例,其分割原理如图 1 所示:

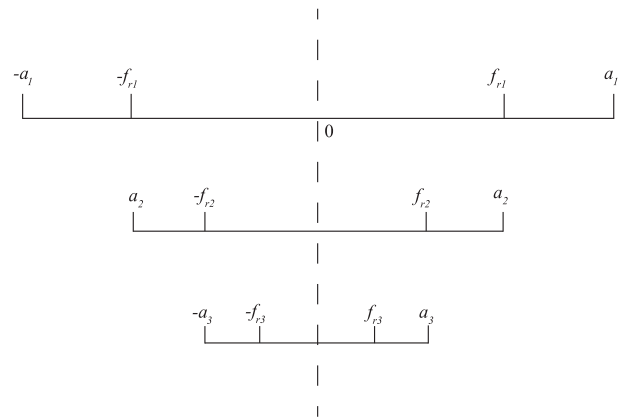


图1 区域分割原理示意图

其中, $[-a_1, a_1]$ 为一维搜索空间边界,经过交叉后粒子集中于 $[-f_1, f_1]$ 中,从而形成新的搜索区域 $[-a_2, a_2]$,再次交叉后粒子集中于 $[-f_2, f_2]$,形成更小搜索区域 $[-a_3, a_3]$,接着交叉后粒子集中于 $[-f_3, f_3]$ ……,通过不断循环迭代分割区域最终使所有粒子均集中于最优解附近. 这里的关键是进行交叉的粒子是从较优粒子中随机选择,而非两个最优粒子,也即较劣粒子被取代的本质不是向最优粒子或者某个特定粒子靠拢,而是向整个界限内靠近,从而既达到收敛的目的同时也保证了粒子的多样性.

3.3 自适应变异

基于以上改进,算法的收敛速度及收敛精度均有大幅提高,但仍有可能陷入局部最优而无法跳出. 因此,在此基础上做进一步改进. 具体策略为:

判断粒子是否集中于极小区域,判断标准为满足

以下两条件之一:

(1) g_{best} 在连续 b 次迭代内均无改善, b 的理想取值为:

$$5\% \leq \frac{b}{t_{\max}} \leq 10\% \quad (8)$$

(2) 假定全局最优粒子为 k , 定义粒子 j 与全局最优粒子之间的欧氏距离为 l_{kj} , 定义粒子与全局最优粒子的距离函数 $s(l)$ 为:

$$s(l) = \sum_{j=1}^D \frac{1}{\pi l_{kj}}, j \neq k \quad (9)$$

其满足:

$$s(l) < t_i \cdot n \cdot \sqrt{D} \cdot e^c \quad (10)$$

其中, n 为粒子个数, c 为 $[0, 1]$ 之间的常数.

粒子执行自适应变异的前提为粒子满足集中极小区域的条件, 变异扰动采用以下策略:

粒子以其与全局最优粒子的欧氏距离大小决定其变异概率, 此概率表达为:

$$p = d \left(\frac{L_{\max} - l_{kj}}{L_{\max}} \right)^2 \quad (11)$$

其中, d 为变异概率调节因子, 其值在 $[0, 1]$ 之间. 从而 p 在 $[0, d]$ 之间, 且 l_{kj} 越小则 p 越大, 也即粒子离全局最优粒子越近则以越大概率产生扰动, 从而更可能探索新区域, 增大发现最优解的概率. 若粒子产生变异, 也即 $[0, 1]$ 之间随机产生函数 $rand \leq p$, 则对产生变异的粒子 j 执行以下操作以生成新的速度与位置:

$$V_j = V_j + Cauchy \cdot \frac{e^{-t_i}}{l_{kj}} \cdot rand \quad (12)$$

$$X_j = X_j + Cauchy \cdot \frac{e^{-t_i}}{l_{kj}} \cdot rand \quad (13)$$

其中, 引入的柯西变异算子 $Cauchy$ 的表达式为:

$$Cauchy(x) = \frac{1}{\pi} \frac{t}{t^2 + x^2}, -\infty < x < \infty, t > 0 \quad (14)$$

$Cauchy$ 算子的作用是保证数列的收敛性, 从而保证粒子的速度与位置不会因为变异过大而无法收敛. 引入与当前迭代次数相关的指数衰减因子 e^{-t} , 使其与迭代次数关联起来, 可使粒子变异的速度及位置而在迭代后期不致过大; 进一步引入 $rand$ 函数增加了粒子的随机性和独立性.

4 仿真实验及结果分析

4.1 实验设置

实验硬件环境为 Pentium(R) Dual-Core CPU E5800 (3.2GHz), 内存为 4GB, 软件平台为 MATLAB 2014a. 本文选用测试函数为 CEC2015 最新基准测试函数集中的部分测试函数, 如表 1 所示. 表中特征“U”表示此函数为单峰函数, “M”与“S”分别表示多峰与动态函数. 所选取的 8 个函数 ($f_1 - f_8$), 既有单峰测试其收敛速度和收敛精度, 也有多峰函数测试其逃离局部最优的能力, 还有动态函数测试算法动态优化性能; 选取的维度覆盖了从低维到高维等众多维度, 以及更为复杂的复合函数, 能够比较有效客观全面检测不同算法的性能.

选择对比的算法为: 标准 PSO; 文献[10]中提出的 GLbest-PSO, 此算法将全局最优与局部最优引入惯性权重中; 文献[11]中提出的 ALEMSPSO, 此算法将群体分为自主学习和精英群的多子群, 不同种群进行通信来确定全局最优; 文献[12]中提出的 EICS-CPSO, 此算法采用免疫克隆选择的策略并采用协同进化的思想保持种群多样性. 算法采用统一参数设置, 为了减少偶然误差带, 所有实验均重复 30 次, 结果为 30 次均值, 种群大小 $N = 30$, 最大迭代次数 $T = 1000$, $c_1 = c_2 = 1.495$. 由于本文算法参数设置满足公式(3)的收敛性原则, 所以算法满足收敛性. 不同算法的独立参数均按原文献设置.

表 1 测试函数

表达式	维度	特征	取值范围	理论最优
$f_1(X) = \sum_{i=1}^{30} x_i^2$	30	US	(-100, 100)	$f(0) = 0$
$f_2(X) = \sum_{i=1}^{29} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	0	MS	(-30, 30)	$f(1) = 0$
$f_3(X) = \frac{1}{4000} \sum_{i=1}^{30} x_i^2 - \prod_{i=1}^{30} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	M	(-600, 600)	$f(0) = 0$
$f_4(X) = -20 \exp(-0.2 \sqrt{\frac{1}{30} \sum_{i=1}^{30} x_i^2}) - \exp\left(\frac{1}{30} \sum_{i=1}^{30} \cos 2\pi x_i\right) + 20 + e$	30	MS	(-32, 32)	$f(0) = 0$
$f_5(X) = 0.5 + \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2}$	2	M	(-100, 100)	$f(0) = 0$

续表

表达式	维度	特征	取值范围	理论最优
$f_6(X) = \sum_{i=1}^{30} x_i + \prod_{i=1}^{30} x_i $	30	U	(-10, 10)	$f(0) = 0$
$f_7(X) = \sum_{i=1}^{10} [A^{(\frac{i-1}{9})} (x_i^2)], A = 1 \times 10^6$	10	US	(-100, 100)	$f(0) = 0$
$f_8(X) = 0.1 \sin^2(3\pi x_i) + \sum_{i=1}^{29} (x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1})] + (x_{30} - 1)^2 [1 + \sin^2(2\pi x_{30})] + \sum_{i=1}^{30} u(x_i, 5, 100, 4)$	30	M	(-50, 50)	$f(0) = 0$

4.2 RSVPSO 算法参数设定

RSVPSO 中固定参数均根据多次实验统计得出, 如表 2 所示:

表 2 RSVPSO 参数设置

参数符号	参数值
a	0.5
b	50
c	0.35
d	0.8
e	1.0

4.3 不同算法的对比实验及分析

实验在 2 种方案下进行: 固定迭代次数、固定迭代

表 3 不同算法的收敛精度及稳定性对比

函数	评价指标	PSO	GLbest-PSO	ALEMSPSO	EICS-CPSO	RSVPSO
f_1	Mean	1.13e-1	3.47e-2	1.50e-146	9.28e-235	4.20e-320
	Deviation	1.88e-2	7.48e-2	7.40e-144	2.53e-233	1.37e-319
	t -test	33.43	1.93	0.01	0.20	0
f_2	Mean	1.33e+1	8.07e+0	1.47e-11	1.49e-14	1.23e-27
	Deviation	3.20e+1	5.14e+0	7.31e-10	5.81e-14	2.26e-26
	t -test	2.31	8.74	0.11	1.43	0
f_3	Mean	1.18e-1	9.01e-3	1.23e-9	2.47e-11	2.65e-14
	Deviation	1.43e-1	2.36e-2	1.95e-9	5.54e-10	2.72e-14
	t -test	4.62	2.13	3.51	0.25	0
f_4	Mean	8.12	4.56	5.78e-11	9.32e-12	3.24e-15
	Deviation	7.11	5.37	9.34e-10	6.32e-11	6.87e-15
	t -test	6.34	4.75	0.34	0.82	0
f_5	Mean	2.27e-3	3.56e-6	9.97e-7	2.87e-11	1.36e-15
	Deviation	2.25e-2	2.48e-5	2.81e-6	1.15e-10	1.41e-14
	t -test	0.17	0.25	0.62	0.43	0
f_6	Mean	1.41e+1	1.95	1.03e-72	4.25e-141	1.11e-239
	Deviation	2.83e+1	4.75	1.67e-72	9.75e-140	3.32e-237
	t -test	2.76	2.29	3.43	0.24	0
f_7	Mean	6.78e-7	2.78e-26	4.66e-136	9.08e-178	7.21e-282
	Deviation	1.24e-8	9.94e-26	2.78e-135	5.07e-178	2.47e-280
	t -test	1.81	3.31	0.53	5.93	0
f_8	Mean	1.08e-4	2.55e-6	3.34e-15	2.66e-22	1.98e-29
	Deviation	4.12e-4	6.08e-6	9.57e-14	9.18e-22	4.77e-28
	t -test	1.46	2.34	0.20	1.61	0

时间. 分别从多角度全面地对比算法的优劣.

实验 1 固定迭代次数 $T = 1000$, 其他设置均不变. 对比的指标为 30 次结果平均值 (Mean), 标准差 (Deviation), 并采用 t -test 统计检测法来更全面评判算法优劣, 由数理统计可知, $t = 1.645$ 时, 表明本文算法有 95% 的置信度优于其他算法. 通过最小值和平均值来评价算法收敛精度, 通过标准差来评价算法稳定性, 通过 t 检测来衡量算法优劣. 实验数据如表 3.

由表 3 可知, RSVPSO 在 12 个测试函数上无论是适应值平均值及标准差上均具有明显优势. 具体来讲, RSVPSO 在单峰函数上相较其他算法在精度上有较大提高, 尤其是在 f_1 , f_6 及 f_7 的均值上相比其他算法提高

了 85 ~ 319 个数量级,其标准差也提高了类似数量级;说明 RSVPSO 在单峰函数的收敛精度及稳定性比其他算法要好,这得益于 RSVPSO 采用的区域分割的思想达到了快速收敛并优化粒子的目的,并且采用动态非线性惯性权重保证了粒子的独立性和灵活性.此外,RSVPSO 在多峰函数上的两个参考指标也有不同程度的提高.相比于 PSO 及 GLbest-PSO 具有较大提高,说明算法对复杂的多峰函数也有较好效果,即引入柯西算子的自适应变异策略大幅提高了粒子逃离局部最优的能力,算法具有较强的全局搜索能力.并且 RSVPSO 在动态函数 f_1 , f_2 , f_4 及 f_7 上的寻优精度和稳定性也较其它

算法有明显提高,说明它具有良好的动态寻优性能.在 t -test 检测中,RSVPSO 的 t -test 值大多大于或等于 1.645,说明算法在大多数情况下优于其他算法的置信度超过 95%,从而更全面客观地论证了本算法的优越性.

实验 2 固定迭代时间 $t = 0.5s$,其他设置均不变,对比指标为在固定时间下的收敛精度(Fbest)及迭代次数(FEs).通过 Fbest 来检验 RSVPSO 算法收敛速度,通过 FEs 来检验算法的执行效率,即算法的复杂性.实验数据见表 4.

表 4 不同算法的收敛速度效率对比

测试函数	评价指标	PSO	GLbest-PSO	ALEMSPSO	EICS-CPSO	PSAPSO
f_1	Fbest	6.28e-2	2.89e-2	2.73e-14	1.41e-9	1.05e-258
	FEs	1103	1022	208	177	645
f_2	Fbest	8.23e+1	3.67e+1	3.45e-7	2.58e-6	2.45e-27
	FEs	1092	981	198	156	612
f_3	Fbest	3.13e+0	1.67e-1	4.01e-6	1.29e-8	3.89e-13
	FEs	639	568	202	162	337
f_4	Fbest	1.35e+1	9.42e+0	8.37e-3	9.83e-2	6.45e-15
	FEs	722	649	203	151	323
f_5	Fbest	5.32e-3	5.32e-3	1.53e-5	3.23e-8	6.82e-13
	FEs	905	848	228	165	462
f_6	Fbest	9.58e+0	2.03e+0	6.81e-9	5.50e-6	1.30e-103
	FEs	1106	969	235	171	496
f_7	Fbest	5.24e-2	2.81e-22	8.06e-24	9.98e-26	4.07e-213
	FEs	964	912	195	172	648
f_8	Fbest	3.24e-2	8.79e-5	1.24e-3	7.24e-4	2.28e-27
	FEs	768	702	164	148	384

由表 4 可知,以更合理的固定时间来检测算法性能,RSVPSO 优势更为明显.这是因为 GLbest-PSO 算法结构虽然简单,但是其改进也相对有限,仅仅是对惯性权重的改进,而没有对算法的结构进行改进,虽然在固定时间内同 PSO 一样迭代次数较多,但是收敛速度及跳出局部陷阱的能力较弱;而 ALEMSPSO 及 EICS-CPSO 虽然在固定迭代次数时性能相比 PSO 及 GLbest-PSO 有较大提升,但是在固定时间的前提下,性能则提升不大,甚至在某些指标上不如 GLbest-PSO(如 f_8).这是由于其进行了复杂的多种群设计,大大增加了单次迭代中的时间损耗,因而导致迭代次数较少,效率不高;而 RSVPSO 首先是对权重系数进行了较大优化,此改进所增加时间有限,同时区域分割策略所使用的交叉策略既能有效提升算法收敛速度,也保持了算法低

复杂度的特性,因而在固定时间内,迭代次数相对较多,收敛速度及精度也有较大提高,综合优势更为明显.

为了更直观地对比各算法的收敛速度及精度,给出算法在测试函数 $f_1 - f_8$ 的适应值变化的曲线图,如图 2 所示.

图 2 直观地反映各算法在 8 个测试函数的收敛速度及收敛精度.相较于其他算法,RSVPSO 算法在这两个最为重要的指标上均有明显优势.其较好的收敛性得益于其采用了动态非线性惯性权重,同时信息交叉使粒子速度及位置变化远比基本粒子群算法快,大大提高了粒子搜索效率;最后基于柯西算子的自适应变异也增大了粒子逃出局部最优的能力,从而保证了粒子搜索解的精度.

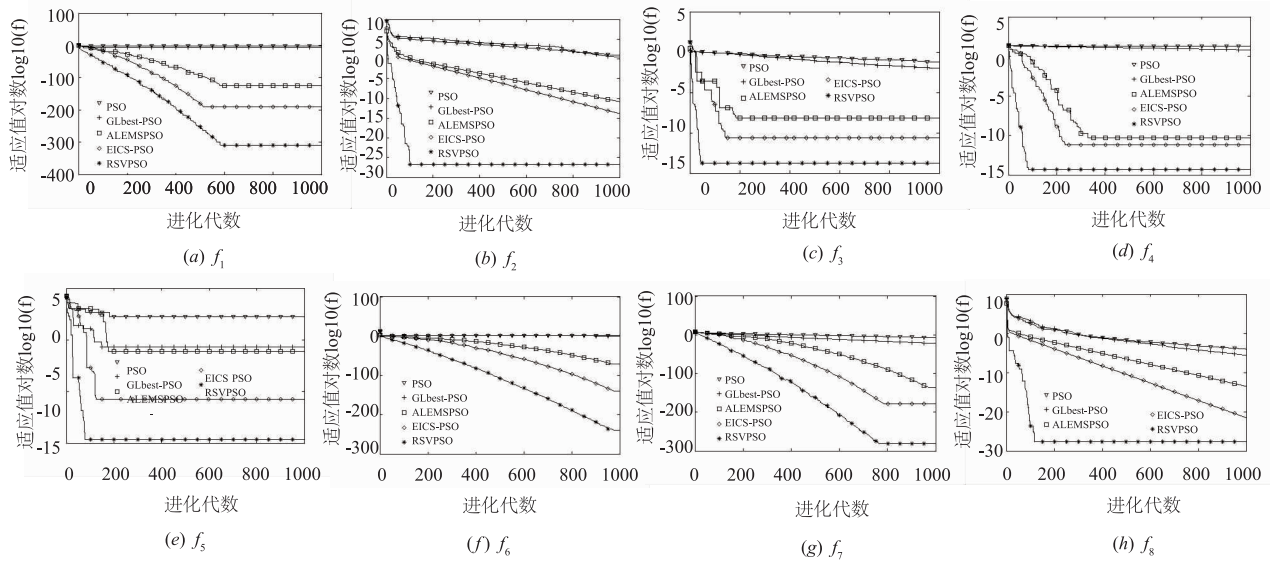


图2 不同算法在测试函数上的收敛曲线

5 结论

针对基本 PSO 算法收敛速度与“早熟”问题之间的矛盾,通过对惯性权重进行非线性优化,并利用信息交叉的区域分割思想提高收敛速度,同时采用柯西算子的自适应变异确保收敛精度。理论及仿真结果均表明 RSVPSO 算法对比基本 PSO 及部分改进算法均有较大优势,在收敛速度、收敛精度、执行效率、稳定性及算法复杂度方面具有良好性能,可以作为一种优良的 PSO 改进算法。

参考文献

- [1] Kennedy J, Eberhart R C. Particle swarm optimization [A]. Proceedings of IEEE International Conference on Neural Networks [C]. Perth, Australia, 1995. 1942 – 1948.
- [2] Shi Y, Eberhart R C. Fuzzy adaptive particle swarm optimization [A]. Proceeding of the IEEE International Conference on Evolutionary Computation [C]. Annapolis, IEEE, 1998. 68 – 73.
- [3] Lovbjerg M, Rasmussen T K, Krink T, et al. Hybrid particle swarm optimizers with breeding and subpopulations [A]. Proc of the Genetic and Evolutionary Computation Conference [C]. San Francisco: Morgan Kaufmann, 2001. 469 – 476.
- [4] Yangyang Li, Licheng Jiao, et al. Dynamic-context cooperative quantum-behaved particle swarm optimization based on multi-level thresholding applied to medical image segmentation [J]. Information Sciences, 2015, 294: 408 – 422.
- [5] Yangyang Li, Rongrong Xiang, Licheng jiao and Ruochen Liu. An improved cooperative quantum-behaved particle

swarm optimization [J]. Soft Computing, 20 – 12, 16 (6): 1061 – 1069.

- [6] Angeline P J. Using selection to improve particle swarm optimization [A]. Proceedings of the IEEE International Conference on Evolutionary Computation [C]. Alaska, USA: IEEE, 1998. 84 – 89.
- [7] Xin B, Chen J, Peng Z H, et al. An adaptive hybrid optimizer based on particle swarm and differential evolution for global optimization [J]. Science China (Information Sciences), 2011, 53 (5): 980 – 989.
- [8] 周剑波, 刘宏立, 徐琨. 一种结合粒子群和虚拟力的动态节点部署策略 [J]. 计算机工程与应用, 2016, 52 (10): 118 – 123.
ZHOU Jianbo, LIU Hongli, XU Kun. Dynamic sensor deployment strategy to combine particle swarm and virtual-force [J]. Computer Engineering and Applications, 2016, 52 (10): 118 – 123. (in Chinese)
- [9] 吴晓军, 李峰. 均匀搜索粒子群算法的收敛性分析 [J]. 电子学报, 2012, 29 (11): 1115 – 1120.
WU Xiaojun, LI Feng. The convergence analysis of the uniform search particle swarm optimization [J]. Acta Electronica Sinica, 2012, 29 (11): 1115 – 1120. (in Chinese)
- [10] M Senthil Arumugam, MVC Rao. On the improved performances of the particle swarm optimization algorithms with adaptive parameters, crossover operators and root meansquare (RMS) variants for computing optimal control of a class of hybrid systems [J]. Appl Soft Comput, 2008; 324 – 336.
- [11] 姜海燕, 王芳芳, 郭小清, 等. 基于自主学习和精英群的多子群粒子群算法 [J]. 控制与决策, 2014, 29 (11): 2034 – 2040.

JIANG Haiyan, WANG Fangfang, GUO Xiaoqing, et al. Multi-swarm particle swarm optimization based on autonomous learning and elite swarm [J]. Control and Decision, 2014, 29(11): 2034 - 2040. (in Chinese)

[12] 刘朝华, 李小花, 章兢. 精英免疫克隆选择的协同进化

粒子群算法[J]. 电子学报, 2013, 41(11): 2167 - 2173.

LIU Zhaohua, LI Xiaohua, ZHANG Jing. Co-evolutionary particle swarm optimization algorithm based on elite immune clonal selection [J]. Acta Electronica Sinica, 2013, 41(11): 2167 - 2173. (in Chinese)

作者简介



陈侃松 男, 1972 年生, 教授, 博士生导师. 主要研究方向为无线传感网络及应用.
E-mail: kschen1999@aliyun.com



阮玉龙 男, 1993 年生, 硕士研究生, 主要研究方向为无线传感网络节点部署.