

群体智能优化算法

王艳玲, 李龙澍, 胡哲

(安徽大学 计算机科学与技术学院, 安徽 合肥 230039)

摘要: 群体智能优化算法利用群体的优势, 在没有集中控制并且不提供全局模型的前提下, 为寻找复杂的分布式问题的解决方案提供了基础。介绍了两种群体智能算法模型: 蚁群算法模型和粒子群算法模型, 研究了两种算法的原理机制、基本模型、流程实现、改进思想和方法; 通过仿真把蚁群算法与其他启发式算法的计算结果作对比, 验证了蚁群算法具有很强的发现较好解的能力, 不容易陷入局部最优; 微粒群算法保留了基于种群的、并行的全局搜索策略, 采用简单的速度-位移模型操作, 在实际应用中取得了较高的成功率。

关键词: 群体智能; 蚁群算法; 粒子群算法; 启发式算法

中图分类号: TP18

文献标识码: A

文章编号: 1673-629X(2008)08-0114-04

Swarm Intelligence Optimization Algorithm

WANG Yan-ling, LI Long-shu, HU Zhe

(School of Computer Science and Tech., Anhui Univ., Hefei 230039, China)

Abstract: By the use of groups' advantages, in the absence of centralized control and without providing the overall model situation, swarm intelligence optimum algorithm provides the foundation on finding complex distributed solutions to the problem. Introduces the two swarm intelligence algorithm models: ant colony algorithm model and the particle swarm algorithm model, researches on principle mechanism, the basic model, process realization and improved ideas and methods; and by comparing the calculation results of the ant colony algorithm and other heuristic algorithm through the simulation, proved that the ant colony algorithm has a strong ability to find better solutions, and is not easy for a local optimum. The algorithm which based on the population of reservations, parallel global search strategy, using simply speed-displacement model operation, has been made in a higher success rate in practical application.

Key words: swarm intelligence; ant colony optimization algorithm; particle swarm optimization; heuristic algorithm

0 引言

受社会性昆虫行为的启发, 计算机工作者通过对社会性昆虫的模拟产生了一系列对于传统问题的新的解决方法, 这些研究就是群体智能的研究。群体智能中的群体指的是“一组相互之间可以进行直接通信或者间接通信(通过改变局部环境)的主体, 这组主体能够合作进行分布问题求解”。而所谓群体智能指的是“无智能的主体通过合作表现出智能行为的特性”。群体智能在没有集中控制并且不提供全局模型的前提下, 为寻找复杂的分布式问题的解决方案提供了基础。

在计算智能领域有两种基于群体智能的算法: 蚁群算法(Ant Colony Optimization, ACO)^[1,2]和粒子群算

法(Particle Swarm Optimization, PSO)^[3]。前者是对蚂蚁群落食物采集过程的模拟, 已经成功运用在很多离散优化问题上; 后者是源于对鸟群捕食行为的研究, 算法简单容易实现并且没有许多参数需要调整, 目前已广泛应用于函数优化、神经网络训练、模糊系统控制以及其他遗传算法的应用领域。

1 蚁群优化算法

1.1 蚁群算法原理

受蚂蚁觅食时的通信机制的启发, 20世纪90年代Dorigo提出了蚁群算法来解决经典的“旅行商问题”。蚁群算法设计虚拟的“蚂蚁”将摸索不同路线, 并留下会随时间逐渐消失的虚拟“信息素”。虚拟的“信息素”也会挥发, 每只蚂蚁每次随机选择要走的路径, 它们倾向于选择路径比较短的、信息素比较浓的路径。根据“信息素较浓的路线更近”的原则, 即可选择出最佳路线。由于这个算法利用了正反馈机制, 使得较短的路径能够有较大的机会得到选择, 并且由于采用了

收稿日期: 2007-11-27

基金项目: 国家自然科学基金资助项目(60273043); 安徽省自然科学基金资助项目(050420204)

作者简介: 王艳玲(1981-), 女, 安徽亳州人, 硕士研究生, 研究方向为智能软件; 李龙澍, 教授, 博士生导师, 研究方向为智能软件。

概率算法,所以它能够不局限于局部最优解。实验结果表明蚁群优化算法具有较强的鲁棒性和搜索较好解的能力,但同时也存在一些缺陷,如收敛速度慢、易出现停滞现象等。

1.2 AS 算法模型

考虑到真实蚁群的行为与 TSP 问题的相似性,蚁群算法(AS)首先被应用于平面上 n 个城市的 TSP 问题^[4]。 n 个城市的 TSP 问题即求从某一个城市出发,经过 $n-1$ 个城市各一次,最后回到出发点的最短环路。

为模拟实际蚂蚁的行为,首先引入如下记号:

m 表示蚁群中蚂蚁的数量; n 表示城市的数量; d_{ij} 表示城市 i 和城市 j 的距离; $\tau_{ij}(t)$ 表示 t 时刻在边 (i,j) 上的信息素轨迹强度; η_{ij} 表示边 (i,j) 的能见度,在蚂蚁算法中, η_{ij} 通常取城市 i 和城市 j 之间距离的倒数,即 $\eta_{ij} = 1/d_{ij}$; $\Delta\tau_{ij}^k$ 表示蚂蚁 k 在边 (i,j) 上留下的单位长度轨迹信息素量; p_{ij}^k 表示蚂蚁 k 的转移概率, j 是尚未访问的城市。

各路径上的信息量相等,设 $\tau_{ij}(0) = C$ (C 为常数)。每只蚂蚁根据路径上的保留信息独立地选择下一个城市。在时刻 t , 蚂蚁 k 从城市 i 转移到城市 j 的概率 p_{ij}^k 为:

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha \cdot \eta_{ij}^\beta}{\sum_{i \in \text{allowed}_k} \tau_{is}^\alpha \cdot \eta_{is}^\beta}, & \text{if } j \in \text{allowed}_k \\ 0, & \text{否则} \end{cases} \quad (1)$$

其中, α 表示蚂蚁在运动过程中所积累的信息量的重要程度; β 表示蚂蚁在运动过程中启发信息在蚂蚁选择路径中的重要程度。 $\text{allowed}_k = \{0, 1, 2, \dots, n-1\} - \text{tabu}_k$ 表示蚂蚁 k 下一步允许选择的所有城市,列表 tabu_k 记录了当前蚂蚁 k 走过的城市,当所有 n 个城市都加入到 tabu_k 中时,蚂蚁 k 便完成了一次循环,此时蚂蚁 k 所走过的路径便是问题的一个解。当所有蚂蚁完成一次循环后,各路径上的信息要根据(2)式调整:

$$\tau_{ij}(t+n) = (1-\rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij} \quad \rho \in (0,1)$$

$$\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k(t) \quad (2)$$

其中 ρ 表示路径上信息的蒸发系数, $1-\rho$ 表示信息的保留系数; $\Delta\tau_{ij}$ 表示本次循环路径 ij 上信息的增量,如果蚂蚁 k 没有经过路径 ij , 则 $\Delta\tau_{ij}^k$ 的值为零, 否则为 Q/L_k (其中 Q 为常数, L_k 表示第 k 只蚂蚁在本次循环中所走过的路径的长度)。

1.3 蚁群优化算法

针对 AS 算法的不足,一些学者提出了许多改进的蚂蚁算法,如蚁群系统、最大-最小蚁群系统和带精英策略的蚁群系统^[5]等。

蚁群系统:蚁群系统是对 AS 算法的选路和信息更新策略作了相应的改进,即:

1) 采用伪随机比率选择规则的选路方式,即对于在城市 i 的蚂蚁按公式(3)选择下一个城市 j :

$$j = \begin{cases} \arg \max_{u \in \text{allowed}_i} \{ \tau_{iu}^\alpha \cdot \eta_{iu}^\beta \} & \text{if } q \leq q_0 \\ s & \text{否则,按公式(2)进行概率式搜索} \end{cases} \quad (3)$$

其中, q 是在 $[0,1]$ 区间均匀分布的随机数, q_0 是一个参数 ($0 \leq q_0 \leq 1$), s 为根据方程式(3)给出的概率分布所选出的一个随机变量。

2) 局部信息更新。蚂蚁从城市 i 转移到城市 j 后,路径 (i,j) 上的信息量按公式(4)进行更新:

$$\tau_{ij} = (1-\xi) \cdot \tau_{ij} + \xi \cdot \tau_0, \quad \xi \in (0,1) \quad (4)$$

其中 τ_0 为常数, $\xi \in (0,1)$ 为可调参数。

3) 全局信息更新。针对全局最优解所属的边按公式(5)进行信息更新:

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \rho \cdot \Delta\tau_{ij}^{\text{gb}}(t), \quad \rho \in (0,1)$$

$$\Delta\tau_{ij}^{\text{gb}} = 1/L^{\text{gb}} \quad (5)$$

其中 L^{gb} 为当前全局最优解的长度。

最大-最小蚁群系统(MMAS)^[6]:与 AS 相比,主要作了如下改进:

① 每次循环结束后,只有最优解所属路径上的信息被更新;

② 为了避免搜索时出现停滞现象,各路径上的信息量被限制在范围 $[\tau_{\min}, \tau_{\max}]$ 内;

③ 初始时刻,各路径上的信息量取最大值。所有蚂蚁完成一次循环后,按公式(6)对路径上的信息作全局更新:

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \rho \cdot \Delta\tau_{ij}^{\text{best}}(t), \quad \rho \in (0,1)$$

$$\Delta\tau_{ij}^{\text{best}} = 1/L^{\text{best}} \quad (6)$$

其中, L^{best} 为本次循环的最优解。

1.4 算法流程

步骤 1 $nc \leftarrow 0$ (nc 为迭代步数或搜索次数);各 τ_{ij} 和 $\Delta\tau_{ij}$ 的初始化;将 m 只蚂蚁置于 n 个顶点上;

步骤 2 将每个蚂蚁的初时出发点置于当前解集中;对每个蚂蚁 k ($k = 1, 2, \dots, m$), 按概率移至下一个顶点 j ; 将顶点 j 置于当前解集;

步骤 3 计算每个蚂蚁的路径长度 L_k ($k = 1, 2, \dots, m$); 记录当前的最好解;

步骤 4 按信息更新方程修改轨迹强度;

步骤 5 对各边弧 (i,j) , 置 $\Delta\tau_{ij} \leftarrow 0$, $nc \leftarrow nc + 1$;

步骤 6 若 $nc <$ 预定的迭代次数且无退化行为, 则转步骤 2;

步骤 7 输出目前最好解。

1.5 蚁群算法的应用及与其他算法的比较

蚁群算法作为一种新的群体智能启发式优化算法, 主要用于求解组合优化问题, 其中包括旅行商问题 (TSP)、车间任务调度问题 (VRP)^[7]、图着色问题 (GCP)^[8]、有序排列问题 (SOP) 以及网络路由问题等。

为了说明蚁群算法的优点与不足, 给出用 ACS 求解 TSP 的实验结果, 该实验中除 ACS 外的其他结果都来源于文献[9], 取 10 次实验的平均值。进行对比的优化算法有: 模拟退火法 (SA), 进化计算 (EP), 遗传算法 (GA) 和模拟退火与遗传算法相结合的算法 (AG) (见表 1)。ACS 的参数设为 $M = 10, \beta = 2, q_0 = 0.9, \alpha = \rho = 0.1, \tau_0 = (n \cdot L_{nn})^{-1}$ 。

表 1 TSP 问题的多种优化算法对比

标准问题名称	ACS	GA	EP	SA	AG
Oliver30	420	421	420	424	420
(30 城市)	(420.371)	(N/A)	(423.74)	(N/A)	(N/A)
迭代次数	[1470]	[3200]	[40000]	[24617]	[12620]
Eil50	432	428	426	443	436
(50 城市)	(432.172)	(N/A)	(427.86)	(N/A)	(N/A)
迭代次数	[2412]	[25000]	[100000]	[68512]	[28111]

从实验结果可以发现, 蚁群算法具有很强的发现较好解的能力, 不容易陷入局部最优。这是因为该算法不仅利用了正反馈原理, 在一定程度上可以加快进化过程, 而且是一种本质并行的算法, 个体间不断进行信息交流和传递, 有利于发现较好解。

2 粒子群算法

2.1 粒子群算法原理

粒子群优化算法是一种基于种群的迭代搜索算法, 种群内的个体 (粒子) 不断追随最优个体进行寻优搜索。算法首先在搜索空间内随机初始化一群粒子, 每个粒子的位置是优化问题的一个解, 将其带入目标函数计算出适应值, 再根据此适应值的大小来衡量粒子的优劣。每个粒子的速度决定了其运动的方向和步长, 粒子根据本身的记忆信息和整个种群的共享信息, 不断更新自己的速度和位置, 去试探搜索空间内的不同解。在迭代过程中, 每个粒子更新速度时, 总是在原来速度的基础上调整以趋向于两个位置, 一个是粒子本身目前所找到的最优解 (pBest), 另一个是整个种群目前找到的最优解 (gBest), 并期望在向两个位置移动的过程中, 发现更好的解, 以取代 pBest 或 gBest, 通过种群中粒子的不断交互, 逐渐收敛到最优解。

2.2 基本粒子群模型

假设 m 个粒子组成一个种群, 在 D 维的空间 $[X_{\min}, X_{\max}]^D$ 中搜索。第 i 个粒子在第 t 代的位置为 $X_i(t) = (x_{i1}, x_{i2}, \dots, x_{iD})$, 速度为 $V_i(t) = (v_{i1}, v_{i2}, \dots, v_{iD})$ 。粒子本身目前所找到的最优解 pBest 为 $P_i(t) = (p_{i1}, p_{i2}, \dots, p_{iD})$, 整个种群目前找到的最优解 gBest 为 $P_g(t) = (p_{g1}, p_{g2}, \dots, p_{gD})$ 。按追随最好位置的原理, 在每个迭代周期中, 粒子按 (7)、(8) 式更新速度和位置。

$$v_{id}(t+1) = \omega \cdot v_{id}(t) + c_1 \cdot \text{rand}() \cdot (p_{id}(t) - x_{id}(t)) + c_2 \cdot \text{rand}() \cdot (p_{gd}(t) - x_{id}(t)) \quad (7)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t) \quad (8)$$

其中 c_1 为认知因子, c_2 为社会因子, 通常 $c_1 = c_2 = 2$, $\text{rand}()$ 为 $[0, 1]$ 之间的随机数。为了使速度不至于过大, 把 v_{id} 限制在 $[V_{\min}, V_{\max}]$ 之内, V_{\min} 和 V_{\max} 为常数, 通常 $V_{\min} = X_{\min}, V_{\max} = X_{\max}$ 。当 $v_{id} > V_{\max}$, 取 $v_{id} = V_{\max}$; 当 $v_{id} < V_{\min}$, 取 $v_{id} = V_{\min}$ 。 ω 为递减的惯性权值, 一般从 0.9 递减到 0.4。

2.3 粒子群优化

粒子群算法可以通过速度松弛迭代策略增强局部搜索能力, 加速收敛; 用精英集团来增加多样性, 提高全局搜索能力。

速度松弛迭代策略为: 如果当前粒子的适应度好于前一代粒子的适应度, 那么下一代粒子的速度保持不变, 否则就按照速度更新方程 (7) 对速度更新。

精英集团选择: 粒子在按 (7) 式更新速度时, 从两个方面获得信息, 一个是粒子本身的信息, 本身目前所找到的最优解 P_i ; 另一个是种群共享信息, 整个种群目前搜索到的最优位置 P_g 。事实上, 有时其他一些粒子的 P_i 比 P_g 的解只是稍微差一点, 都是较好的解, 但由于 (7) 式只利用了 P_g 的信息, 没有使那些较好 P_i 的信息在种群中共享。精英集团的概念是, 在每步迭代中, 对 P_i 的适应度从小到大排序取前 k ($k \leq m$) 个粒子作为精英集团 Ω_k , 集团内的个体都有机会作为种群的最优解 P_g 带入公式 (7) 进行计算。这样不但解决了种群的多样性问题, 同时使优势粒子的信息得到了充分利用。

2.4 算法流程

步骤 1 初始化 m 个粒子的位置和速度;

步骤 2 把每个粒子的位置带入目标函数评价其优劣;

步骤 3 对每个粒子的函数值与其本身目前最优位置 P_i 的函数作比较, 如果好于 P_i , 则将其作为目前最好位置 P_i ;

步骤4 对整个粒子群目前最优位置进行 P_i 排序,前 k 个作为精英集团 Ω_k ;

步骤5 每个粒子从精英集团 Ω_k 中,随机选取 P_r 作为 P_g ;

步骤6 如果适应度变坏, P_g 代入速度迭代方程,重新计算速度,否则,速度不变;

步骤7 检查速度各个分量是否在 $[V_{\min}, V_{\max}]$ 范围内,如果大于 V_{\max} 设为 V_{\max} ;如果小于 V_{\min} 设为 V_{\min} ;

步骤8 按位置更新方程,更新粒子位置;

步骤9 检查各个分量是否在 $[X_{\min}, X_{\max}]$ 范围内,如果超出,在 $[X_{\min}, X_{\max}]$ 内随机取一个值,设为该分量;

步骤10 如果未达到预先设定的最大代数或未达到足够好的函数值,则返回步骤2。

2.5 粒子群算法的应用

由于粒子群算法出色的性能,目前已广泛应用于函数优化、神经网络训练、模糊系统控制等众多领域。下面简要介绍粒子群算法在神经网络中的应用。

当粒子群算法用于神经网络训练网络权值时,粒子就表示神经网络的一组权,粒子的纬度就是神经网络中权值的个数。一般神经网络的初始值介于 -1 和 $+1$ 之间,训练结束后的权值也是 -1 和 $+1$ 之间,因此,粒子的范围可以设定为 -1 和 $+1$ 之间。惯性因子 ω 的取值既要考虑到避免陷入局部极小,又要保证收敛性,算法的初期阶段让惯性因子 ω 取较大的值,有利于跳出局部极小点,逐步调整 ω ,使其递减,以保证算法的收敛性。实验结果表明,粒子群优化算法训练的神经网络收敛速度明显加快。

3 结束语

群体智能是新兴的用于寻找全局最优解的算法,已经广泛地应用于许多领域,取得很好的效果。群体智能的特点和优点是:群体中相互合作的个体是分布式的,这样更能够适应当前网络环境下的工作状态;没有中心的控制与数据,这样的系统更具有鲁棒性,不会由于某一个或者某几个个体的故障而影响整个问题的

求解。可以不通过个体之间直接通信而是通过非直接通信进行合作,这样的系统具有更好的可扩充性。由于系统中个体的增加而增加的系统的通信开销在这里十分小。系统中每个个体的能力十分简单,这样每个个体的执行时间比较短,并且实现也比较简单,具有简单性。因为具有这些优点,虽说群体智能的研究还处于初级阶段,并且存在许多困难,但可预言群体智能的研究代表了以后计算机研究发展的一个重要方向。

参考文献:

- [1] Dorigo M, Gambardella L M. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem [J]. IEEE Transactions on Evolutionary Computations, 1997, 1(1): 53-66.
- [2] Gambardella L M, Dorigo M. Solving Symmetric and Asymmetric TSPs by Colonies [C]//In proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '96). [s.l.]: IEEE Press, 1996: 622-627.
- [3] Kennedy J, Eberhart R C. Particle swarm optimization [C]//In: Proceedings of IEEE International Conference on Neural Networks. Piscataway, NJ: [s.n.], 1995: 1942-1948.
- [4] Dorigo M, Maniezzo V, Colomi A. The ant system: optimization by a colony of cooperating agents [J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B, 1996, 26(1): 29-41.
- [5] Bullnheimer B, Hartl R F, Strauss C. A New Rank-based Version of the ant system: A Computational Study [R]. Vienna: Institute of Management Science, University of Vienna, 1997.
- [6] Stutzle T, Hoos H H. MAX-MIN Ant System [J]. Future Generation Computer Systems, 2000, 16(8): 889-914.
- [7] Colomi A, Dorigo M, Maniezzo V, et al. Ant system for job-shop scheduling [J]. Belgian Journal of Operations Research, Statistics and Computer Science (JORBEL), 1994, 34: 39-53.
- [8] Costa D, Hertz A. Ants can color graphs [J]. Journal of Operational Research Society, 1997, 48: 295-305.
- [9] Durbin R, Willshaw D. An Analogue Approach to the Traveling Salesman Problem Using an Elastic Net Method [J]. Nature, 1987(326): 689-691.

(上接第113页)

- [7] 金玉坚, 刘焱. 新型网络信息检索效果评价指标体系设计 [J]. 现代情报, 2005(4): 185-188.
- [8] 李振龙. web信息检索的技术分析与发展策略研究 [J]. 计算机科学, 2006, 33(4): 181-184.
- [9] 卫琳. 基于搜索结果的个性化推荐系统研究 [J]. 计算机

技术与发展, 2007, 17(9): 66-70.

- [10] Zamir O, Etzioni O. Web Document Clustering: A Feasibility Demonstration [C]//In: Proc. of SIGIR'98. New York: ACM Press, 1998: 46-54.