

文章编号: 1673-3851 (2014) 03-0287-05

基于局部搜索机制快速求解 TSP 问题的自适应遗传算法

夏 凯¹, 戴文战²

(1. 浙江理工大学机械与自动控制学院, 杭州 310018; 2. 浙江工商大学信息与电子工程学院, 杭州 310018)

摘 要: 提出了一种基于局部搜索机制快速求解 TSP 的遗传算法。基于局部搜索机制, 自适应地将标准遗传算法与局部启发式算法结合, 使得局部启发式算法只在有效改善种群个体质量的情况下才允许执行, 有效地避免了因局部搜索次数过多而引起的陷入局部最优和计算负担过重现象的发生。仿真结果表明, 该算法具有较强的全局优化能力及较快的收敛速度, 在求解 TSP 问题时具有较高效率。

关键词: 局部搜索机制; 自适应; 遗传算法; 旅行商问题

中图分类号: U461; TP308 **文献标志码:** A

0 引 言

旅行商问题 (traveling salesman problem, TSP) 一直具有重要理论研究价值和广泛工程应用价值。精确算法如分支定界法、动态规划法等, 在问题规模较小时能得到最优解, 但算法的时间复杂度和空间复杂度都会随着问题规模的增加而成指数增长。由于近似算法能够在合理的时间和存储空间范围内求出问题的最优解或近似最优解, 因而众多学者对其进行了大量的研究。遗传算法 (genetic algorithm, GA) 因其具有较强的全局搜索能力、高效的群体搜索策略以及与目标梯度信息无关等优点, 是求解 TSP 问题的一种理想方法。

GA 中一个较难解决的问题是如何较快地找到最优解并防止“早熟”收敛问题。许多研究者提出了各种改进方法来提高 GA 的搜索性能。彭丹平等^[1]从相似性的思想出发, 按适应值相似性将群体分级, 在不同的级内采用不同的操作, 产生数目不等的解并加速算子使其更接近局部极小值, 较好地解决了群体多样性与收敛性的矛盾。姜昌华^[2]等结合遗传算法和 2-opt 领域搜索优化技术, 提出了 K 近邻点集以缩减搜索空间从而加快求解速度。杨

辉^[3]等提出了构建基因库与遗传算法相结合的算法, 利用基因库实现初始种群的优生和指导种群的进化方向, 并用全局搜索算子和局部搜索算子增强 GA 的“探测”和“开发”能力, 加快了算法的收敛速度和寻优能力。Ren 等^[4]构建了混沌搜索算法, 并结合单亲遗传算法和贪婪局部搜索算法求解旅行商问题, 能较好逃离局部最优, 获得全局最优或近似最优解。Yang 等^[5]引入了迁移机制, 当进化中的种群多样性缺失时, 通过迁移机制, 将种群中的一部分个体用优良的个体替代。Hua 等^[6]提出了一种自适应调整控制参数的遗传算法, 自适应的调整交叉和变异概率, 并与 2-opt 算法相结合。许多研究结果表明将遗传算法与局部启发式算法结合起来可以有效地提高求解大型 TSP 问题的质量, 利用遗传算法自适应的随机搜索性能“勘探”潜在的最优空间, 而局部启发式搜索则在这些子空间进行深入“开发”。这类算法在提高收敛速度和寻求全局最优之间找到一个平衡点^[7]。文献[1-6]虽然将局部搜索算法加入到遗传算中, 但往往导致过多的局部搜索而陷入局部最优和造成计算时间上的不必要浪费。

针对这个问题, 提出了一种基于局部搜索机制的自适应遗传算法 (adaptive genetic algorithm,

收稿日期: 2013-11-07

基金项目: 国家自然科学基金(61374022); 国家高新技术研究发展项目(2009AA04Z139)

作者简介: 夏 凯(1988-), 男, 浙江富阳人, 硕士研究生, 主要从事智能优化与模式识别等方面的研究。

AGA)。在算法中,局部搜索是根据种群个体适应度值标准偏差和种群个体的平均巡回路径长度的变化情况自适应地与全局搜索相结合,局部搜索只在能有效改善种群个体质量的情况下才允许执行,因而能有效避免陷入局部最优,减轻了计算负担。

1 TSP 问题模型

TSP 模型可简单描述如下:

假设一名旅行商需要访问 n 个城市且每个城市的坐标为 (x_i, y_i) , $(i \in [1, n])$, 城市 i 与城市 j 之间的欧拉距离是:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (1)$$

旅行商需要找到一条哈密顿回路,即从任意一城市出发,路途中经过每个城市当且仅当一次然后回到原出发点,使其路径长度最短。即:

$$L = \min(\sum_{i=1}^{n-1} d_{i,i+1} + d_{n,1}) \quad (2)$$

此外,TSP 问题模型也可以从图论的角度来描述。给出有向图 $G=(V, E)$, 图中每条边 $e \in E$, 有一个非负的权值 $\omega(e)$, 目的是在有向图 G 上发现一个具有最小权值和的哈密顿回路^[8]。如下:

$$W(C) = \min \sum_{e \in E(C)} \omega(e) \quad (3)$$

2 局部搜索策略

本研究采用了 2-opt 邻域搜索优化技术作为与 GA 结合的局部搜索算法。利用 GA 自适应性的随机搜索性能勘探潜在的最优解空间,2-opt 在这些子空间内进行深入搜索。按照特定的适应性机制,将 GA 与 2-opt 局部搜索算法自适应地结合,能较好地解决提高收敛速度和寻求全局最优之间的矛盾,使它们达到平衡。其基本思想为:局部搜索算法只在需要的时候才允许执行。提出了以下适应性机制,确保只在局部搜索算法能有效地改善种群个体解的质量的前提下才允许执行。

2.1 局部搜索机制

局部搜索只在能提供给种群进化有用的信息时才被执行。一开始种群个体适应度值标准偏差 σ 和种群个体的平均巡回路径距离 ρ 都比较大,但是随着种群的不断进化,最后种群中的个体非常相似或趋于一致,此时 σ 很小或趋于 0, ρ 也变得很小。

偏差 σ 的计算公式如下:

$$\sigma = \sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 / (n-1)} \quad (4)$$

其中 n 为种群中个体的总数, x_i 为种群中第 i 个个

体的适应度, \bar{x} 为种群中所有个体适应度的平均值。

为了跟踪 σ 的变化,本文定义 V 来表示两个连续代之间 σ 的变化率。 V 的计算公式如下:

$$V = \frac{\sigma(i)}{\sigma(i-1)} \quad (5)$$

其中 $\sigma(i)$ 表示第 i 代种群个体适应度值的标准偏差。如果 $V > 1$, 则说明种群中第 i 代种群中个体适应度值偏离种群平均适应度值的程度比第 $i-1$ 代大,即种群中的个体适应度分布比较分散。

同时,为了跟踪 ρ 的变化,本文定义 W 来表示连续两代之间 ρ 的变化率。 W 的计算公式如下:

$$W = \frac{\rho(i)}{\rho(i-1)} \quad (6)$$

其中 $\rho(i)$ 表示第 i 代种群个体所代表的巡回路径距离平均值。如果 $W > 1$, 则说明第 i 代种群中个体的平均距离比第 $i-1$ 代大,即种群中个体的平均个体解质量比前代差。

如果同时满足 $V > 1$ 且 $W > 1$, 则表明第 i 代种群个体解的质量比第 $i-1$ 代差,说明算法正在探索新的解空间,这些解子空间可能是通过遗传算法的交叉或变异算子产生,由于新产生的个体的适应度值与种群中剩余个体相比相差较大,这样会导致种群的标准偏差值增加和个体平均适应度值减少,此时需要进一步对新的解子空间进行开发,则启用局部搜索。

当种群中最优个体适应度值 f_{best} 与个体平均适应度值 f_{ave} 的偏差率 φ 很小时,此时,局部搜索不但不能有效的改善解的质量,而且会额外的增加计算时间,则结束局部搜索。偏差率的参考值为 1%。

φ 的计算公式如下:

$$\varphi = \frac{(f_{best} - f_{ave})}{f_{best}} \times 100\% \quad (7)$$

3 算法流程

算法流程描述如下:

Step1 遗传代数计数器初始化: $t \leftarrow 0$ 。

Step2 基于基因库初始化群体 $P(t)$, 对 $P(t)$ 进行预处理并计算个体适应度。

Step3 采用赌轮转方式选择个体,同时采用精英保留策略,即保存当前群体中的最优个体。

Step4 对种群个体进行杂交、变异操作,即进行标准遗传算法操作,并计算种群个体适应值。

Step5 判断是否应该执行局部搜索。如果满足 $V > 1$, $W > 1$ 且 $\varphi > 1\%$, 转到 Step6 执行,否则转到 Step3。

Step6 局部搜索优化。随机从种群中选择一

定比例的个体,对这些个体进行局部搜索优化,将优化后的个体替代原先的个体。

Step7 终止条件判断。若 t 总设定的进化迭代次数,则 $t=t+1$,转向 Step3。否则,输出最优结果,算法结束。

4 仿真研究

本文针对城市规模分别为 31^[1]和 50^[4]的典型 TSP 问题,用本算法在 matlab7.0 软件上进行仿真研究。初始化控制参数:种群个体数均设为 30,最大迭

代次数分别设为 100 和 500,交叉概率均设为 0.9,变异概率均设为 0.01。每组实验均分别进行 10 次,得出的城市规模为 31 的最优路径如图 1 所示。最短路径长度为 15 378,最短迭代次数为 18,如图 2 所示。

城市规模为 50 的最优路径如图 3 所示。最短路径长度为 549.96,最短迭代次数为 16,如图 4 所示。仿真结果表明,AGA 算法具有较强的全局搜索能力和较快的收敛速度,能在较短的时间内获得满意的解,确实能够有效地提高 TSP 问题的求解质量和求解速度。

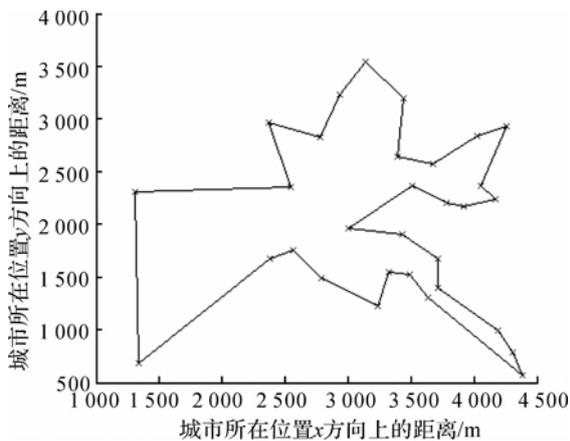


图 1 城市规模为 $N=31$ 时的最优路径

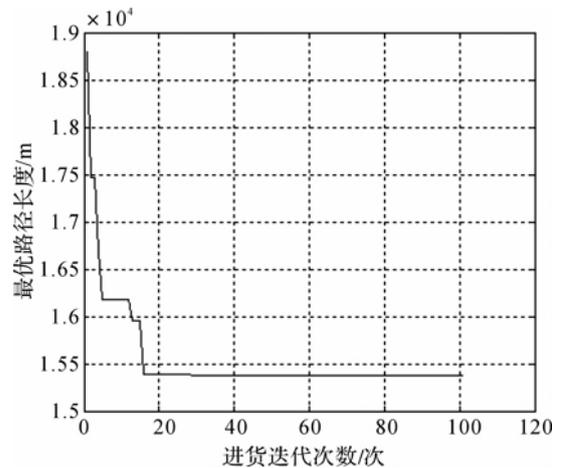


图 2 城市规模为 $N=31$ 时的优化过程

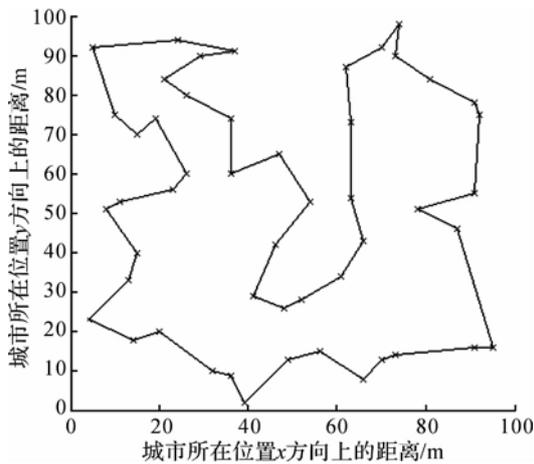


图 3 城市规模为 $N=50$ 时的最优路径

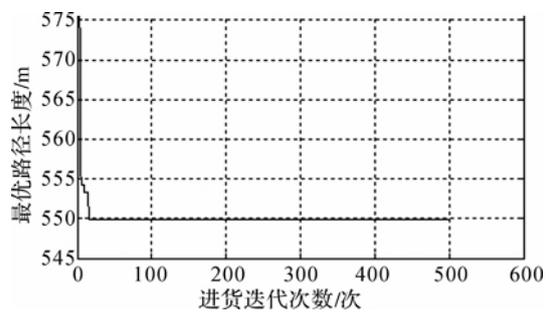


图 4 城市规模为 $N=50$ 时的优化过程

为了更好的检验本算法的性能,将 AGA 算法所得的最优解分别与文献[1]和[4]给出的最优解进

行比较,结果见表 1。

表 1 城市规模 $N=31,50$ 的仿真结果

城市规模/个	本文结果/m	已公布最优解/m	偏差/m	本文的改进率/%
31	15 378	15 385 ^[1]	7	+0.045
50	549.960	551.314 ^[4]	1.354	+0.245

其中,城市规模指代的是具有不同城市个数的 TSP 问题;本文结果指代的是采用本文算法求

解对应 TSP 问题获得的最短路径长度;已公布最优解指代的是现有已公布的算法求解对应 TSP 问

题所获得的最短路径长度;偏差=本研究算法获得的结果-已知公布结果;改进率=(本研究算法获得的最短路径长度-已知公布结果)/已知公布结果。从表1中可以看出,通过本文提出的算法求解规模分别为31和50的TSP问题,所得的最优解明显优于文献[1]和[4]给出的最优解,证明了本算法的有效性。

本研究运行了来自TSPLIB的多个实例。将计算结果与标准遗传算法(SGA)、混合遗传算法(HGA)(将SGA与2opt结合的混合算法)进行比较。其中HGA和AGA算法的最大迭代次数设为500,SGA的最大迭代次数设为2000。仿真结果具体见表2。表中的每个实例都是在连续运行10次

后得到的统计结果。实例表示的是带有不同城市数的TSP具体问题;最优表示的是目前所获得的求解所对应具体TSP实例的最优路径长度;算法表示的是求解所对应具体实例问题所使用的不同求解方法,最好表示的是不同算法求解对应的TSP实例问题时得到的最短的路径长度;平均表示的是指每种算法针对某一实例在10次试验中所得的在规定的迭代次数内收敛到某一值后不在发生变化的最短路径长度的平均值。从实验结果可以看出,AGA算法求得的最好解与TSPLIB提供的最优解非常接近,而且求解的平均代数与其它算法相比要少,均优于文献[1]和[4]所求得结果。AGA算法确实能够有效的提高TSP的求解质量和求解速度。

表2 SGA、HGA、AGA算法实验结果的比较

实例	算法	最优路径长度/m	10次试验最好解/m	10次试验平均值/m
Eil51	SGA	426.00	508.71	555.87
	HGA	428.98	430.65	280.70
	AGA	428.98	430.11	176.30
St70	SGA	675.00	985.74	1125.80
	HGA	677.11	685.26	251.90
	AGA	677.11	680.09	198.80
Eil76	SGA	538.00	760.17	868.56
	HGA	550.01	556.25	280.80
	AGA	545.83	550.03	187.10
Eil101	SGA	629.00	1099.10	1960.90
	HGA	651.27	660.00	369.40
	AGA	642.82	653.45	201.10
Koral00	SGA	21282	45968	1969.10
	HGA	21299	21545	328.90
	AGA	21285	21327	223.40

5 结论

针对TSP问题,提出了一种基于局部搜索机制的自适应遗传算法。设计了局部搜索机制,在该机制的指导下,把全局搜索与局部搜索方法自适应地结合,有效减小了算法陷入局部最优解的可能,增强了算法的寻优能力和速度。实验结果表明,该算法是有效的。对生产调度、路径规划等排列组合优化问题具有现实指导意义。

由于提出的算法在求解小规模TSP问题时比较有效,而在求解大规模TSP问题时存在搜索质量差和速度慢等缺点,仍需进一步改进算法的性能加以解决。

参考文献:

- [1] 彭丹平,林志毅,王江晴. 求解TSP的一种改进遗传算法[J]. 计算机工程与应用, 2006, 42(13): 91-93.
- [2] 姜昌华,胡幼华. 一种求解旅行商问题的高效混合遗传算法[J]. 计算机工程与应用, 2004, 40(22): 67-70.
- [3] 杨辉,康立山,陈毓屏. 一种基于构建基因库求解TSP问题的遗传算法[J]. 计算机学报, 2003, 26(12): 1753-1758.
- [4] Ren S, Wang J, Zhang X J. Research on chaos partheno-genetic algorithm for TSP[C]//Computer Application and System Modeling (ICCSM), 2010 International Conference on. IEEE, 2010, 1: 290-293.
- [5] Yang W, Hu Y, G K. Parallel search strategies for TSPs using a greedy genetic algorithm[C]//Natural Computation, 2007. ICNC 2007. Third International Conference on. IEEE, 2007, 3: 786-790.

- [6] Hua F D, Xiao L L, Xue L. An improved genetic algorithm for combinatorial optimization[C]//Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on. IEEE, 2011, 1: 58-61.
- [7] 高海昌, 冯博琴, 朱 利. 智能优化算法求解 TSP 问题[J]. 控制与决策, 2006, 21(3): 241-247.
- [8] 王宇平, 李英华. 求解 TSP 的量子遗传算法[J]. 计算机学报, 2007, 30(5): 748-755.

Adaptive Genetic Algorithm Based on Local Search Mechanism Quickly Solving TSP

XIA Kai¹, DAI Wen-zhan²

(1. School of Mechanical Engineering and Automation, Zhejiang Sci-Tech University, Hangzhou, 310018, China; 2. School of Information and Electronic Engineering, Zhejiang Gongshang University, Hangzhou 310018, China)

Abstract: This paper proposes a genetic algorithm based on local search mechanism quickly solving TSP. Based on local search mechanism, this paper adaptively combines standard genetic algorithm with local heuristic algorithm, so that local heuristic algorithm is allowed to execute only when individual quality is effectively improved, and thus avoid effectively phenomena of local optimum or computational overburden due to excessive number of local search. Simulation results show that this algorithm has a strong global optimization ability, fast convergence speed, and high efficiency in solving TSP.

Key words: local search mechanism; adaptive; genetic algorithm; traveling salesman problem

(责任编辑: 康 锋)