

文章编号: 1001-0920(2012)06-0886-05

函数优化的蜂群算法

刘勇^{1,2}, 马良¹

(1. 上海理工大学管理学院, 上海 200093; 2. 盐城工学院基础教学部, 江苏 盐城 224051)

摘要: 基于蜂群觅食的特点提出一种蜂群算法. 在优化过程中, 每个个体的寻优策略由其以往的寻优经验和整个群体共享的信息决定; 通过定义个体的调整系数和个体与群体间的差异系数实现算法全局探索和局部开发能力的平衡; 给出了算法的实现步骤, 并利用压缩映射定理分析了算法的收敛性. 通过典型的基准函数测试算法的性能, 实验结果表明了算法的优越性.

关键词: 群体智能; 蜂群算法; 优化

中图分类号: TP301.6

文献标识码: A

Bees algorithm for function optimization

LIU Yong^{1,2}, MA Liang¹

(1. School of Management, University of Shanghai for Science and Technology, Shanghai 200093, China, 2. Department of Fundamental Teaching, Yancheng Institute of Technology, Yancheng 224051, China. Correspondent: LIU Yong, E-mail: liuyong.seu@163.com)

Abstract: Based on the foraging characteristics of bees, bees algorithm is proposed. In this method, the searching behavior is guided by the individual previous foraging experience and shared information among the swarm. The algorithm keeps the balance between exploration and exploitation by the individual adjustment factor and the difference factor between the individual and group. The implementation techniques of the algorithm are given and the convergence of the algorithm is analyzed by using compression mapping theorem. The test results show the good performance of the algorithm by optimizing typical benchmark functions.

Key words: swarm intelligence; bees algorithm; optimization

1 引言

近年来, 群体智能的研究正受到人们的广泛关注^[1-2]. 群体智能是指多个简单个体组成的群体, 通过个体之间的交互作用表现出复杂的智能行为. 如蚁群算法是模拟实际蚁群觅食的过程^[3], 微粒群算法源于对鸟群群体运动行为的研究^[4]. 这些方法在许多领域中得到了成功的应用.

自然界的蜂群是一种典型的群居性昆虫, 其行为表现出系统性、自组织性和正反馈性等特征. 利用蜂群智能求解优化问题主要基于其蜂后概念、繁殖机理和觅食原理3个方面. 早期的研究主要利用蜂后保存优良基因的思想改进遗传算法^[5-7], 并在实际应用中取得较好的优化效果. 对于基于繁殖原理的蜂群算法, Abbass^[8-10]提出了一种求解3-SAT(可满足性)问题的蜜蜂繁殖优化算法; Bozorg^[11]给出了一种

优化水资源管理问题的蜂群繁殖优化算法. 随后不少学者对算法进行了改进, 并用于连续优化^[12]、动态规划^[13]、分区和调度^[14]以及数据挖掘^[15-16]等方面. 但模拟蜂群的觅食行为用于函数优化的研究还处于初始阶段. 文献[17]提出一种虚拟蜂群算法, 仅用于求解两个变量的函数优化问题; 为求解多变量函数优化问题, [18]提出一种蜜蜂算法; [19]提出一种人工蜂群算法. 这些算法各有特色, 但均未考虑蜂群在寻优过程中搜索行为的自适应调整, 无法保证全局探索和局部开发能力的平衡, 易陷入局部极值和早熟收敛.

本文基于蜂群觅食的特点, 提出一种蜂群优化算法. 蜂群寻找最好食物源的过程类似于优化算法探寻最优解的过程. 在优化过程中, 蜜蜂的搜索行为由其以往寻优经验和共享的群体信息确定. 利用压缩映射定理对算法的收敛性进行分析, 采用基准测试函数对

收稿日期: 2010-12-01; 修回日期: 2011-03-31.

基金项目: 国家自然科学基金项目(70871081); 上海市重点学科建设项目(S30504).

作者简介: 刘勇(1982—), 男, 博士生, 从事智能优化、系统工程等研究; 马良(1964—), 男, 教授, 博士生导师, 从事智能优化、系统工程等研究.

算法进行测试, 并与 PSO 算法进行比较, 所得结果验证了算法的可行性和有效性.

2 蜂群算法

群居性生物的个体行为非常简单, 但由这些个体组成的群体通过个体之间的交互作用可表现出复杂的智能行为^[1-4]. 群体智能算法的基本思想是模拟自然界生物的群体行为构造优化算法, 智能的寻优方法是通过整个群体的行为特征来实现的. 本文受蜂群觅食行为的启发, 提出一种蜂群算法.

蜂群觅食是一种典型的群体智能行为. 根据昆虫学家的长期观察研究发现^[20-21], 每只蜜蜂的智能并不高, 没有统一的指挥中心, 却能协同合作, 蜂群在没有任何可见提示下找到质量最好的食物源, 并能随环境而变化, 适应性地搜索新的最优食物源, 整个蜂群体现出超出单个个体的智能. 在觅食过程中, 蜜蜂能记住形状、颜色和气味等食物特征, 逐步形成自己的搜索经验. 在群体智能形成过程中, 蜜蜂之间的信息交流是最重要的. 蜜蜂通过一种特殊的舞蹈动作将有关食物的质量和位置等信息告诉同伴. 通过这种信息共享方式, 蜜蜂会停止采集质量较差的食物, 并根据共享的信息调整搜索策略, 朝着质量好的食物源方向移动. 通过信息的不断交流, 质量差的食物源被放弃, 质量好的食物源被保存下来. 在这种机制下, 最终整个蜂群聚集在质量最好的食物源上.

蜂群寻找最好食物源的过程相当于优化算法探寻最优解的过程. 在蜂群算法中, 蜜蜂的位置即为食物的位置, 并用优化问题的解表示, 食物的质量则用目标函数值表示. $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ 表示第 i 只蜜蜂的位置向量. 第 i 只蜜蜂在搜索解空间时, 保存其搜索到的最优位置 $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})$. 其中: $i = 1, 2, \dots, N$; N 为群体规模; D 为向量维数. 共享的信息用平均最优位置 p_m 表示. 在寻优过程中, 蜜蜂 i 根据自身经验和共享的信息动态调整搜索方向, 位置更新方程为

$$p_m = \left(\sum_{i=1}^N p_{i1} / N, \sum_{i=1}^N p_{i2} / N, \dots, \sum_{i=1}^N p_{iD} / N \right), \quad (1)$$

$$x_{id} = \alpha x_{id} + \beta (p_{md} - x_{id}), \quad (2)$$

$$i = 1, 2, \dots, N, d = 1, 2, \dots, D.$$

其中: α 为个体调整系数, β 为个体和群体的差异系数, 且 $\alpha + \beta = c$ ($\alpha > 0, \beta > 0, c$ 为常数); $(p_{mi} - x_{id})$ 为平均最优位置与当前个体位置的距离. 在优化过程中, 通过自身经验的积累和信息的共享, 个体寻优能力逐渐增强, 个体和群体的差别越来越小. 相应地, α 值逐渐增大, β 值逐渐减小.

蜂群算法的主要步骤可描述如下:

Step 1: 设置算法参数, 初始化群体;

Step 2: 计算各蜜蜂的目标函数值, 并记录当前的最好解;

Step 3: 按式 (1) 更新平均最优位置;

Step 4: 按式 (2) 计算每只蜜蜂的位置;

Step 5: 对每只蜜蜂, 将其位置与所经历过的最好位置进行比较, 若较好, 则将其作为当前的最好位置;

Step 6: 若当前迭代次数小于预设最大迭代次数且无退化行为 (即找到的都是相同的解), 则转 Step 2;

Step 7: 输出目前的最好解.

3 收敛性分析

蜂群算法的优化过程可表示为 $x_{k+1} = f(x_k)$, 是一个迭代过程. 若存在一个点 x^* , 使得 $x^* = f(x^*)$, 则算法收敛.

3.1 预备知识

定义 1^[22-23] 设 X 是非空集合, 对于 X 中的任意两个元素 x 和 y , 按某一法则都对应于唯一的实数 $d(x, y)$, 而且满足下述 3 个公理:

1) 非负性: $d(x, y) \geq 0$; $d(x, y) = 0$ 当且仅当 $x = y$;

2) 对称性: $d(x, y) = d(y, x)$;

3) 三角不等式: 对于任意的 $x, y, z \in X$, 恒有

$$d(x, y) \leq d(x, z) + d(z, y).$$

则称 d 为 X 上的度量, 称 (X, d) 为度量空间.

定义 2^[22-23] 度量空间 (X, d) 中的点列 $\{x_n\}$ (称为 Cauchy 点列或基本点列) 是指: 对于任意给定的 $\varepsilon > 0$, 始终存在自然数 N , 使得当 $m, n > N$ 时, $d(x_m, x_n) < \varepsilon$; 或当 $m, n \rightarrow +\infty$ 时, $d(x_m, x_n) \rightarrow 0$. 若度量空间 (X, d) 中每一个 Cauchy 点列都收敛, 则称 (X, d) 为完备的度量空间.

定义 3^[22-23] 设 (X, d) 是度量空间, f 是 X 到 X 的映射. 若一个常数 θ ($0 \leq \theta < 1$) 使得对于一切 $x, y \in X$, 总有

$$d(fx, fy) \leq \theta d(x, y), \quad (3)$$

则称 f 是 X 上的一个压缩映射.

定理 1 (Banach 压缩映射定理)^[22-23] 设 (X, d) 是完备的度量空间, f 是 $X \rightarrow X$ 上的一个压缩映射, 则 f 有一个唯一的不动点 $x^* \in X$, 使得对于任意 $x_0 \in X$, 满足

$$x^* = \lim_{i \rightarrow \infty} f^i(x_0). \quad (4)$$

其中: $f^0(x_0) = x_0, f^{i+1}(x_0) = f(f^i(x_0))$.

3.2 收敛性证明

在算法中, 定义平均最优位置

$$p_m = \left(\sum_{i=1}^N p_{i1}/N, \sum_{i=1}^N p_{i2}/N, \dots, \sum_{i=1}^N p_{iD}/N \right),$$

其中 p_i 为第 i 个个体的迄今最优的位置. 考虑算法每次迭代情况, 设 P^k 为第 k 次迭代时所有个体最优位置的集合, 即 $P^k = \{p_1^k, p_2^k, \dots, p_N^k\}$, 所有 P^k 的集合记为 S . 定义度量 $d: S \times S \rightarrow R$ 为

$$d(P^m, P^n) = \begin{cases} \sum_{p_i^m \in P^m} |f(p_i^m) - M| + \sum_{p_i^n \in P^n} |f(p_i^n) - M|, \\ P^m \neq P^n; \\ 0, P^m = P^n. \end{cases} \quad (5)$$

其中 M 为目标函数的下界, 对于一个最小值问题, M 必存在. 因此可以得出:

1) $d(P^m, P^n) \geq 0; d(P^m, P^n) = 0$ 当且仅当 $P^m = P^n$;

2) $d(P^m, P^n) = d(P^n, P^m)$;

3) $d(P^m, P^n) =$

$$\begin{aligned} & \sum_{p_i^m \in P^m} |f(p_i^m) - M| + \sum_{p_i^n \in P^n} |f(p_i^n) - M| \leq \\ & \sum_{p_i^m \in P^m} |f(p_i^m) - M| + \sum_{p_i^k \in P^k} |f(p_i^k) - M| + \\ & \sum_{p_i^k \in P^k} |f(p_i^k) - M| + \sum_{p_i^n \in P^n} |f(p_i^n) - M| = \\ & d(P^m, P^k) + d(P^k, P^n). \end{aligned}$$

可见 (S, d) 是一个度量空间. 因 S 只含有有限数量的个体, 故对于任何群体的 Cauchy 序列 P^1, P^2, \dots 存在 k , 当 $n > k$ 时有 $P^n = P^k$, 即对于所有 Cauchy 序列 $P^i (i \rightarrow \infty)$ 都收敛^[23-24], 因此 (S, d) 是一个完备的度量空间.

将算法的操作视为一映射 $f: S \rightarrow S$, 根据算法的设计, 在计算平均最优位置时, 对每个个体的最优位置都进行比较更新. 每迭代一次均可能产生比上一次更优的位置, 在整个求解过程中每个个体各代最优位置函数值的序列形成一个非增序列, 即

$$f(p_i^{k+1}) \leq f(p_i^k) \leq f(p_i^{k-1}), \quad (6)$$

或

$$f(f(f(p_i^{k-1}))) \leq f(f(p_i^{k-1})) \leq f(p_i^{k-1}), \quad (7)$$

其中 $f(p_i^k)$ 为第 k 次迭代时第 i 个个体最优位置 p_i 的函数值. 从而对于映射 f , 有

$$\begin{aligned} d(f(P^m), f(P^n)) &= \\ & \sum_{p_i^m \in P^m} |f(f(p_i^m)) - M| + \sum_{p_i^n \in P^n} |f(f(p_i^n)) - M| \leq \\ & \sum_{p_i^m \in P^m} |f(p_i^m) - M| + \sum_{p_i^n \in P^n} |f(p_i^n) - M| \leq \end{aligned}$$

$$\varepsilon d(P^m, P^n), 0 < \varepsilon < 1. \quad (8)$$

所以 f 是一个压缩映射.

综上所述, 算法满足 Banach 压缩映射定理的条件是: (S, d) 是一个完备的度量空间, f 是一个压缩映射, 该算法是收敛的. 当 S 中所有个体都有相同的值 (最优值) 时, 即得到唯一的不动点. 当优化的目标函数有多个最优解时, 可将原有的度量空间分解成多个小的度量空间, 使每个小的度量空间满足不动点定理的要求, 在这种情况下, 算法收敛到其中一个可能的最优解.

4 仿真实验

仿真环境为 Windows XP 操作系统, Intel 处理器, 2.40 GHz, 1 G 内存, 仿真软件 Matlab7.4. 选用 4 个典型的基准测试函数来验证算法的性能, 并与 PSO 算法进行比较. 4 个测试函数分别是:

1) Sphere Model

$$f_1(x) = \sum_{i=1}^D x_i^2, \\ -100 \leq x_i \leq 100, \min(f_1) = f_1(0, \dots, 0) = 0;$$

2) Ackley function

$$f_2(x) = -20 \exp \left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2} \right) - \\ \exp \left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi x_i) \right) + 20 + e, \\ -32 \leq x_i \leq 32, \min(f_2) = f_2(0, \dots, 0) = 0;$$

3) Griewank function

$$f_3(x) = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos \left(\frac{x_i}{\sqrt{i}} \right) + 1, \\ -600 \leq x_i \leq 600, \min(f_3) = f_3(0, \dots, 0) = 0;$$

4) Rastrigin function

$$f_4(x) = \sum_{i=1}^D (x_i^2 - 10 \cos(2\pi x_i) + 10), \\ -5.12 \leq x_i \leq 5.12, \min(f_4) = f_4(0, \dots, 0) = 0.$$

其中: Sphere 为单模函数, 在定义域内只有全局最优解, 没有局部极值, 是测试群体智能算法的经典函数; Ackley, Griewank 和 Rastrigin 为多模函数, 在定义域内有多个局部最优解, 极难优化, 常用来测试算法的探索和开发能力.

为便于比较算法的性能, 测试算法的群体规模设为 50, 最大迭代次数为 500. 蜂群算法的其他参数设置按目前通用的方法根据实验情况确定, 本文算法的参数设置为 $c = 1$; 个体和群体的差异系数 β 是迭代次数的函数, 从 0.95 线性递减到 0 且满足 $\beta(k) = -0.95(k/\max it) + 0.95$ (k 为当前迭代次数, $\max it$ 为最大迭代次数); 个体调整系数 $\alpha(k) = c - \beta(k)$.

PSO 算法包括 3 种类型: 基本粒子群算法 (PSO)^[25], 量子粒子群算法 (QPSO)^[26]和动态多子群粒子群算法 (DMSPSO)^[27]. 实验中 3 种算法的参数设置采用相应文献中推荐的参数值. 4 种测试函数的维数设为 30, 表 1 给出了每种函数独立运行 30 次测试结果的统计数据.

表 1 函数优化测试结果

函数	算法	最好值	最差值	平均值	标准差
Sphere	PSO	1.861 9e-011	7.612 4e-008	1.273 6e-008	1.777 5e-008
	QPSO	4.362 2e-006	5.703 9e-004	1.220 9e-004	1.553 6e-004
	DMSPSO	1.296 3e-015	2.122 9e-015	1.675 9e-015	1.134 0e-016
	BA	5.981 7e-234	5.841 1e-232	1.991 6e-232	0
Ackley	PSO	1.494 3e-006	8.778 9e-005	1.404 7e-005	1.633 1e-005
	QPSO	6.350 8e-004	0.009 4	0.002 2	0.001 7
	DMSPSO	1.055 1e-010	2.525 9e-008	2.620 6e-009	6.861 8e-009
	BA	8.881 8e-016	8.881 8e-016	8.881 8e-016	0
Griewank	PSO	2.355 9e-009	0.044 2	0.009 7	0.010 7
	QPSO	5.439 0e-006	0.029 7	0.008 9	0.009 8
	DMSPSO	9.465 8e-013	3.021 4e-011	7.087 8e-012	6.432 5e-012
	BA	0	0	0	0
Rastrigin	PSO	2.069 5e-010	3.374 4e-005	1.193 2e-006	6.152 4e-006
	QPSO	15.944 4	92.696 4	32.063 6	13.220 9
	DMSPSO	14.924 4	57.707 5	38.040 6	10.499 4
	BA	0	0	0	0

由表 1 分析可知, 对于所有测试的函数, 无论是最优值、最差值还是平均值、标准差, 蜂群算法均获得了较好的结果. 与 3 种 PSO 算法相比, 本文提出的蜂群算法具有更高的寻优精度, 更好的稳定性, 能有效求解复杂的函数优化问题.

为进一步分析算法的优化性能, 图 1~图 4 给出了各算法对 4 个测试函数的收敛曲线图(为便于比较,

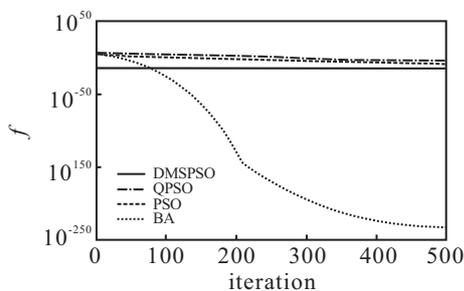


图 1 Sphere 寻优曲线

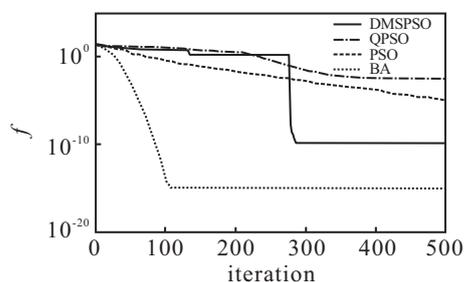


图 2 Ackley 寻优曲线

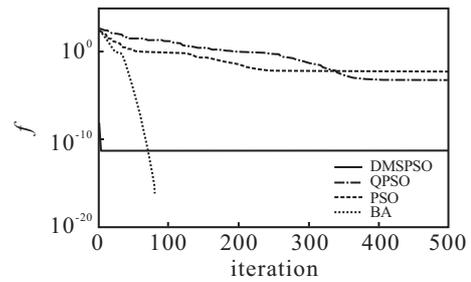


图 3 Griewank 寻优曲线

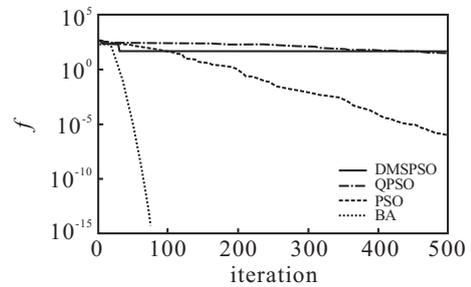


图 4 Rastrigin 寻优曲线

函数值用以 10 为底的对数表示). 由图可知, 蜂群算法不仅在收敛精度方面表现出色, 而且在收敛速度方面也同样表现突出. 分析原因, 在算法中定义了平均最优位置 p_m 表示共享的信息, 比单个的个体最优位置包含了更多的有关最优位置的信息, 利用了群体中所有个体搜索到的信息. 通过个体调整系数 α 和个体与群体的差异系数 β 的变化, 保证每个个体可动态地根据自身的搜索经验和共享的信息调整寻优位置; 同时通过系数 α 和 β 的调整, 实现了算法全局探索和局部开发能力的平衡. 在这种机制下, 算法能有效避免早熟收敛, 随着搜索过程的继续逐渐逼近全局最优解.

5 结 论

本文根据蜂群觅食的特点提出一种蜂群算法. 详细地分析了算法的寻优过程, 并利用压缩映射定理证明了算法的收敛性. 采用典型的测试函数进行仿真实验, 所得结果表明, 算法在收敛精度和收敛速度方面均有优异的表现, 具有良好的优化性能.

参考文献(References)

- [1] Blum C, Merkle D. Swarm intelligence: Introduction and applications[M]. Berlin: Springer-Verlag, 2008.
- [2] Peng F, Tang K, Chen G, et al. Population-based algorithm portfolios for numerical optimization[J]. IEEE Trans on Evolutionary Computation, 2010, 14(5): 782-800.
- [3] Dorigo M, Stützle T. Ant colony optimization: Overview and recent advances[J]. Handbook of Metaheuristics, 2010, 146: 227-263.
- [4] Olsson A E. Particle swarm optimization: Theory, techniques and applications[M]. New York: Nova Science Publishers, 2011.

- [5] Sung H J. Queen-bee evolution for genetic algorithms[J]. *Electronic Letters*, 2003, 39(6): 575-576.
- [6] Kara A. Imitation of bee reproduction as a crossover operator in genetic algorithms[C]. *Lecture Notes in Computer Science*. Springer, 2004, 3157: 1015-1016.
- [7] Azeem M F, Saad A M. Modified queen bee evolution based genetic algorithm for tuning of scaling factors of fuzzy knowledge base controller[C]. *IEEE INDICON 2004 Proc of the India Annual Conf. Kharagpur, 2004*: 299-303.
- [8] Abbass H A. A single queen single worker honey-bees approach to 3-SAT[C]. *The Genetic and Evolutionary Computation Conf. San Francisco, 2001*: 807-814.
- [9] Abbass H A, Teo J. A true annealing approach to the marriage in honey-bees optimization algorithm[C]. *Proc of the Inaugural Workshop on Artificial Life. Adelaide, 2011*: 1-14.
- [10] Abbass H A. MBO: Marriage in honey bees optimization a haplometrosis polygynous swarming approach[C]. *Proc of the Congress on Evolutionary Computation. Seoul, 2001*: 207-214.
- [11] Bozorg H O, Afshar A, Mariano M A. Honey-bees mating optimization(HBMO) algorithm: A new heuristic approach for water resources optimization[J]. *Water Resources Management*, 2006, 20(5): 661-680.
- [12] Afshar A, Bozorg Haddad O, Marino M A, et al. Honey-bee mating optimization(HBMO) algorithm for optimal reservoir operation[J]. *J of the Franklin Institute*, 2007, 344(5): 452-462.
- [13] Chang H S. Converging marriage in honey-bees optimization and application to stochastic dynamic programming[J]. *J of Global Optimization*, 2006, 35(3): 423-441.
- [14] Koudil M, Benatchba K, Tarabet A, et al. Using artificial bees to solve partitioning and scheduling problems in code sign[J]. *Applied Mathematics and Computation*, 2007, 186(2): 1710-1722.
- [15] Benatchba K, Admane L, Koudil M. Using bees to solve a data mining problem expressed as a max-sat one[C]. *Proc of IWINAC'2005. Canary Islands, 2005*: 212-220.
- [16] Fathian M, Amiri B, Maroosi A. Application of honey-bee mating optimization algorithm on clustering[J]. *Applied Mathematics and Computation*, 2007, 190(2): 1502-1513.
- [17] Yang X S. Engineering optimizations via nature-inspired virtual bee algorithms[C]. *Lecture Notes in Computer Science*. Springer, 2005, 3562: 317-323.
- [18] Pham D T, Ghanbarzadeh A, Koc E, et al. The bees algorithm — A novel tool for complex optimization problems[C]. *Proc of 2nd IPROMS Virtual Int Conf on Intelligent Production Machines and Systems. Wales, 2006*: 254-461.
- [19] Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony(abc) algorithm[J]. *J of Global Optimization*, 2007, 39(3): 459-471.
- [20] Menzel R, Erber J. Learning and memory in bees[J]. *Scientific American*, 1978, 239(1): 80-87.
- [21] Whitfield J. Animal behavior: The wisdom of the bees[J]. *Nature*, 2010, 467(7316): 658-659.
- [22] 梁永锦. 泛函分析讲义[M]. 北京: 科学出版社, 2011. (Liang Y J. *Functional analysis lecture*[M]. Beijing: Science Press, 2011.)
- [23] 刘培德. 泛函分析基础[M]. 北京: 科学出版社, 2010. (Liu P D. *Fundamental functional analysis*[M]. Beijing: Science Press, 2010.)
- [24] Michalewicz Z. Genetic algorithm + data structure = evolution programs[M]. Berlin: Springer-Verlag, 1999.
- [25] Eberhart R C, Shi Y H. Comparing inertia weights and constriction factors in particle swarm optimization[C]. *Proc of the Congress on Evolutionary Computing. La Jolla, 2000*: 84-89.
- [26] Sun J, Xu W B, Feng B. A global search strategy of quantum-behaved particle swarm optimization[C]. *Proc of the 2004 IEEE Conf on Cybernetics and Intelligent Systems. Singapore, 2004*: 291-294.
- [27] Liang J J, Suganthan P N. Dynamic multi-swarm particle swarm optimizer[C]. *Proc of the 2005 IEEE Conf on Swarm Intelligence Symposium. Singapore, 2005*: 124-129.