Vol. 41 吉林大学学报(理学版) No. 4
2003年10月 JOURNAL OF JILIN UNIVERSITY (SCIENCE EDITION) 477～480

粒子群优化算法求解旅行商问题

黄 岚, 王康平, 周春光, 庞 巍, 董龙江, 彭 利
(吉林大学计算机科学与技术学院, 长春 130012)

提要: 首先介绍粒子群优化的搜索策略与基本算法, 然后通过引入交换子和交换序的概念, 构造一种特殊的粒子群优化算法, 并用于求解旅行商问题. 实验表明了在求解组合优化问题中的有效性.

关键词: 粒子群优化算法; 旅行商问题; 组合优化
中图分类号: TP31 文献标识码: A 文章编号: 1671-5489(2003)04-0477-04

粒子群优化算法(Particle swarm optimization, 简称PSO)最初由Kennedy和Eberhart[1]提出, 是一种基于叠代的优化方法, 因其概念简单、实现容易, 而引起学术界的广泛重视. 目前已被应用于多目标优化、模式识别、信号处理和决策支持等领域[2～4]. 旅行商问题(Traveling salesman problem, 简称TSP)描述为: 给定n个城市和两两城市之间的距离, 求一条访问各城市一次且仅一次的最短路线. TSP是著名的组合优化问题, 是NP难题, 常被用来验证智能启发式算法的有效性[5,6].

目前, PSO算法在很多连续优化问题中得到成功应用, 而在离散域上的研究和应用还很少, 尤其是用PSO求解TSP问题是一个新的研究方向.

1 基本粒子群算法

在PSO算法中, 粒子群在一个n维空间中搜索, 其中的每个粒子所处的位置都表示问题的一个解. 粒子通过不断调整自己的位置X来搜索新解. 每个粒子都能记住自己搜索到的最好解, 记作P_{id}, 以及整个粒子群经历过的最好位置, 即目前搜索到的最优解, 记作P_{gd}. 每个粒子都有一个速度, 记作V,

$$V_{id}' = \omega V_{id} + \eta_1 \mathrm{rand}(\ )(P_{id} - X_{id}) + \eta_2 \mathrm{rand}(\ )(P_{gd} - X_{id}), \tag{1.1}$$

其中V_{id}表示第i个粒子第d维上的速度, ω为惯性权重, η_1, η_2为调节P_{id}和P_{gd}相对重要性的参数, rand( )为随机数生成函数. 这样, 可以得到粒子移动的下一位置:

$$X_{id}' = X_{id} + V_{id}. \tag{1.2}$$

从(1.1)式和(1.2)式可以看出, 粒子的移动方向由三部分决定, 自己原有的速度V_{id}、与自己最佳经历的距离$(P_{id} - X_{id})$和与群体最佳经历的距离$(P_{gd} - X_{id})$, 并分别由权重系数ω, η_1和η_2决定其相对重要性.

PSO的基本算法步骤描述如下:

(1) 初始化粒子群, 即随机设定各粒子的初始位置X和初始速度V;

(2) 计算每个粒子的适应度值;

(3) 对每个粒子, 比较它的适应度值和它经历过的最好位置P_{id}的适应度值, 如果更好, 更新P_{id};

(4) 对每个粒子, 比较它的适应度值和群体所经历最好位置P_{gd}的适应度值, 如果更好, 更新P_{gd};

(5) 根据(1.1)式和(1.2)式调整粒子的速度和位置;

(6) 如果达到结束条件(足够好的位置或最大迭代次数), 则结束; 否则转步骤(2).

收稿日期: 2003-07-10.

作者简介: 黄 岚(1974～), 女, 博士研究生, 讲师, 从事智能算法与应用的研究, E-mail: lanh@21cn.com. 联系人: 周春光(1947～), 男, 教授, 博士生导师, 从事计算智能的研究, E-mail: cgzhou@mail.jlu.edu.cn.

基金项目: 国家自然科学基金(批准号: 60175024)和教育部"符号计算与知识工程"重点实验室基金.

PSO 是一种进化计算方法, 它有以下几个进化计算的典型特征: 有一个初始化过程, 在这个过程中, 群体中的个体被赋值为一些随机产生的初始解; 通过产生更好的新一代群体来搜索解空间; 新一代群体产生在前一代的基础上.

2 旅行商问题

TSP 是运筹学、图论和组合优化中的 NP 难题, 常被用来验证智能启发式算法的有效性. 主要的智能启发式算法包括最近邻域搜索、模拟退火、神经网络方法、遗传算法和蚂蚁算法等.

旅行商问题描述如下: 给定 n 个城市及两两城市之间的距离, 求一条经过各城市一次且仅一次的最短路线. 其图论描述为: 给定图 $G=(V,A)$, 其中 V 为顶点集, A 为各顶点相互连接组成的弧集, 已知各顶点间连接距离, 要求确定一条长度最短的 Hamilton 回路, 即遍历所有顶点一次且仅一次的最短回路. 设 d_{ij} 为城市 i 与 j 之间的距离, 即弧 (i,j) 的长度. 引入决策变量:

$$x_{ij}=\begin{cases}1, & \text{若旅行商访问城市 } i \text{ 后访问城市 } j;\\ 0, & \text{否则},\end{cases} \tag{2.1}$$

则 TSP 的目标函数为

$$\min Z=\sum_{i,j=1}^{n} x_{ij} d_{ij}. \tag{2.2}$$

TSP 问题描述非常简单, 但最优化求解很困难, 若用穷举法搜索, 则要考虑所有可能情况, 并两两对比, 找出最优, 其算法复杂性呈指数增长, 即所谓的"组合爆炸". 所以, 寻求和研究 TSP 的有效启发式算法, 是问题的关键.

PSO 算法虽然成功地应用于连续优化问题中, 但在组合优化问题中的研究和应用还很少. 下面将通过引入交换子和交换序的概念, 对基本 PSO 算法进行改造, 并将其应用于求解 TSP 问题中.

3 交换子和交换序

定义 3.1 设 n 个节点的 TSP 问题的解序列为 $S=(a_i)$, $i=1,\cdots,n$. 定义交换子 $\mathrm{SO}(i_1,i_2)$ 为交换解 S 中的点 a_{i_1} 和 a_{i_2}, 则 $S'=S+\mathrm{SO}(i_1,i_2)$ 为解 S 经算子 $\mathrm{SO}(i_1,i_2)$ 操作后的新解, 这里为符号"+"赋予了新的含义.

例 3.1 有一个 5 节点的 TSP 问题, 其解为 $S=(1\ 3\ 5\ 2\ 4)$, 交换算子为 $\mathrm{SO}(1,2)$, 则

$$S'=S+\mathrm{SO}(1,2)=(1\ 3\ 5\ 2\ 4)+\mathrm{SO}(1,2)=(3\ 1\ 5\ 2\ 4).$$

定义 3.2 一个或多个交换子的有序队列就是交换序, 记作 SS.

$$\mathrm{SS}=(\mathrm{SO}_1,\mathrm{SO}_2,\cdots,\mathrm{SO}_n), \tag{3.1}$$

其中 $\mathrm{SO}_1,\mathrm{SO}_2,\cdots,\mathrm{SO}_n$ 是交换子, 它们之间的顺序是有意义的.

交换序作用于一个 TSP 解上意味着这个交换序中的所有交换子依次作用于该解上, 即

$$S'=S+\mathrm{SS}=S+(\mathrm{SO}_1,\mathrm{SO}_2,\cdots,\mathrm{SO}_n)=[(S+\mathrm{SO}_1)+\mathrm{SO}_2]+\cdots+\mathrm{SO}_n. \tag{3.2}$$

定义 3.3 不同的交换序作用于同一解上可能产生相同的新解, 所有有相同效果的交换序的集合称为交换序的等价集.

定义 3.4 若干个交换序可以合并成一个新的交换序, 定义 $\oplus$ 为两个交换序的合并算子.

例 3.2 设两个交换序 SS_1 和 SS_2, 按先后顺序作用于解 S 上, 得到新解 S'. 假设另外有一个交换序 SS' 作用于同一解 S 上, 能够得到相同的解 S', 可定义

$$\mathrm{SS}'=\mathrm{SS}_1\oplus\mathrm{SS}_2, \tag{3.3}$$

SS' 和 $\mathrm{SS}_1\oplus\mathrm{SS}_2$ 属于同一等价集. 一般来说, SS' 不惟一.

定义 3.5 在交换序等价集中, 拥有最少交换子的交换序称为该等价集的基本交换序.

可按如下的方法构造一个基本交换序. 设给定两个解路径 A 和 B, 需要构造一个基本交换序 SS, 使得 $B+\mathrm{SS}=A$.

$$A:\ (1\ 2\ 3\ 4\ 5);\qquad B:\ (2\ 3\ 1\ 5\ 4).$$

可以看出, $A(1)=B(3)=1$, 所以第一个交换子是 $SO(1,3)$, $B_1=B+SO(1,3)$, 得到 B_1: (1 3 2 5 4), $A(2)=B_1(3)=1$, 所以第二个交换子是 $SO(2,3)$, $B_2=B_1+SO(2,3)$, 得到 B_2: (1 2 3 5 4).

同理, 第三个交换子是 $SO(4,5)$, $B_3=B_2+SO(4,5)=A$. 这样, 就得到一个基本交换序:

$$SS=A-B=(SO(1,3),SO(2,3),SO(4,5)).$$

4 求解 TSP 的 PSO 算法

基本 PSO 算法中的速度算式(1.1)已不适合 TSP 问题, 于是重新构造了速度算式:

$$V_{id}'=V_{id}\oplus\alpha(P_{id}-X_{id})\oplus\beta(P_{gd}-X_{id}), \tag{4.1}$$

其中 $\alpha,\beta(\alpha,\beta\in[0,1])$ 为随机数. $\alpha(P_{id}-X_{id})$ 表示基本交换序 $(P_{id}-X_{id})$ 中的所有交换子以概率 α 保留; 同理, $\beta(P_{gd}-X_{id})$ 表示基本交换序 $(P_{gd}-X_{id})$ 中的所有交换子以概率 β 保留. 由此可以看出, α 的值越大, $(P_{id}-X_{id})$ 保留的交换子就越多, P_{id} 的影响就越大; 同理, β 的值越大, $(P_{gd}-X_{id})$ 保留的交换子就越多, P_{gd} 的影响就越大.

求解 TSP 的 PSO 算法步骤描述如下:

(1) 初始化粒子群, 即给群体中的每个粒子赋一个随机的初始解和一个随机的交换序;

(2) 如果满足结束条件, 转步骤(5);

(3) 根据粒子当前位置 X_{id}, 计算其下一个位置 X_{id}', 即新解;

1) 计算 P_{id} 和 X_{id} 之间的差 A, $A=P_{id}-X_{id}$, 其中 A 是一个基本交换序, 表示 A 作用于 X_{id} 得到 P_{id};

2) 计算 $B=P_{gd}-X_{id}$, 其中 B 也是一基本交换序;

3) 根据(4.1)式计算速度 V_{id}', 并将交换序 V_{id}' 转换为一个基本交换序;

4) 计算搜索到的新解

$$X_{id}'=X_{id}+V_{id}; \tag{4.2}$$

5) 如果找到一个更好的解, 则更新 P_{id};

(4) 如果整个群体找到一个更好的解, 更新 P_{gd}. 转步骤(2).

(5) 显示求出的结果值.

5 实验与结论

我们用 14 个点的 TSP 标准问题(问题来源及最好解见 http://www.crpc.rice.edu/softlib/tsplib/) 来验证算法的有效性. 实验环境为 PC(PentiumIV-2GHz CPU, 256M RAM, Win2000 OS, VC++ 6.0). 14 点 TSP 的问题描述列于表 1, 初始的随机解与本算法获得的最好解如图 1 所示, 算法性能分析列于表 2.

Table 1 TSP with 14 nodes

Node	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Coordinate X	16.47	16.47	20.09	22.39	25.23	22.00	20.47	17.20	16.30	14.05	16.53	21.52	19.41	20.09
Coordinate Y	96.10	94.44	92.54	93.37	97.24	96.05	97.02	96.29	97.38	98.12	97.38	95.59	97.13	94.55

Table 2 Analyses of the algorithm performance

Size of solution space	(14-1)!/2= 3 113 510 400
Number of particles in the swarm	100
Average number of iterations	20 000
Average size of search space	20 000* 100= 2 000 000
Search space/solution space	2 000 000/3 113 510 400= 0.064%
Best solution of the algorithm	1→10→9→11→8→13→7→12→6→5→4→3→14→2
Length	30.878 5(Equal to the best known result in the world)

从实验结果可以看出, 算法只搜索了一个很小的区域就得到了一个已知最好的解, 收敛速度很快, 这表明算法有效. 我们实验所采用的虽然是只有 14 个点 TSP 算例, 并且算法比目前解决 TSP 问题的经典算法

(如 Lin-Kernighan[7]算法) 在解决问题的能力和速度方面有一定的差距, 但应用 PSO 算法解决 TSP 问题是一种崭新的尝试.

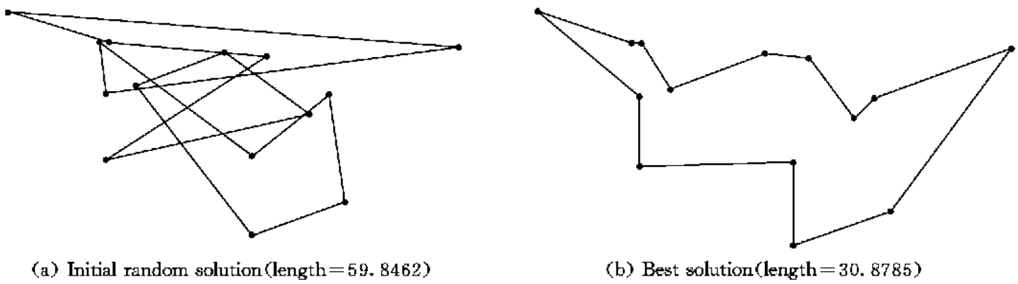

Fig.1 The solution paths of TSP with 14 nodes

参 考 文 献

[1] Eberhart R, Kennedy J. A New Optimizer Using Particles Swarm Theory [C]. Proc Sixth International Symposium on Micro Machine and Human Science. Nagoya, Japan: IEEE Service Center, Piscataway, 1995. 39～43.

[2] Xie X, Zhang W, Yang Z. Adaptive Particle Swarm Optimization on Individual Level [C]. International Conference on Signal Processing (ICSP 2002). Beijing: 2002. 1215～1218.

[3] Parsopoulos K E, Vrahatis M N. Recent Approaches to Global Optimization Problems Through Particle Swarm Optimization [J]. *Natural Computing*, 2002, **1**(2～3): 235～306.

[4] Ray T, Liew K M. A Swarm Metaphor for Multiobjective Design Optimization [J]. *Engineering Optimization*, 2002, **34**(2): 141～153.

[5] Zhou Chun-guang(周春光), Liang Yan-chun(梁艳春). Computational Intelligence(计算智能) [M]. Changchun (长春): Jinlin University Press(吉林大学出版社), 2001. 269～277.

[6] Huang Lan(黄 岚), Wang Kang-ping(王康平), Zhou Chun-guang(周春光), *et al*. Hybrid Ant Colony Algorithm for Traveling Salesman Problem(基于蚂蚁算法的混合方法求解旅行商问题) [J]. *Journal of Jilin University (Science Edition)* [吉林大学学报(理学版)], 2002, **40**(4): 369～373.

[7] Lin S, Kernighan B W. An Effective Heuristic Algorithm for the Traveling Salesman Problem [J]. *Operations Res*, 1973, **21**: 498～516.

Particle Swarm Optimization for Traveling Salesman Problems

HUANG Lan, WANG Kang-ping, ZHOU Chun-guang, PANG Wei, DONG Long-jiang, PENG Li
(*College of Computer Science and Technology, Jilin University, Changchun* 130012, *China*)

Abstract: This paper introduces the basic algorithm and search strategies of particle swarm optimization (PSO), *via* presenting the concepts of swap operator and swap sequence an algorithm of a kind of special particle swarm optimization is constructed and then proposes its application to traveling salesman problems(TSP). The experiments show the new PSO can achieve good results.

Keywords: particle swarm optimization; traveling salesman problem; combinatorial optimization

(责任编辑: 赵立芹)