

粒子群算法的交互性与随机性分析

刘建华¹ 刘国买² 杨荣华¹ 胡文瑜¹

摘要 在现有分析结论的基础上, 分别采用优化的凸性理论和概率收敛理论, 分析了粒子群 (Particle swarm optimization, PSO) 算法的交互性和随机性对算法的影响. 分析得出, 在不考虑随机性的条件下, 当 PSO 算法优化单峰函数时, 交互性使粒子最终收敛于全局最优粒子位置; 当 PSO 算法优化多峰函数时, 交互性未必使粒子最终收敛于全局最优位置. 但如果考虑随机性, 算法优化的目标函数无论是单峰函数还是多峰函数, 粒子都会依概率收敛于最优位置. 通过基准函数的实验验证了分析的结论.

关键词 粒子群算法, 收敛性, 交互性, 随机性

引用格式 刘建华, 刘国买, 杨荣华, 胡文瑜. 粒子群算法的交互性与随机性分析. 自动化学报, 2012, 38(9): 1471–1484

DOI 10.3724/SP.J.1004.2012.01471

Analysis of Interactivity and Randomness in Particle Swarm Optimization

LIU Jian-Hua¹ LIU Guo-Mai² YANG Rong-Hua¹ HU Wen-Yu¹

Abstract Based on the existing theoretical results, this paper analyzes the influence on convergence of particle swarm optimization (PSO) exerted by interactivity of PSO using the conver optimization theory, then analyzes the randomness of PSO using the probability convergence theory. It is concluded that when PSO is used to optimize the unimodal function without consideration of randomness of particle movement, the trajectory of particle will eventually converge to the global optimal location; when PSO is used to optimize the multimodal function particles, the trajectory of particle may not necessarily converge to the global optimal location. However, if the randomness of particle movement is considered for analysis of PSO, particles will all converge to the optimal location in probabilities no matter the objectives function is unimodal or multimodal. Experiments are conducted on beckmark function to test these conclusions.

Key words Particle swarm optimization (PSO), convergence, interactivity, randomness

Citation Liu Jian-Hua, Liu Guo-Mai, Yang Rong-Hua, Hu Wen-Yu. Analysis of interactivity and randomness in particle swarm optimization. *Acta Automatica Sinica*, 2012, 38(9): 1471–1484

粒子群算法 (Particle swarm optimization, PSO) 是 Eberhart 和 Kennedy 于 1995 年开发的一种进化计算技术^[1–2], 源于对鸟群捕食的行为研究. Shi 等^[3] 在 PSO 算法中引入惯性权重 w , 更好地控制算法的收敛 (Convergence) 和探索 (Exploration), 形成了标准 PSO 算法. PSO 算法的提出只有十几年, 其研究主要集中在算法效率的改进及其应用上, 而算法分析相对偏少, 尤其对算法的收敛性分析目前并不完善^[4–8]. PSO 算法分析一直是 PSO 算法研究的难点, 因为 PSO 算法具有随机性并且粒子之间相互作用, 其粒子运行轨迹很复杂, 使得很多常规数学方法手段对其无效. 目前, PSO 算法的分

析方法基本上是在简化 PSO 算法模型的条件下去掉的, 将 PSO 算法的随机性去掉、使粒子群的最优位置固定不变、并且在二维空间上分析, 不考虑粒子间的相互作用.

早期, Ozcan 等假定最优位置 p_i 和 p_g 为常数且两者相等, 随机数 c_1 和 c_2 固定不变, 并对不同区域进行了轨迹分析^[9], 得出粒子轨迹是随时间变化的波形, 随后对简化模型稍作扩展, 进一步分析参数对轨迹的影响^[10]. Clerc 等在文献 [11] 中分析了 PSO 算法迭代公式的参数, 先将随机变量固定为常量, 再用状态转移矩阵研究一个粒子的运动轨迹, 从而得到算法的单个粒子轨迹收敛的参数约束条件. 尽管 Clerc 的分析只给出 PSO 算法的参数约束条件, 但对 PSO 算法的应用起到指导作用. 由于 Solis 等对随机优化算法提出了其全局收敛须满足的条件^[12], Van den Bergh 在此基础上, 对 PSO 算法收敛性做了进一步的分析^[13], 证明了原始 PSO 算法是无法保证局部收敛的. Van den Bergh 在文献 [13] 中分析得出, 粒子群是一个离散线性系统, 如果特征根的值小于 1, 则粒子将收敛于平衡点. 因为系统的特征值是系统参数的函数, 这种模型能建议如何设置参数可以保证算法收敛, 并且 Van de Bergh

收稿日期 2011-07-25 录用日期 2012-01-10
Manuscript received July 25, 2011; accepted January 10, 2012
福建省资助省属高校基金 (JK2011035), 福建工程学院人才启动基金 (E0600100) 资助
Supported by Foundation of Supporting Province's College and University of Fujian (JK2011035) and Research Driving Foundation of Fujian University of Technology (E0600100)
本文责任编辑 贾云得
Recommended by Associate Editor JIA Yun-De
1. 福建工程学院计算机与信息科学系 福州 350108 2. 福建工程学院管理学院 福州 350108
1. Department of Computer and Information, Fujian University of Technology, Fuzhou 350108 2. School of Management, Fujian University of Technology, Fuzhou 350108

等在文献 [14] 中分析出, PSO 算法不能保持收敛于问题的局部最优点, 并由提出一种全局收敛的 PSO 改进算法. 在类似的简化模型上, Blackwell 分析了粒子的空间聚集度与时间的变化关系^[15], 其分析的结论得出, 在没有交互性情况下, 粒子空间聚集度与时间是呈指数级的降低. 这种分析考察了 PSO 算法的多样性问题. Trelea 对简化 PSO 模型的动态行为和收敛性进行了分析^[16], 应用离散动态系统理论推导出对参数选择的参考准则. 最近, 对非全局性拓扑结构的 PSO 算法也展开了分析研究, 并分析了算法的局部随机性质^[17]. 文献 [18] 对线性和离散 PSO 算法作了随机稳定性分析.

国内有不少文献分析了 PSO 算法的参数选择问题, 提出参数选择的依据^[19-20]. 文献 [21] 通过建立标准粒子群优化算法的三种典型模型, 分析了三种模型的粒子群动态特性, 结果表明粒子群算法的搜索空间是非常有限的, 并且很容易陷入停滞点. 文献 [22] 建立算法的随机过程模型, 分析群体的随机性收敛, 利用随机过程理论进行分析^[22-23]. 文献 [24] 也用随机过程对 PSO 算法分析, 但分析中还是采用数学理论的差分方程. 任子晖等采用随机过程的马尔科夫链的方法对 PSO 算法分析, 但由于 PSO 算法一般计算的连续性问题, 所以必须对问题离散处理^[25], 申元霞等对 PSO 算法的概率特性进行简单分析, 只是为了提出改进 PSO 算法多样性问题^[26], 而文献 [27] 对离散 PSO 算法进行了分析.

由于现有 PSO 算法的分析方法是在简化模型的基础上进行, 即孤立单个个体、固定群体的最优位置和个体的最优位置不变并且去掉算法的随机性. 因此, 最后分析的数学模型是近似的, 没有考虑粒子间的交互性和算法的随机性, 并且需要通过试验验证. 这样的方法存在下列缺点^[4]: 1) 没有考虑粒子间的相互影响, 没有考虑 PSO 算法的“群体性”和“交互性”, 缺少了最优位置的动态变化性; 2) 忽视了算法中随机性的作用, 粒子群算法引入了随机因素, 这种人工噪声的干扰对算法一定有正面的作用. 所以, 目前 PSO 算法的理论分析成果是不完善, 没有真正反映算法的运行机制及真正的收敛性.

本文在现有的 PSO 算法分析结论^[13-14] 的基础上, 通过考虑算法随机性及粒子间交互性, 采用优化的凸性理论, 对 PSO 算法作进一步地深入分析. 首先证明, 算法的交互性能使算法在计算单峰函数时, 粒子的位置最终收敛于群体最优位置. 然后证明, 算法的随机性能使算法计算多峰函数时, 粒子的位置依概率收敛于群体最优位置, 并通过实验验证分析的结论. 本文主要是针对标准 PSO 算法进行分析, 内容具体安排如下: 第 1 节介绍相关的工作; 第 2 节分析 PSO 算法的交互性对粒子轨迹的影响; 第 3 节分析 PSO 算法的随机性对粒子轨迹的影响; 第

4 节分析介绍粒子运动轨迹与算法收敛的关系; 第 5 节是本文的结论.

1 相关的工作

1.1 粒子运动轨迹分析

假设对目标函数 $f(x)$ 的优化为求其最小值, 即 $\min f(x)$. 目标函数 $f(x)$ 的搜索空间 S 是有限的, 并且函数 $f(x)$ 在搜索空间 S 中存在最优解.

标准 PSO 算法描述如下, 假设搜索空间 S 是 D 维的, 粒子群中第 i 个粒子的位置用 $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ 表示, 第 i 个粒子的速度表示为 $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$. 第 i 个粒子迄今为止搜索的最好位置记为 $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})$, 整个粒子迄今为止搜索到的最好位置记为 $p_g = (p_{g1}, p_{g2}, \dots, p_{gD})$. 对于每一个粒子, 其第 d 维 ($1 \leq d \leq D$) 根据如下等式变化:

$$v_{id} = \omega \times v_{id} + c_1 \times \text{rand}(\cdot) \times (p_{id} - x_{id}) + c_2 \times \text{rand}(\cdot) \times (p_{gd} - x_{id}) \quad (1)$$

$$x_{id} = x_{id} + v_{id} \quad (2)$$

其中, $\text{rand}(\cdot)$ 是在区间 $[0, 1]$ 之间均匀采样的随机数.

两个基本式主要用于 D 维空间 n 个粒子的表达. 因简化算法的条件, 粒子轨迹的分析对不同粒子、在不同维中是一样的, 所以不考虑粒子下标 i 和维下标 d , 把两个下标 i 和 d 去掉, 而粒子的轨迹将在离散时间中考虑, 所以 v_{id} 和 x_{id} 可以表示为 $v(t)$ 和 $x(t)$, 其分别表示粒子在时间步 t 时刻的速度与位置, 并且标准 PSO 算法还有下列更新公式:

$$\text{If } f(x(t)) < f(p), \text{ then } p = x(t) \quad (3)$$

$$\text{If } f(p) < f(p_g), \text{ then } p_g = p$$

在简化模型中, 文献 [13] 利用线性系统理论分析了粒子的收敛性, 并且得出, 粒子收敛到一个固定点, 即群体最优点与个体最优点的加权和. 其分析的过程简单叙述如下:

令 $\phi_1 = c_1 \times \text{rand}(\cdot)$, $\phi_2 = c_2 \times \text{rand}(\cdot)$, $\phi = \phi_1 + \phi_2$. 假定 p , p_g , ϕ_1 和 ϕ_2 保持固定的值, 群体最优值 p_g 和个体最优值 p 也保持不变, 则由式 (1) 和式 (2) 消去速度相关项, 经过求解并可以推得^[13]:

$$x(t) = k_1 + k_2 \alpha^t + k_3 \beta^t \quad (4)$$

这里 k_1 , k_2 和 k_3 是常数, 由系统的初始条件决定. 进一步可推得:

$$k_1 = \frac{\phi_1 p + \phi_2 p_g}{\phi_1 + \phi_2} \quad (5)$$

根据式(4), 如果 $x(t)$ 要收敛, 那么 $\max(\|\alpha\|, \|\beta\|) < 1$. 当不考虑随机性, 且个体最优点和群体最优点为常数时, 文献[13]根据 $\max(\|\alpha\|, \|\beta\|) < 1$ 推得, PSO 算法收敛条件是参数 ω, ϕ_1, ϕ_2 需要满足下列关系:

$$\begin{cases} 0 < \omega < 1 \\ 0 < \phi < 4 \\ \omega > \frac{1}{2}\phi - 1 \end{cases} \quad (6)$$

而根据式(4)和式(5), 粒子的位置 $x(t)$ 按下式收敛:

$$\lim_{t \rightarrow \infty} x(t) = (1-a)p + ap_g, \quad a \in [0, 1] \quad (7)$$

结论式(6)及式(7)是在一些假定条件下成立的, 其假定去掉 PSO 算法的随机性且粒子运动过程中个体最优位置 p 和群体最优位置 p_g 固定不变. 这些假定条件约束粒子的运动, 与 PSO 算法真实运行条件有差距. 因此, 当 PSO 算法存在随机性且个体最优位置与群体最优位置动态变化时, 需要进一步分析其运行轨迹.

1.2 凸集与凸函数

PSO 算法是智能优化算法, 而优化算法的理论常用到的基本概念是凸集与凸函数, 它对于优化算法的收敛性分析有非常重要作用. 本节简要介绍凸集和凸函数^[28].

定义 1. 设 S 为 n 维欧氏空间 E_n 中的一个集合. 若对任意 $x_1, x_2 \in S$ 都有 $\lambda x_1 + (1-\lambda)x_2 \in S$, $\lambda \in [0, 1]$, 则称 S 为凸集.

定义 1 说明, 若对 S 中的任意两点, 它们的连线仍然属于 S .

凸函数通常也叫做单峰函数, 其定义如下:

定义 2. 设 $f(x)$ 是在闭凸集 S 上定义的实函数, $f(x)$ 在闭凸集 S 上的极小点是 x_0 , 并且对任意的 $x_1, x_2 \in S, x_1 < x_2$, 有:

$$\text{当 } x_2 \leq x_0 \text{ 时, } f(x_1) > f(x_2);$$

$$\text{当 } x_0 \leq x_1 \text{ 时, } f(x_2) > f(x_1)$$

则称函数 $f(x)$ 是在闭凸集 S 上的凸函数, 即单峰函数.

不满足上述定义的函数叫非凸函数, 即多峰函数. 根据定义, 单峰函数 $f(x)$ 的全局最小值是 $f(x_0)$, 并且不存在其他的局部最小值. 因此, 单峰函数有下列性质:

引理 1. 若 $f(x)$ 是在闭凸集 S 上的一个单峰函数, 则对任意 $x_1, x_2 \in S$, 有下式成立:

$$f((1-\alpha)x_1 + \alpha x_2) < \max(f(x_1), f(x_2))$$

其中 $a \in [0, 1]$.

根据定义, 单峰函数在 S 上的局部极小点是在 S 上的全局最小点, 并且多峰函数是由若干个单峰函数组成.

2 PSO 算法的交互性对粒子轨迹的影响

粒子个体最优位置与群体最优位置分别是 p 和 p_g , PSO 算法粒子之间相互协作、相互交互是通过这两个位置的变化实现. 分析 PSO 算法的交互性对粒子轨迹的影响需要分析, p 和 p_g 的动态变化如何影响 PSO 算法的粒子轨迹. 因 p 和 p_g 随迭代时间 t 变化, 可以分别表示为 $p(t)$ 和 $p_g(t)$. 为了分析简单, 这里假定算法去掉随机性. 根据 PSO 算法, $p(t), p_g(t)$ 分别按下列两式变化:

$$p(t+1) = \begin{cases} p(t), & \text{if } f(x(t+1)) \geq f(p(t)) \\ x(t+1), & \text{if } f(x(t+1)) < f(p(t)) \end{cases} \quad (8)$$

$$p_g(t+1) = \begin{cases} p_g(t), & \text{if } f(p(t+1)) \geq f(p_g(t)) \\ p(t+1), & \text{if } f(p(t+1)) < f(p_g(t)) \end{cases} \quad (9)$$

其中, $f(x)$ 是优化问题的目标函数. 这两个式子反映 PSO 算法粒子间的交互性.

2.1 最优位置的收敛性

本节先证明 $\lim_{t \rightarrow \infty} p(t), \lim_{t \rightarrow \infty} p_g(t)$ 存在, 然后再证明粒子位置的收敛.

定义 3. 若 S 为优化目标函数 $f(x)$ 的搜索空间, 则集合 $S_v = \{x | f(x) = v, x \in S\}$ 为函数 $f(x)$ 的同 v 值区.

显然, 若 v^* 是函数 $f(x)$ 的最小值, 则 S_{v^*} 是函数 $f(x)$ 的最小值区.

引理 2. 若 PSO 算法去掉随机性, 且具有式(8)和式(9)的交互性, 则 $\lim_{t \rightarrow \infty} p(t), \lim_{t \rightarrow \infty} p_g(t)$ 存在.

证明. 假定目标函数 $f(x)$ 的搜索空间 S 是有限的, 并且目标函数 $f(x)$ 在搜索空间 S 中存在全局最优值, 则根据 $p(t), p_g(t) \in S (t = 1, 2, \dots, \infty)$, 数列 $\{f(p(t)), t = 1, 2, \dots, \infty\}$ 与 $\{f(p_g(t)), t = 1, 2, \dots, \infty\}$ 一定存在下界.

再根据式(8)和式(9), 数列 $\{f(p(t)), t = 1, 2, \dots, \infty\}$ 及数列 $\{f(p_g(t)), t = 1, 2, \dots, \infty\}$ 是一个单调非递增数列.

因此, 数列 $\{f(p(t)), t = 1, 2, \dots, \infty\}$ 和数列 $\{f(p_g(t)), t = 1, 2, \dots, \infty\}$ 是单调有界的数列, 一定存在极限. 令 $\lim_{t \rightarrow \infty} f(p(t)) = f_p^*$, $\lim_{t \rightarrow \infty} f(p_g(t)) = f_g^*$;

假设 $S_{f_p^*} = \{x|f(x) = f_p^*, x \in S\}$ 和 $S_{f_g^*} = \{x|f(x) = f_g^*, x \in S\}$, 则有:

$$\lim_{t \rightarrow \infty} p(t) \in S_{f_p^*} = \{x|f(x) = f_p^*, x \in S\}$$

$$\lim_{t \rightarrow \infty} p_g(t) \in S_{f_g^*} = \{x|f(x) = f_g^*, x \in S\}$$

根据式 (8) 和式 (9), $\lim_{t \rightarrow \infty} p(t)$ 和 $\lim_{t \rightarrow \infty} p_g(t)$ 的值不能分别在以上两集合内变化, 必定为集合某一固定值.

所以, $\lim_{t \rightarrow \infty} p(t), \lim_{t \rightarrow \infty} p_g(t)$ 存在. \square

如果 $\lim_{t \rightarrow \infty} p(t), \lim_{t \rightarrow \infty} p_g(t)$ 存在, 那么 $\lim_{t \rightarrow \infty} ((1 - \alpha)p(t) + \alpha p_g(t))$ 也一定存在. 随着算法迭代运行, 数列 $\{f(p(t)), t = 1, 2, \dots, \infty\}$ 与 $\{f(p_g(t)), t = 1, 2, \dots, \infty\}$ 将逐步趋于某值, 而 $p(t)$ 和 $p_g(t)$ 也将趋于稳定或停止, 一定会有下式成立.

$$\lim_{t \rightarrow \infty} x(t) = \lim_{t \rightarrow \infty} ((1 - \alpha)p(t) + \alpha p_g(t)) = (1 - \alpha)p^* + \alpha p_g^* \tag{10}$$

这里, $p^* = \lim_{t \rightarrow \infty} p(t), p_g^* = \lim_{t \rightarrow \infty} p_g(t)$.

引理 3. 若 PSO 算法去掉随机性, 但具有式 (8) 和式 (9) 的交互性, 则 $\lim_{t \rightarrow \infty} x(t) = (1 - a)p^* + \alpha p_g^*$.

证明. 根据引理 2, 对任意的正数 ε , 存在时间 T_1 , 当 $t > T_1$ 时, $|p(t) - p^*| < \varepsilon/2, |p_g(t) - p_g^*| < \varepsilon/2$. 若此时 $p(t)$ 不变, 令其为 p , 则有 $|p - p^*| < \varepsilon/2$; 同样, 此时 $p_g(t)$ 不变, 令其为 p_g , 则有 $|p_g - p_g^*| < \varepsilon/2$.

此时, 符合式 (7) 条件, 则有 $\lim_{t \rightarrow \infty} x(t) = (1 - a)p + \alpha p_g$, 所以, 也存在 T_2 , 当 $t > T_2$ 时, $|x(t) - (1 - a)p + \alpha p_g| < \varepsilon/2$. 而 $|(1 - a)p + \alpha p_g - [(1 - a)p^* + \alpha p_g^*]| \leq (1 - a)|p - p^*| + \alpha|p_g - p_g^*| < \varepsilon/2$.

所以, 存在 $T = \max\{T_1, T_2\}$, 当 $t > T$ 时, 有:

$$\begin{aligned} |x(t) - [(1 - a)p^* + \alpha p_g^*]| &< |x(t) - \\ &[(1 - a)p + \alpha p_g]| + |(1 - a)p + \alpha p_g - \\ &[(1 - a)p^* + \alpha p_g^*]| < \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon \end{aligned}$$

所以, $\lim_{t \rightarrow \infty} x(t) = (1 - a)p^* + \alpha p_g^*$. \square

引理 2 表明, PSO 算法的最优位置是收敛的; 而引理 3 说明, 当 PSO 算法去掉随机性, 但算法具有交互性时, 粒子位置 $x(t)$ 的轨迹按式 (10) 收敛.

2.2 收敛定理

根据式 (8), 个体最优位置 $p(t)$ 的轨迹在不断变化. 当 PSO 算法的目标函数 $f(x)$ 是单峰函数时, 个体最优位置 $p(t)$ 具有下列性质:

定理 1. 当 PSO 算法优化的目标函数 $f(x)$ 是单峰函数时, 假定 PSO 算法的随机性被去掉且

$p_g(t)$ 保持固定值 p_g , 而 $p(t)$ 保持动态变化, 即满足式 (8); 则 $\lim_{t \rightarrow \infty} p(t) = p_g$.

证明. 由于 $f(p(t)) \geq f(p_g)$, 又由式 (9) 可得 $f(p(t)) \geq f(p(t + 1))$, 因此数列 $\{f(p(t)), t = 1, 2, \dots, \infty\}$ 是一个单调非递增数列且有界, 数列 $\{f(p(t)), t = 1, 2, \dots, \infty\}$ 一定存在极限.

令 $\lim_{t \rightarrow \infty} f(p(t)) = f^*, S_{f^*} = \{x|f(x) = f^*, x \in S\}$ (即 S_{f^*} 为函数 $f(x)$ 同 f^* 值区), 则根据引理 2, 数列 $\{p(t), t = 1, 2, \dots, \infty\}$ 存在极限. 令 $\lim_{t \rightarrow \infty} p(t) = p^*$.

接下来利用反证法:

1) 当 $f^* \neq f(p_g)$ 时, 显然 $\lim_{t \rightarrow \infty} p(t) \neq p_g$. 又 $\lim_{t \rightarrow \infty} p(t) = p^*$, 则 $p^* \neq p_g$.

根据引理 3, 有:

$$\lim_{t \rightarrow \infty} x(t) = (1 - a)p^* + \alpha p_g, a \in (0, 1)$$

由算法的条件可知, $f(p^*) > f(p_g)$, 且函数 $f(x)$ 是单峰函数, 因此根据引理 1 可得 $f((1 - a)p^* + \alpha p_g) \leq f(p^*)$. 再根据 $p(t)$ 的更新公式 (8), $p(t)$ 要发生变化, 即 $p(t) = (1 - a)p^* + \alpha p_g$. 因此, 数列 $\{p(t), t = 1, 2, \dots, \infty\}$ 的极限不为 p^* , 这与 $\lim_{t \rightarrow \infty} p(t) = p^*$ 矛盾. 所以 $f^* \neq f(p_g)$ 不成立.

因此 $f^* = f(p_g)$.

2) 当 $f^* = f(p_g)$ 时, 若 $\lim_{t \rightarrow \infty} p(t) \neq p_g$, 令 $\lim_{t \rightarrow \infty} p(t) = p^*$, 则 $p^* \neq p_g$.

根据引理 3, 有:

$$\lim_{t \rightarrow \infty} x(t) = (1 - a)p^* + \alpha p_g, a \in (0, 1)$$

由于函数 $f(x)$ 是单峰函数且 $f(p^*) < f(p_g)$, 根据引理 1, 可得 $f((1 - a)p^* + \alpha p_g) \leq f(p^*)$. 由于 $p(t)$ 要发生变化, 存在 $p(t) = (1 - a)p^* + \alpha p_g$. 因此, 数列 $\{p(t), t = 1, 2, \dots, \infty\}$ 的极限不可能为 p^* , 这与前面条件矛盾.

综合 1) 和 2), 可知 $\lim_{t \rightarrow \infty} p(t) = p_g$ 成立. \square

对于定理 1 成立的条件说明如下:

1) 目标函数 $f(x)$ 是单峰函数, 其搜索空间 S 有界且为凸集. 目标函数 $f(x)$ 在其搜索空间 S 满足引理 1 的性质, 即对任意 $x_1, x_2 \in S$, 有 $f((1 - \alpha)x_1 + \alpha x_2) < \max(f(x_1), f(x_2))$.

2) 条件 $f(p_g) \leq f(p(t))$ 始终满足. 根据 PSO 算法的 p_g 与 $p(t)$ 含义, 条件 $f(p_g) \leq f(p(t))$ 能够始终满足.

结合式 (7) 及定理 1 的结论, 可以得到 $\lim_{t \rightarrow \infty} x(t) = p_g$. 因此有下列结论:

定理 2. 当 PSO 算法优化的目标函数 $f(x)$ 是单峰函数时, 如果去掉 PSO 算法的随机性, 且 $p_g(t)$ 保持不变, 即为固定值 p_g ; 但 $p(t)$ 动态变化, 即满足式 (8), 则 PSO 算法有 $\lim_{t \rightarrow \infty} x(t) = p_g$.

证明. 根据引理 3, 有:

$$\lim_{t \rightarrow \infty} x(t) = (1 - a)p(t) + ap_g, \quad a \in (0, 1)$$

而根据定理 1, 有 $\lim_{t \rightarrow \infty} p(t) = p_g$, 所以, 有 $\lim_{t \rightarrow \infty} x(t) = p_g$. \square

定理 1 和定理 2 说明, 当 PSO 算法去掉随机性且具有交互性, 并且求解的目标函数 $f(x)$ 是单峰函数时, 一定有 $\lim_{t \rightarrow \infty} x(t) = p_g$, 即粒子的轨迹一定会收敛群体最优位置. 这个定理说明, 交互性使算法在求解单峰函数时, 粒子的轨迹收敛于群体最优粒子 p_g .

2.3 仿真实验与分析

本节将通过仿真实验验证定理 1 和定理 2 的正确性. 为了更好地设计实验, 首先说明两个定理成立的条件, 列出如下:

- 1) 参数 $\omega, \phi_1, \phi_2, p_g$ 是固定不变的常数, 且满足关系式 (6);
- 2) 目标函数 $f(x)$ 是单峰函数, 即满足引理 1;
- 3) $p(t)$ 按式 (8) 变化, 即如果 $f(x(t+1)) < f(p(t))$, 则 $p(t+1) = x(t+1)$, 但 $f(p_g) \leq f(p(t))$ 始终满足.

2.3.1 实验设计

这里选择八个基准函数做试验, 一类是单峰函数, 另一类是多峰函数, 各为四个函数, 列出如下:

第一类是单峰函数:

- 1) 单峰函数 f_1 是 Spherical 函数:

$$f_1(x) = \sum_{i=1}^n x_i^2, \quad -100 < x_i < 100$$

- 2) 单峰函数 f_2 是 Quadric 函数:

$$f_2(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2, \quad -100 < x_i < 100$$

- 3) 单峰函数 f_3 是 Quartic 函数:

$$f_3(x) = \sum_{i=1}^n ix_i^4, \quad -1.28 < x_i < 1.28$$

- 4) 单峰函数 f_4 是 Step 函数:

$$f_4(x) = \sum_{i=1}^n (|x_i + 0.5|)^2, \quad -100 < x_i < 100$$

第二类是多峰函数:

- 5) 多峰函数 f_5 是 Schaffer 函数:

$$f_5(x) = \frac{\sin^2 \sqrt{\sum_{i=1}^n x_i^2} - 0.5}{(1 + 0.001 * (\sum_{i=1}^n x_i^2))^2} + 0.5,$$

$$-5.12 < x_i < 5.12$$

- 6) 多峰函数 f_6 是 Griewank 函数:

$$f_6(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1,$$

$$-100 < x_i < 100$$

- 7) 多峰函数 f_7 是 Rastrigin 函数:

$$f_7(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10],$$

$$-5.12 < x_i < 5.12$$

- 8) 多峰函数 f_8 是 Schwefel 函数:

$$f_8(x) = 418.9829n - \sum_{i=1}^n (x_i \sin \sqrt{|x_i|}),$$

$$-500 < x_i < 500$$

八个函数中, 前七个函数的全局最优点为 $(0, \dots, 0)$, 全局最优值为 0. 最后一个函数的全局最优点为 $(420.9687, \dots, 420.9687)$, 全局最优值为 0.

为了更好地理解实验结果, 做了两种实验. 第一种实验在八个函数的一维里进行的, 通过演示函数的轨迹在一维空间随时间步数 t 的动态变化情况, 来判断粒子轨迹是否符合定理 1 和定理 2 的结论. 图 1~图 8 分别是 PSO 算法在计算函数 f_1 到 f_8 时粒子在一维空间的轨迹图, 粒子轨迹是在满足定理 1 和定理 2 的条件下所获得的, 也就是在算法去掉随机性的条件下计算得到的. 参数设置是 $\omega = 0.9, \phi_1 = 1.4, \phi_2 = 1.4$, 这些参数取值满足条件式 (6), 可保证粒子的轨迹收敛. 实验时 p_g 的值都设置为函数各自的最优值, 保持不变的值, 可以使 $f(p_g) < f(p(t))$ 始终满足. $p(t)$ 的值按式 (8) 更新, 初始值设置如下: 速度 $v(0)$ 为 $x(0)$ 的一半, 粒子轨迹初始位置 $x(0)$ 和个体最优初始位置 $p(0)$ 为相同, 但它们的初始值对应不同轨迹图 (见图 1~图 8). 单峰函数 $f_1 \sim f_4$ 粒子轨迹如图 1~图 4, 而多峰函数 $f_5 \sim f_8$ 的轨迹如图 5~图 8. 八个图形中, 粒子轨迹位置 $x(0)$ 和个体最优位置 $p(0)$ 的初始值设置三对值, 其分别对应各图中不同的演示线.

第二类实验选择两个单峰函数 f_1 和 f_3 , 两个多峰函数 f_6 和 f_7 . 实验方法是在 10 维和 20 维空间中, 分别计算四个函数的 PSO 算法粒子离最优

点的距离随算法迭代步数的变化, 其实验结果见图 9~图 16. 每个图形分别演示四种不同初始值的变化情况.

2.3.2 实验结果分析

观察第一类实验, 图 1~图 4 显示的粒子轨迹图表明: 无论初始值 $x(0)$ 取什么值, 计算单峰函数 $f_1 \sim f_4$ 的粒子轨迹都收敛到群体最优点 p_g , 即 $\lim_{t \rightarrow \infty} x(t) = p_g$. 而图 5~图 8 显示的粒子轨迹图表明: 计算多峰函数的粒子轨迹不一定收敛于群体最优点 p_g , 例如, 图 5 在 $x(0) = \pi + 1$ 时的轨迹变化图、图 6 在 $x(0) = 50$ 时的轨迹变化图、图 7 在 $x(0) = 3.12$ 时的轨迹变化图及图 8 在 $x(0) = -300$ 时的轨迹变化图, 粒子轨迹都没有到收敛到最优点 p_g (见图 5~图 8), 而四个图中计算的目标函数都是多峰函数. 因此, 当 PSO 算法计算多峰函数 $f(x)$ 时,

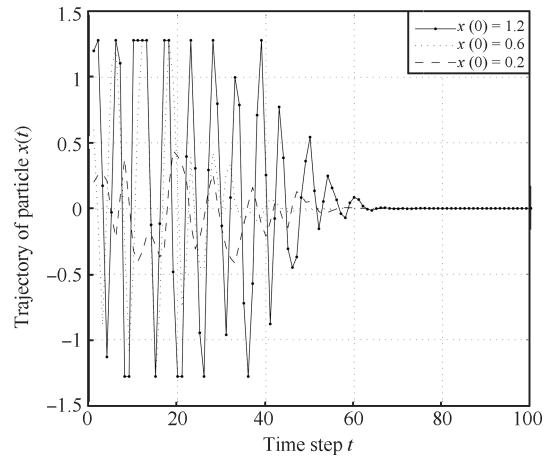


图 3 粒子在 PSO 计算单峰函数 Quartic 时的轨迹
Fig. 3 Trajectories of particle while PSO optimizing unimodal function Quartic

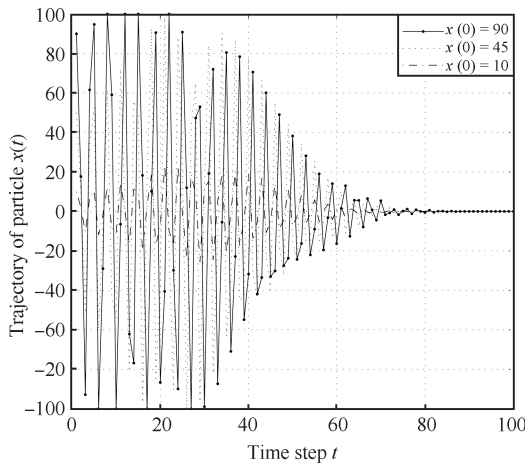


图 1 粒子在 PSO 计算单峰函数 Spherical 时的轨迹
Fig. 1 Trajectories of particle while PSO optimizing unimodal function Spherical

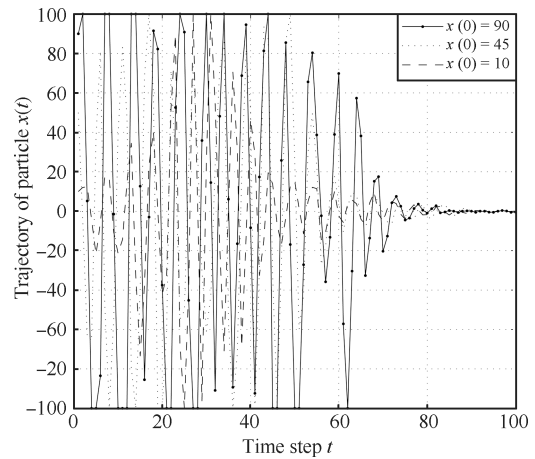


图 4 粒子在 PSO 计算单峰函数 Step 时的轨迹
Fig. 4 Trajectories of particle while PSO optimizing unimodal function Step

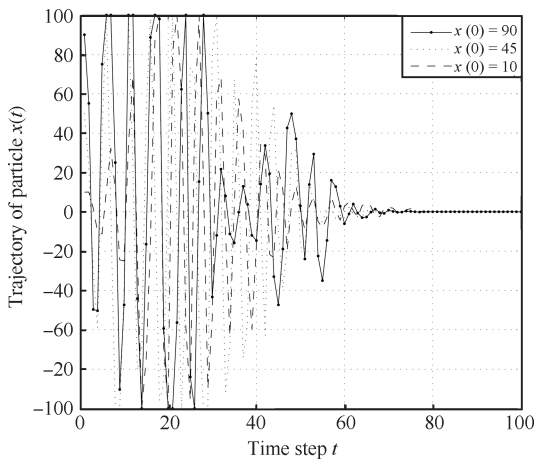


图 2 粒子在 PSO 计算单峰函数 Quadric 时的轨迹
Fig. 2 Trajectories of particle while PSO optimizing unimodal function Quadric

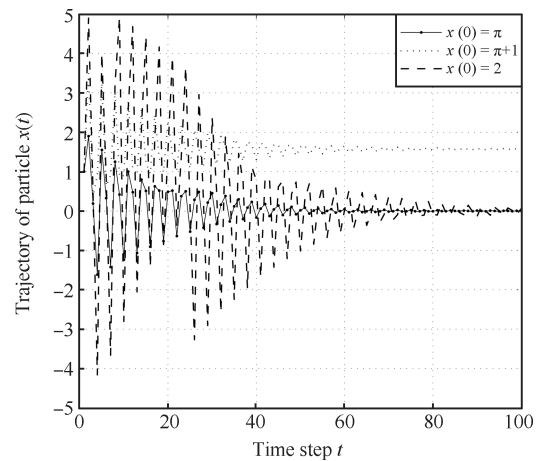


图 5 粒子在 PSO 计算多峰函数 Schaffer 时的轨迹
Fig. 5 Trajectories of particle while PSO optimizing multimodal function Schaffer

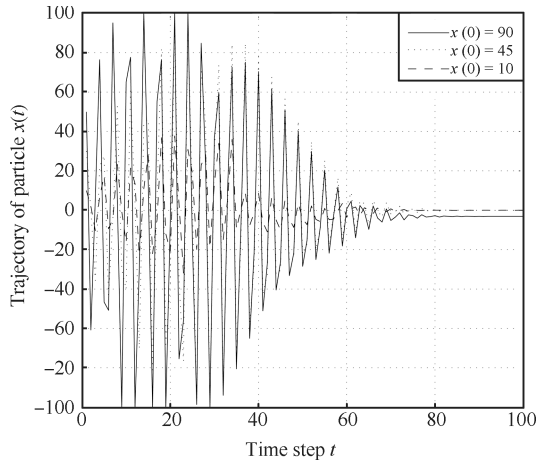


图 6 粒子在 PSO 计算多峰函数 Griewank 时的轨迹
 Fig. 6 Trajectories of particle while PSO optimizing multimodal function Griewank

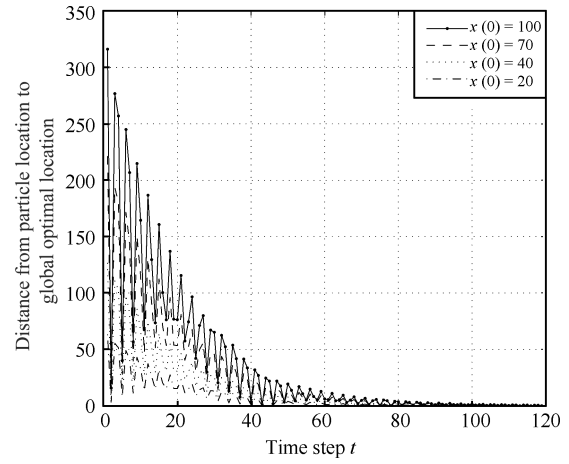


图 9 粒子在计算单峰函数 Spherical 的 10 维空间时的轨迹
 Fig. 9 Trajectories of particle while PSO optimizing unimodal fuction Spherical in 10 dimension

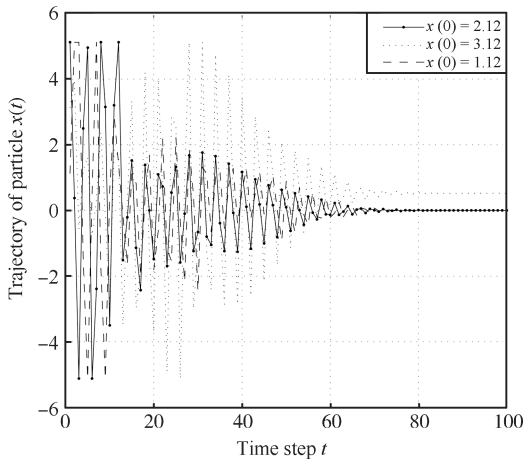


图 7 粒子在 PSO 计算多峰函数 Rastrigin 时的轨迹
 Fig. 7 Trajectories of particle while PSO optimizing multimodal function Rastrigin

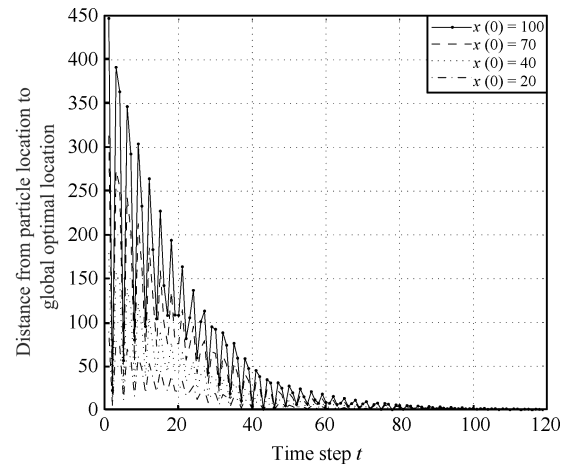


图 10 粒子在计算单峰函数 Spherical 的 20 维空间时的轨迹
 Fig. 10 Trajectories of particle while PSO optimizing unimodal fuction Spherical in 20 dimension

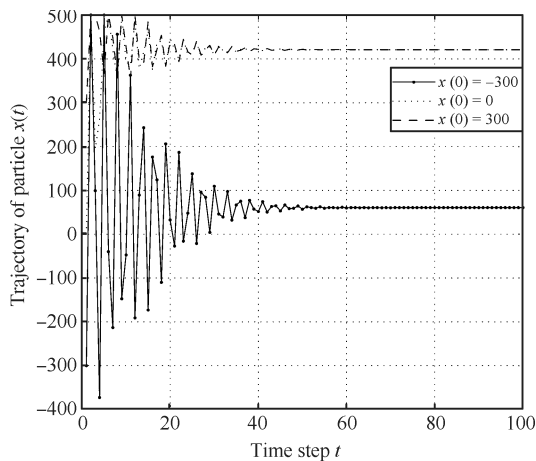


图 8 粒子在 PSO 计算多峰函数 Schwefel 时的轨迹
 Fig. 8 Trajectories of particle while PSO optimizing multimodal fuction Schwefel

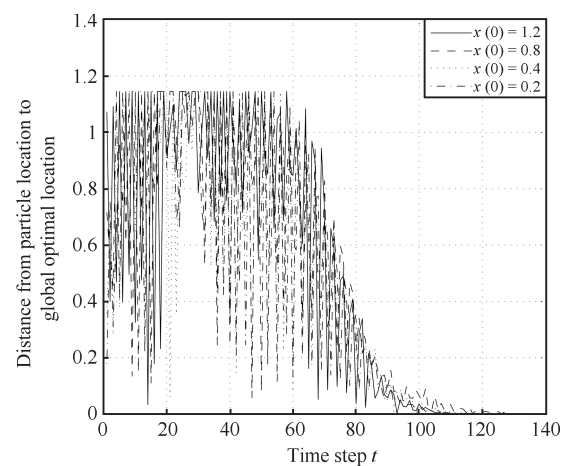


图 11 粒子在计算单峰函数 Quartic 的 10 维空间时的轨迹
 Fig. 11 Trajectories of particle while PSO optimizing unimodal fuction Quartic in 10 dimension

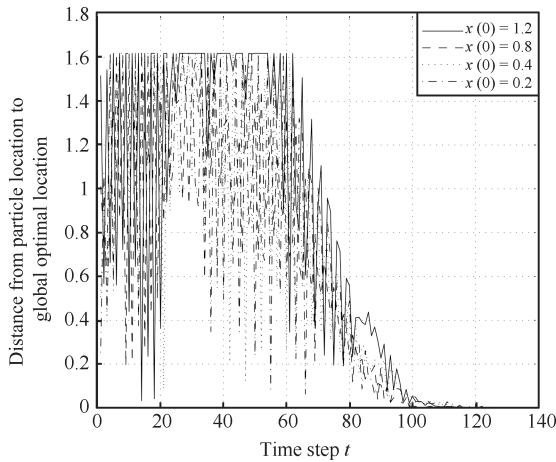


图 12 粒子在计算单峰函数 Quartic 的 20 维空间时的轨迹
Fig. 12 Trajectories of particle while PSO optimizing unimodal fuction Quartil in 20 dimension

粒子轨迹 $x(t)$ 可能收敛于 p_g (如图 5, $x(0) = \pi$ 和 $x(0) = 2$; 如图 6, $x(0) = 30$ 和 $x(0) = 10$; 如图 7, $x(0) = 5.12$ 和 $x(0) = 1.12$; 如图 8, $x(0) = 300$); 也可能不收敛于 p_g (如图 5, $x(0) = \pi + 1$; 如图 6, $x(0) = 50$; 如图 7, $x(0) = 3.12$; 如图 8, $x(0) = -300$ 和 $x(0) = 0$), 这取决于初始位置 $x(0)$ 的值.

观察第二类实验, 图 9 ~ 图 12 是 PSO 算法计算单峰函数 f_1 和 f_3 在 10 维和 20 维时, 其粒子位置离最优位置的距离随迭代步数 t 的变化轨迹. 观察四个图形, 粒子轨迹在几种初值的情况下, 距离最终都会为零, 这说明粒子的轨迹都会收敛于最优位置点. 而图 13 ~ 图 16 是 PSO 算法计算多维函数 f_6 和 f_7 在 10 维和 20 维粒子位置离最优位置的距离随迭代步数的变化轨迹. 根据这四个图, 粒子轨迹在有的初始条件下, 不会收敛于最优点, 例如, 函数 f_6 的初值为 $x(0) = 100$ 或 $x(0) = 20$ 时, 函数 f_7 的初值为 $x(0) = 5$ 或 $x(0) = 4$ 时. 这个实验进一步验证定理 2 的结论, 算法在去掉随机性且计算单峰函数时, 粒子的轨迹最终会收敛全局最优位置, 而计算多峰函数时, 未必会收敛全局最优位置.

针对单峰函数, 图 1 ~ 图 4 的结果验证定理 2 的结论. 对于多峰函数, 定理 2 的结论并不一定成立, 粒子的轨迹并不一定收敛于 p_g . 但是, 根据图 5 ~ 图 8, PSO 算法优化多峰函数时, 粒子的轨迹也可能收敛于 p_g . 因为在定理 2 成立的条件下, 当 p 不变且 $p \neq p_g$ 时, $f((1 - \alpha)p + \alpha p_g)$ 有可能小于 $f(p)$ 也有可能大于 $f(p)$, 这由 p 坐标值来决定. 当算法去掉随机性时, α 值固定不变, $(1 - \alpha)p + \alpha p_g$ 也是不变的. 此时, 如果 $f((1 - \alpha)p + \alpha p_g) > f(p)$ 满足, 则 p 不变. 由于 $f((1 - \alpha)p + \alpha p_g) > f(p)$ 始终满足, p 也就永远不变, 这时 $\lim_{t \rightarrow \infty} x(t) = p_g$ 不成立, 如果 p 不变时, 不会出现 $f((1 - \alpha)p + \alpha p_g) > f(p)$, 则 $\lim_{t \rightarrow \infty} x(t) = p_g$ 一定会成立, 这是图 1 ~ 图 8 中

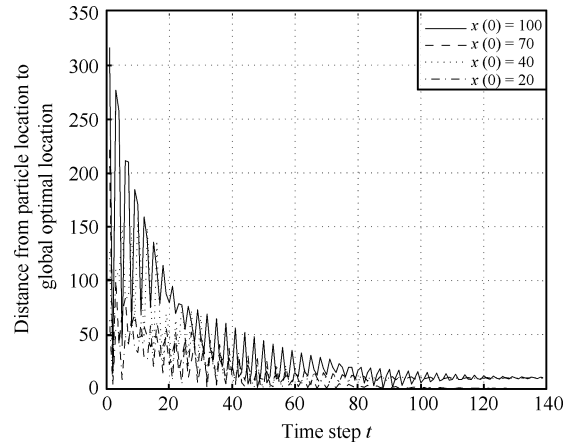


图 13 粒子在计算多峰函数 Griewank 的 10 维空间时的轨迹
Fig. 13 Trajectories of particle while PSO optimizing multimodal fuction Griewank in 10 dimension

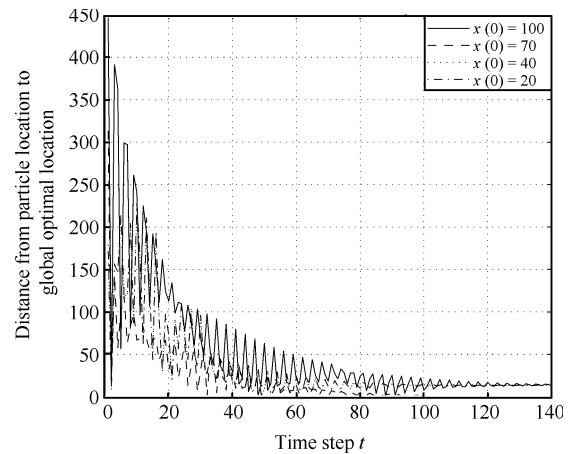


图 14 粒子在计算多峰函数 Griewank 的 20 维空间时的轨迹
Fig. 14 Trajectories of particle while PSO optimizing multimodal fuction Griewank in 20 dimension

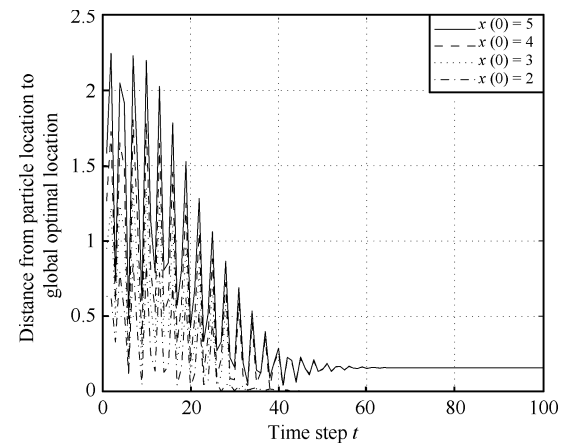


图 15 粒子在计算多峰函数 Rastrigin 的 10 维空间时的轨迹
Fig. 15 Trajectories of particle while PSO optimizing multimodal fuction Rastrigin in 10 dimension

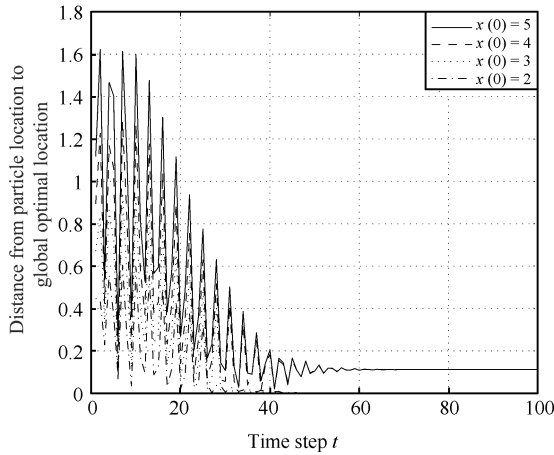


图 16 粒子在计算多峰函数 Rastrigin 的 20 维空间时的轨迹

Fig. 16 Trajectories of particle while PSO optimizing multimodal function Rastrigin in 20 dimension

轨迹图显示结论的直观分析.

3 PSO 算法的随机性对粒子轨迹的影响

3.1 随机性对粒子轨迹结论

PSO 算法是一种带有随机性的启发式算法, 正是因为随机性使算法具有了全局的探索性. 定理 2 成立的条件是去掉了算法的随机性, 但如果把算法的随机性也考虑进去, 则定理 2 的 α 值是随机的. 由于 $\alpha = \phi_1 / (\phi_1 + \phi_2)$, 其中 $\phi_1 = c_1 \times \text{rand}(\cdot)$, $\phi_2 = c_2 \times \text{rand}(\cdot)$, 所以 ϕ_1, ϕ_2 是一个随机数, α 也是一个随机数, 并且在 $[0, 1]$ 之间均匀采样的随机数. 当算法具有随机性时, 粒子轨迹会发生什么变化? 下面证明, 当目标函数 $f(x)$ 无论是单峰函数还是多峰态函数时, 粒子轨迹都会依概率收敛于群体全局最优位置 p_g .

定理 3. 当 PSO 算法具有随机性, 且 $p_g(t)$ 保持不变, 为固定值 p_g ; 但 $p(t)$ 动态变化, 满足式 (8); 则对任意的目标函数 f , 当 $t \rightarrow \infty$ 时, $x(t) \xrightarrow{p} p_g$.

证明. 当 PSO 算法运行时, 若 $p(t)$ 变化, 而 $p_g(t)$ 不变且 $p(t) \neq p_g(t)$, 则此时令 $p_g(t)$ 为 p_g , t 为 t_0 , 所以 $p(t_0) \neq p_g$.

由于 α 在区间 $[0, 1]$ 之间任意随机取值, 所以 $(1 - \alpha)p(t_0) + \alpha p_g$ 在点 $p(t_0)$ 与 p_g 之间的线段随机采样; 令 $l(t_0)$ 表示 p_g 与 $p(t_0)$ 之间的线段; 则 $(1 - \alpha)p(t_0) + \alpha p_g$ 在 $l(t_0)$ 上随机均匀采样.

由于 $f(p_g) < f(p(t_0))$, 所以存在 p_g 的邻域 $N(p_g)$, 使 $x \in l_0 = l(t_0) \cap N(p_g)$, 有 $f(x) < f(p(t_0))$.

对任意 $\varepsilon > 0$, 在线段 $l(t_0)$ 上从 p_g 端取线段 l_1 , 使 $|l_1| = \min\{\varepsilon, |l_0|\}$, 这里 $|l_1|$ 和 $|l_0|$ 分别表线段 l_1 和 l_0 的长度. 因此, 当 $x \in l_1$ 时, $|x - p_g| < \varepsilon$ 成立.

根据 PSO 算法的更新式 (3), 当 $t > t_0$ 时, $p(t)$ 不落入 l_1 的概率为

$$P(p(t) \notin l_1) = \left(1 - \frac{|l_1|}{|l(t_0)|}\right)^{t-t_0}$$

所以, 当 $t > t_0$ 时, $p(t)$ 落入 l_1 的概率为

$$P(p(t) \in l_1) = 1 - \left(1 - \frac{|l_1|}{|l(t_0)|}\right)^{t-t_0}$$

因为 $0 < 1 - \frac{|l_1|}{|l(t_0)|} < 1$, 所以

$$\lim_{t \rightarrow \infty} P(p(t) \in l_1) = \lim_{t \rightarrow \infty} (1 - (1 - \frac{|l_1|}{|l(t_0)|})^{t-t_0}) = 1$$

因为 $p(t) \in l_1$ 推得 $|p(t) - p_g| < \varepsilon$, 所以 $\lim_{t \rightarrow \infty} P(|p(t) - p_g| < \varepsilon) = 1$. 因此, $p(t)$ 依概率收敛于 p_g .

再根据式 (10), $\lim_{t \rightarrow \infty} x(t) = \lim_{t \rightarrow \infty} (1 - a)p(t) + ap_g$, 所以一定会存在某个迭代步 T , 当 $t > T$ 时, 有 $x(t) \in l_1$, 此时 $|x(t) - p_g| < \varepsilon$, 所以 $\lim_{t \rightarrow \infty} P(|x(t) - p_g| < \varepsilon) = 1$. 此时, $x(t)$ 依概率收敛于 p_g .

因此, 在随机条件下, PSO 算法的粒子位置 $x(t)$ 一定依概率收敛于全局最优点 p_g . \square

3.2 随机性实验验证

本节通过仿真实验来验证定理 3 的结论, 与第 2.3 节实验类似, 采用两种实验来分析. 但是, 为了能够体现随机性效果, 本节每个实验的轨迹都是在运行 100 次的平均值. 第一种实验与第 2.3 节的第一种实验类似, 在实验中, 粒子速度的更新式子添加了随机性, 其 α 值为随机量, 与标准 PSO 算法一致. 其他设置与第 2.3 节的第一种实验相同. 实验选择的函数也是相同的八个函数, 其中四个为单峰函数, 另四个为多峰函数. 粒子位置 $x(t)$ 的初始值与第 2.3 节的第一种实验取值相同. 实验演示: 当算法的初始值为不同值时, 粒子位置 $x(t)$ 的轨迹变化状态, 实验结果如图 17~图 24.

第二种实验方法与第 2.3 节的第二种实验类似. 同样, 粒子速度的更新式子添加了随机性, 其 α 值为随机量, 与标准 PSO 算法一致. 实验选择函数 f_1 、函数 f_3 、函数 f_6 和函数 f_7 , 实验方法是在 10 维和 20 维空间中, 其实验结果见图 25~图 32.

对比图 1~图 16 和图 17~图 32, 在这两组实验图形中, 前一组去掉随机性, 后一组保持了随机性. 在前一组图中轨迹不收敛的初始条件在后一组图中变成收敛了 (例如, 图 5~图 8 和图 21~24, 图 13~图 16 和图 29~图 32). 这说明增加随机性给算法带来收敛的效果, 而原来收敛轨迹图, 现在同样还保持收敛.

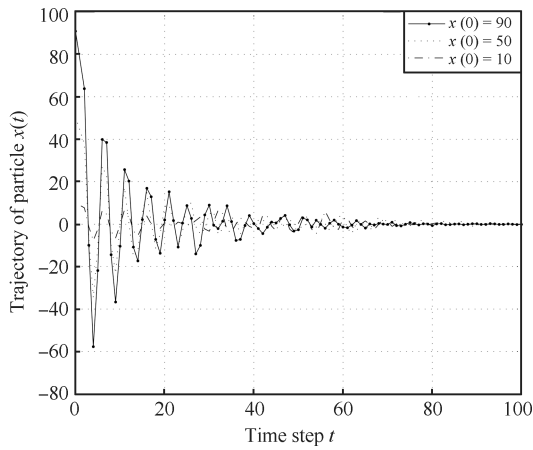


图 17 PSO 带随机性计算单峰函数 Spherical 时的粒子轨迹
 Fig. 17 Trajectories of particle while PSO with random optimizing unimodal function Spherical

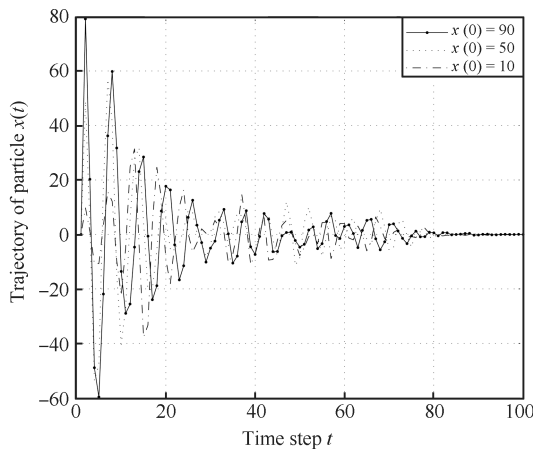


图 18 PSO 带随机性计算单峰函数 Quadric 时的粒子轨迹
 Fig. 18 Trajectories of particle while PSO with random optimizing unimodal function Quadric

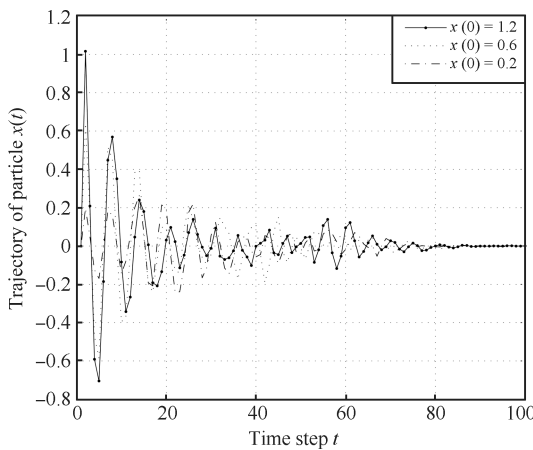


图 19 粒子在 PSO 带随机性计算单峰函数 Quartic 时的轨迹
 Fig. 19 Trajectories of particle while PSO with random optimizing unimodal function Quartic

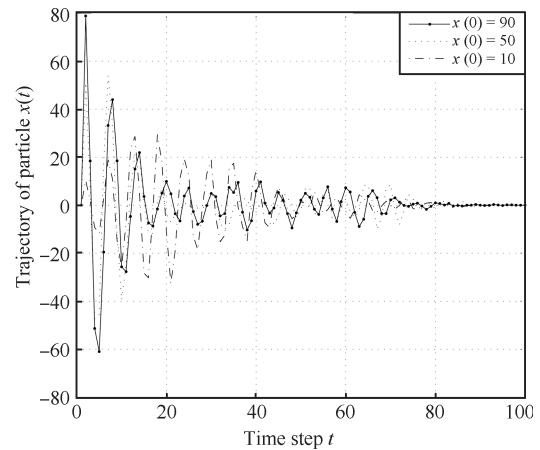


图 20 粒子在 PSO 带随机性计算单峰函数 Step 时的轨迹
 Fig. 20 Trajectories of particle while PSO with random optimizing unimodal function Step

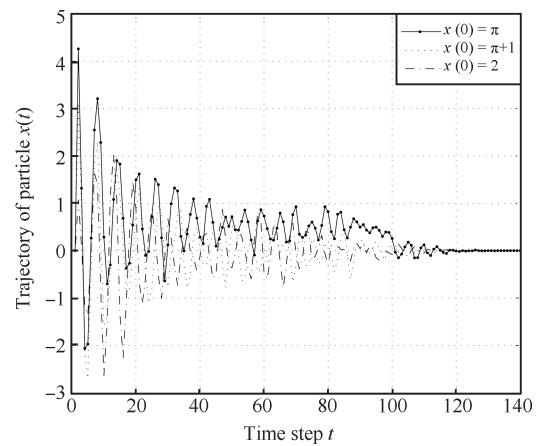


图 21 粒子在 PSO 带随机性计算多峰函数 Schaffel 时的轨迹
 Fig. 21 Trajectories of particle while PSO with random optimizing multimodal function Schaffel

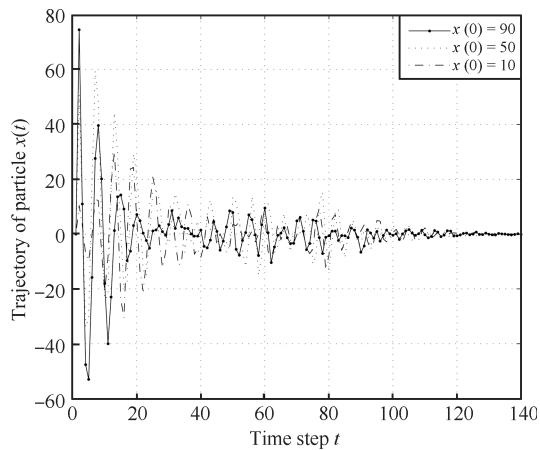


图 22 粒子在 PSO 带随机性计算多峰函数 Griewank 时的轨迹
 Fig. 22 Trajectories of particle while PSO with random optimizing multimodal function Griewank

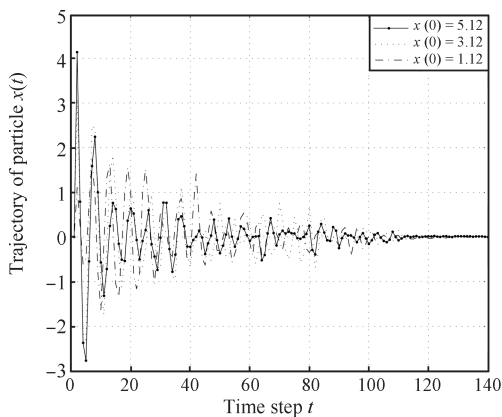


图 23 粒子在 PSO 带随机性计算多峰函数 Rastrigin 时的轨迹

Fig. 23 Trajectories of particle while PSO with random optimizing multimodal function Rastrigin

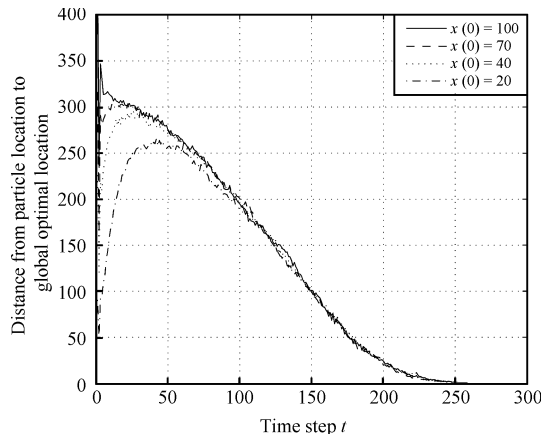


图 26 计算函数 Spherical 在 20 维空间的平均距离的轨迹
Fig. 26 Trajectories of average distance to optimum location for particle while optimizing function Spherical in 20 dimension

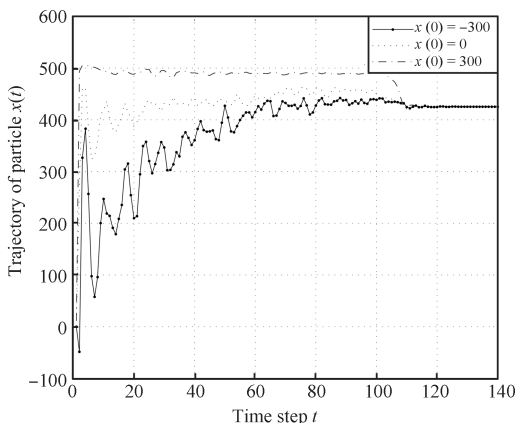


图 24 粒子在 PSO 带随机性计算多峰函数 Schwefel 时的轨迹

Fig. 24 Trajectories of particle while PSO with random optimizing multimodal function Schwefel

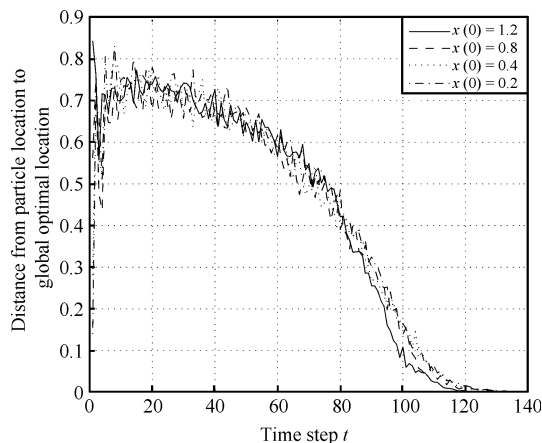


图 27 计算函数 Quartic 在 10 维空间的平均距离的轨迹
Fig. 27 Trajectories of average distance to optimum location for particle while optimizing function Quartic in 10 dimension

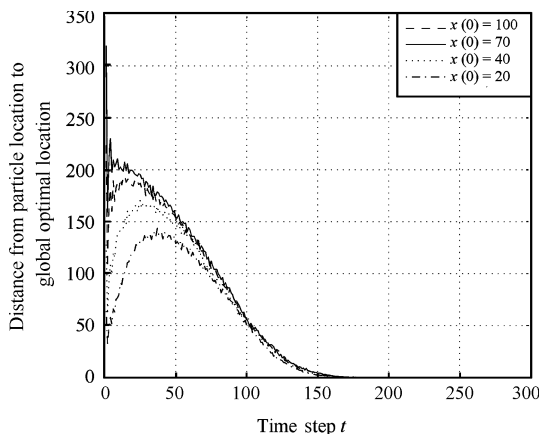


图 25 计算函数 Spherical 在 10 维空间的平均距离的轨迹
Fig. 25 Trajectories of average distance to optimum location for particle while optimizing function Spherical in 10 dimension

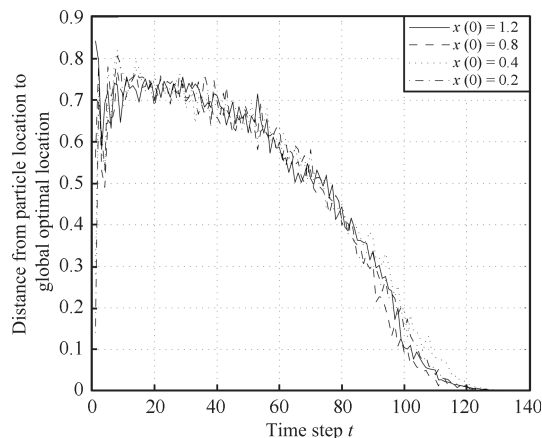


图 28 计算函数 Quartic 在 20 维空间的平均距离的轨迹
Fig. 28 Trajectories of average distance to optimum location for particle while optimizing function Quartic in 20 dimension

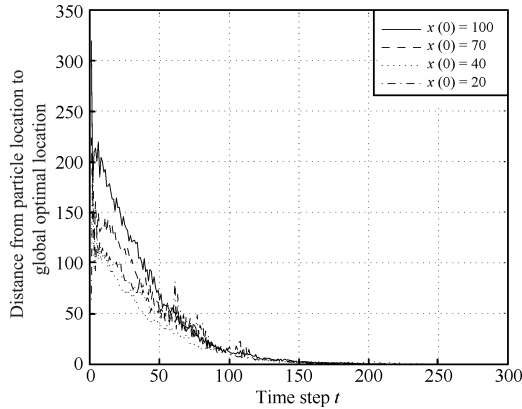


图 29 计算函数 Griewank 在 10 维空间的平均距离的轨迹
 Fig. 29 Trajectories of average distance to optimum location for particle while optimizing function Griewank in 10 dimension

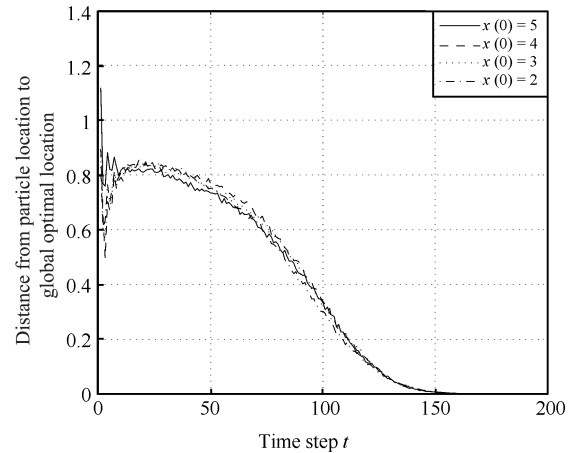


图 32 计算函数 Rastrigin 在 20 维空间的平均距离轨迹
 Fig. 32 Trajectories of average distance to optimum location for particle while optimizing function Rastrigin in 20 dimension

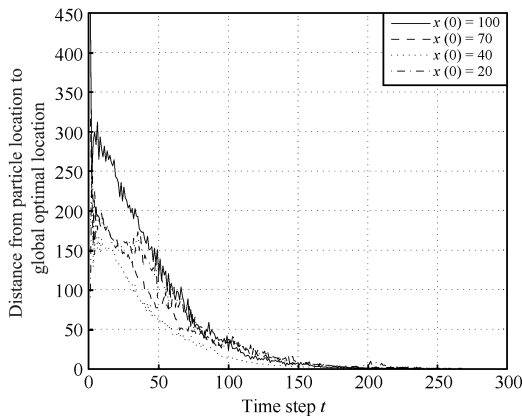


图 30 计算函数 Griewank 在 20 维空间的平均距离的轨迹
 Fig. 30 Trajectories of average distance to optimum location for particle while optimizing function Griewank in 20 dimension

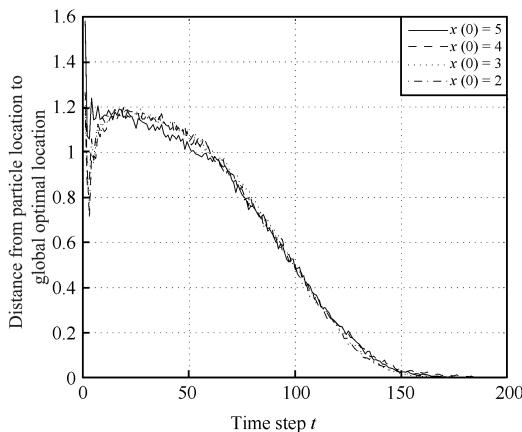


图 31 计算函数 Rastrigin 在 10 维空间的平均距离轨迹
 Fig. 31 Trajectories of average distance to optimum location for particle while optimizing function Rastrigin in 10 dimension

实验说明, 当 α 为随机量时, 算法具有了随机性, 原来不收敛的初始条件却收敛了; 而原来收敛的初始条件仍然收敛, 粒子位置 $x(t)$ 的轨迹总是收敛于粒子群全体最优位置 p_g . 这验证了定理 3 的结论, 说明了随机性能使 PSO 算法无论求解多峰函数还是单峰函数, 都能使算法的粒子位置收敛于群体最优位置.

4 粒子运动轨迹与算法探索性及开拓性的关系

定理 3 的结论不仅说明 PSO 算法的每一个粒子最终会聚集到同一点 p_g . 同样, 定理 2 和定理 3 也能说明 PSO 算法的探索性及开拓性. 所谓探索性指算法的局部搜索能力, 而开拓性指算法的全局搜索能力. 当 PSO 算法参数满足收敛条件时, 算法的粒子初始随机地在空间搜索, 这时粒子全局搜索最优点, 这是算法的开拓性. 而随着算法迭代运行到后期, 随机性使算法靠近群体最优位置 p_g 附近, 交互性使算法只能在 p_g 附近探索最好点, 这时体现了算法探索性.

由定理 3 的收敛性可知, 存在某个 T 值, 当迭代步数 $t > T$ 时, $x_i(t) \in \{p_g \text{ 的邻域}\}$. 当 $t > T$ 时, 粒子轨迹 $x_i(t)$ 在 p_g 的邻域探索, 找 p_g 邻域中最优点, 此时, PSO 算法在行使局部搜索能力. 定理 2 和定理 3 说明粒子轨迹 $x_i(t)$ 只是收敛于群体最优位置 p_g , 不能说明收敛群体最优位置 p_g 附近的局部最优点, 更不能收敛于全局最优点. 这里注意的是, p_g 只是算法目前找到的最优点位置, 不是搜索区域的局部最优点, 更不是全局最优点. 因此, 文献 [13] 的反例说明, PSO 算法并不能保证收敛到 p_g 邻域的局部最优点, 即 PSO 算法并不能保证局部收

敛, 这正与本文分析结论一致.

这里作一个直观分析, 当 PSO 算法运行到迭代步数 T 以后, 某粒子轨迹 $x(t)$ 处于图 33 所示的位置, 此时可以认为粒子群算法正在对单峰函数搜索. 根据速度迭代公式 (2), 若速度 $v(t) < p_g - x(t)$, 则轨迹 $x(t)$ 不会改变全局最优位置 p_g . 如果这个条件在随后迭代步数 t 都能得到满足, 则轨迹 $x(t)$ 只是会收敛 p_g , 但不会改变 p_g 的值. 如果随后粒子群中的每一个粒子在迭代步 T 以后都满足这种情况, 则 p_g 不会收敛到局部最优位置, 这就是 PSO 算法不一定能保证局部收敛的原因.

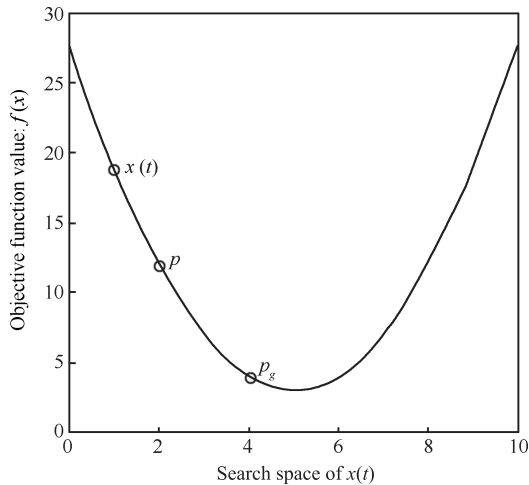


图 33 局部搜索时的粒子轨迹特点

Fig. 33 Trajectories of particle while searching in local field

当 $t < T$, 且目标函数是多峰函数时, p_g 在不同的凹区域动态跳动变化. p_g 的动态跳动变化其实在寻找更好的局部区域, 而此时就是算法在进行全局开拓.

5 结论

本文在分别考虑算法的交互性和随机性的条件下, 进一步分析了 PSO 算法收敛性. 根据优化理论, 在不考虑随机性且计算单峰函数时, PSO 算法的粒子轨迹最终会收敛全局最优位置 p_g ; 而当计算多峰函数时, 粒子最终未必收敛全局最优位置 p_g . 但如果增加随机性, 算法计算的函数无论是单峰函数还是多峰函数, 粒子都会收敛于最优位置 p_g . 本文的分析结果能说明 PSO 算法的随机性及交互性给算法带来的影响, 进一步说明了 PSO 算法如何体现开拓性和探索性, 揭示算法本质性. 与现有的分析方法相比, 本文考虑了算法的随机性和粒子间的交互性, 更进一步考虑了 PSO 算法的因素.

References

1 Eberhart R C, Kennedy J. A new optimizer using particle swarm theory. In: Proceedings of the 6th International

Symposium on Micro Machine and Human Science. Nagoya, Japan: IEEE, 1995. 39–43

2 Kennedy J, Eberhart R C. Particle swarm optimization. In: Proceedings of the 1995 IEEE International Conference on Neural Networks. Perth, WA, Australia: IEEE, 1995. 1942–1948

3 Shi Y H, Eberhart R C. A modified particle swarm optimizer. In: Proceedings of the 1998 IEEE International Conference on Evolutionary Computation. Anchorage, AK, USA: IEEE, 1998. 69–73

4 Poli R, Kennedy J, Blackwell T. Particle swarm optimization: an overview. *Swarm Intelligence*, 2007, 1(1): 33–57

5 Kang Qi, Wang Lei, An Jing, Wu Qi-Di. Approximate dynamic programming based parameter optimization of particle swarm systems. *Acta Automatica Sinica*, 2010, 36(8): 1171–1181

(康琦, 汪镭, 安静, 吴启迪. 基于近似动态规划的微粒群系统参数优化研究. *自动化学报*, 2010, 36(8): 1171–1181)

6 Alfi A. PSO with adaptive mutation and inertia weight and its application in parameter estimation of dynamic systems. *Acta Automatica Sinica*, 2011, 37(5): 541–549

7 Huang Fa-Liang, Xiao Nan-Feng. Discovering overlapping communities based on line graph and PSO. *Acta Automatica Sinica*, 2011, 37(9): 1140–1144

(黄发良, 肖南峰. 基于线图与 PSO 的网络重叠社区发现. *自动化学报*, 2011, 37(9): 1140–1144)

8 Banks A, Vincent J, Anyakoha C. A review of particle swarm optimization. Part II: hybridisation, combinatorial, multicriteria and constrained optimization, and indicative applications. *Natural Computing*, 2008, 7(1): 109–124

9 Ozcan E, Mhoan C K. Analysis of a simple particle swarm optimization system. *Intelligent Engineering Systems Through Artificial Neural Networks*, 1998, 8: 253–258

10 Ozcan E, Mohan C K. Particle swarm optimization: surfing the waves. In: Proceedings of the 1999 Congress on Evolutionary Computation. Washington, DC, USA: IEEE, 1999. 1939–1944

11 Clerc M, Kennedy J. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 2002, 6(1): 58–73

12 Solis F J, Wets R J B. Minimization by random search techniques. *Mathematics of Operations Research*, 1981, 6(1): 19–30

13 Van de Bergh F. An Analysis of Particle Swarm Optimizer [Ph. D. dissertation], University of Pretoria, South Africa, 2002

14 Van de Bergh F, Engelbrecht A P. A convergence proof for the particle swarm optimiser. *Fundamenta Informaticae*, 2010, 105(4): 341–374

15 Blackwell T M. Particle swarms and population diversity. *Soft Computing — A Fusion of Foundations, Methodologies and Applications*, 2005, 9(11): 793–802

16 Trelea I C. The particle swarm optimization algorithm: convergence analysis and parameter selection. *Information Processing Letters*, 2003, 85(6): 317–325

- 17 Ghosh S, Das S, Kundu D, Suresh K, Abraham A. Interparticle communication and search-dynamics of *Ibest* particle swarm optimizers: an analysis. *Information Sciences*, 2012, **182**(1): 156–168
- 18 Fernandez-Martinez J L, Garcia-Gonzalo E. Stochastic stability analysis of the linear continuous and discrete PSO models. *IEEE Transactions on Evolutionary Computation*, 2011, **15**(3): 405–423
- 19 Zheng Y L, Ma L H, Zhang L Y, Qian J X. On the convergence analysis and parameter selection in particle swarm optimization. In: Proceedings of the 2nd International Conference on Machine Learning and Cybernetics. Xi'an, China: IEEE, 2003. 1802–1807
- 20 Zhang L P, Yu H J, Hu S X. Optimal choice of parameters for particle swarm optimization. *Journal of Zhenjian University (Science)*, 2005, **6A**(6): 528–534
- 21 Pan Feng, Chen Jie, Gan Ming-Gang, Cai Tao, Tu Xu Yan. Model analysis of particle swarm optimizer. *Acta Automatica Sinica*, 2006, **32**(3): 368–375
(潘峰, 陈杰, 甘明刚, 蔡涛, 涂序彦. 粒子群优化算法模型分析. 自动化学报, 2006, **32**(3): 368–375)
- 22 Li Ning. Theory Analysis of Particle Swarm Optimization and Its Application Research [Ph.D. dissertation], Huazhong University of Science and Technology, China, 2007
(李宁. 粒子群优化算法的理论分析与应用研究 [博士学位论文], 华中科技大学, 中国, 2007)
- 23 Li Ning, Sun De-Bao, Zou Tong, Qin Yuan-Qing, Wei Yu. An analysis for a particle's trajectory of PSO based on difference equation. *Chinese Journal of Computers*, 2006, **29**(11): 2052–2061
(李宁, 孙德宝, 邹彤, 秦元庆, 尉宇. 基于差分方程的 PSO 算法粒子运动轨迹分析. 计算机学报, 2006, **29**(11): 2052–2061)
- 24 Jin Xin-Lei, Ma Long-Hua, Wu Tie-Jun, Qian Ji-Xin. Convergence analysis of the particle swarm optimization based on stochastic processes. *Acta Automatica Sinica*, 2007, **33**(12): 1263–1268
(金欣磊, 马龙华, 吴铁军, 钱积新. 基于随机过程的 PSO 收敛性分析. 自动化学报, 2007, **33**(12): 1263–1268)
- 25 Ren Zi-Hui, Wang Jian, Gao Yue-Lin. The global convergence analysis of particle swarm optimization algorithm based on Markov chain. *Control Theory and Applications*, 2011, **28**(4): 463–467
(任子晖, 王坚, 高岳林. 马尔科夫链的粒子群优化算法全局收敛性分析. 控制理论与应用, 2011, **28**(4): 463–467)
- 26 Shen Yuan-Xia, Wang Guo-Yin. Probabilistic characteristics analysis of particle swarm optimization and its improved algorithm. *Control and Decision*, 2011, **21**(6): 816–820
(申元霞, 王国胤. 粒子群优化算法的概率特性分析及算法改进. 控制与决策, 2011, **21**(6): 816–820)
- 27 Liu Jian-Hua, Yang Rong-Hua, Sun Shui-Hua. The analysis of binary particle swarm optimization. *Journal of Nanjing University (Natural Sciences)*, 2011, **47**(5): 504–514

(刘建华, 杨荣华, 孙水华. 离散二进制粒子群算法分析. 南京大学学报(自然科学版), 2011, **47**(5): 504–514)

- 28 Horst R, Pardalos P M, Thoai N V [Author], Hao Si-Te [Translator]. *Introduction to Global Optimization*. Beijing: Tsinghua University Press, 2003. 120–207
(Horst R, Pardalos P M, Thoai N V [著], 郝斯特 [译]. 全局优化引论. 北京: 清华大学出版社, 2003. 120–207)



刘建华 博士, 福建工程学院计算机与信息科学系副教授. 主要研究方向为智能计算, 机器学习, 数据挖掘, 智能交通. 本文通信作者.

E-mail: jhliu@fjnu.edu.cn

(LIU Jian-Hua Ph.D., associate professor in the Department of Computer and Information, Fujian University of Technology. His research interest covers intelligence computation, machine learning, data mining, and intelligence transportation. Corresponding author of this paper.)



刘国买 博士, 福建工程学院管理学院教授. 主要研究方向为工程项目风险分析, 智能计算, 智能交通.

E-mail: liuguomai@163.com

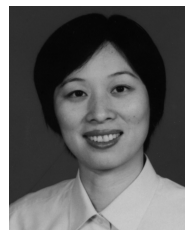
(LIU Guo-Mai Ph.D., professor at the School of Management, Fujian University of Technology. His research interest covers engineering project risk analysis, intelligence computation, and intelligence transportation.)



杨荣华 福建工程学院计算机与信息科学系副教授. 主要研究方向为智能计算, 智能交通. E-mail: yangrh@fjut.edu.cn

(YANG Rong-Hua Associate professor in the Department of Computer and Information, Fujian University of Technology. His research interest covers intelligence computation and intel-

ligence transportation.)



胡文瑜 博士, 福建工程学院计算机与信息科学系副教授. 主要研究方向为智能计算, 数据挖掘.

E-mail: huwenyu@fjut.edu.cn

(HU Wen-Yu Ph.D., associate professor in the Department of Computer and Information, Fujian University of Technology. Her research interest covers intelligence computation and data mining.)