

文章编号: 1001-0920(2004)02-0166-05

复杂环境下基于蚁群优化算法的机器人路径规划

樊晓平, 罗 熊, 易 晟, 张 航
(中南大学 信息科学与工程学院, 湖南 长沙 410083)

摘 要: 针对复杂环境下机器人的路径规划问题, 将蚁群优化算法引入这一新的应用领域, 设计了相应的算法, 解决了以前尚未涉足的带约束条件的连续函数优化问题. 仿真结果验证了所设计算法的实用性和有效性.

关键词: 蚁群优化(ACO)算法; 机器人; 路径规划

中图分类号: TP391.9 文献标识码: A

Path planning for robots based on ant colony optimization algorithm under complex environment

FAN Xiaoping, LUO Xiong, YI Sheng, ZHANG Hang

(College of Information Science and Engineering, Central South University, Changsha 410083, China)

Correspondent: FAN Xiaoping, E-mail: xpfan@mail.csu.edu.cn

Abstract: In order to overcome the current difficulty of the existing algorithms in path planning for robots, the ant colony optimization algorithm is introduced into this new application area. The corresponding ant colony optimization algorithm is presented to solve the continuous function optimization problem containing constraint conditions. The numerical simulation shows the efficiencies of the algorithm.

Key words: ant colony optimization (ACO) algorithm; robot; path planning

1 引言

路径规划算法是实现机器人控制和导航的基础之一. 一般可将路径规划算法分为全局规划和局部规划两大类. 通过两类算法的结合使用, 可有效地实现机器人的路径规划. 在全局规划算法领域中, 目前使用的方法包括 Petri 网算法、基于数据融合的模糊规划、神经网络算法、人工势场法、计算几何法和遗传算法等^[1]. 但上述算法在复杂的工作环境下进行路径规划时, 会存在一些明显不足. 例如, 算法的计算代价过大, 有时甚至得不到最优解; 对于遗传算法和一些近似算法而言, 在初始可行解的有效构造以及针对复杂环境特点设计相应的遗传算子等方面, 存在着较大的困难. 为克服这些算法的不足, 特

别是对包含大量非规则障碍物的复杂环境中机器人轨迹规划问题, 本文使用蚁群优化(ACO)算法对其进行求解.

2 ACO 算法简介

ACO 算法是 Colnmi 和 Dorigo 等在 20 世纪 90 年代初提出的一种新型分布式智能模拟仿生类算法^[2~4], 它模拟和借鉴了现实世界中蚂蚁种群的行为特征. 虽然它仅提出 10 多年, 但已逐渐引起广大学者的注意, 并得到广泛的应用.

生物学家发现自然界中的蚂蚁群在觅食过程中具有一些显著的特征, 例如: 蚂蚁在移动过程中会释放一种称为信息素的物质; 释放的信息素会随着时间的推移而逐步减少; 蚂蚁能在一个特定的范围内

收稿日期: 2003-01-03; 修回日期: 2003-04-09

基金项目: 国家自然科学基金资助项目(69975003).

作者简介: 樊晓平(1961—), 男, 浙江绍兴人, 教授, 博士生导师, 从事智能控制、机器人控制等研究; 罗熊(1976—), 男, 湖南长沙人, 博士生, 从事智能控制、机器人控制等研究.

觉察出是否有同类的信息素轨迹存在; 蚂蚁会沿着信息素轨迹多的路径移动等。正是基于这些基本特征, 蚂蚁能找到一条从蚁巢到食物源的最短路径。

这个寻路过程可通过图 1 说明。其中图 1(a) 表示蚂蚁在没有障碍物的路径上移动; 图 1(b) 表示在蚂蚁移动的路径上出现了障碍物, 此时的蚂蚁群以相等的概率从障碍物的左右两侧通过; 图 1(c) 表示蚂蚁在寻找最短路径过程中的一个中间状态, 此时蚂蚁分泌的信息素在通过障碍物的较短路径上(即图中障碍物的左侧)会沉积得更多; 图 1(d) 表示蚂蚁群选路的最终状态, 所有的蚂蚁都会选择绕过障碍物的较短路径。

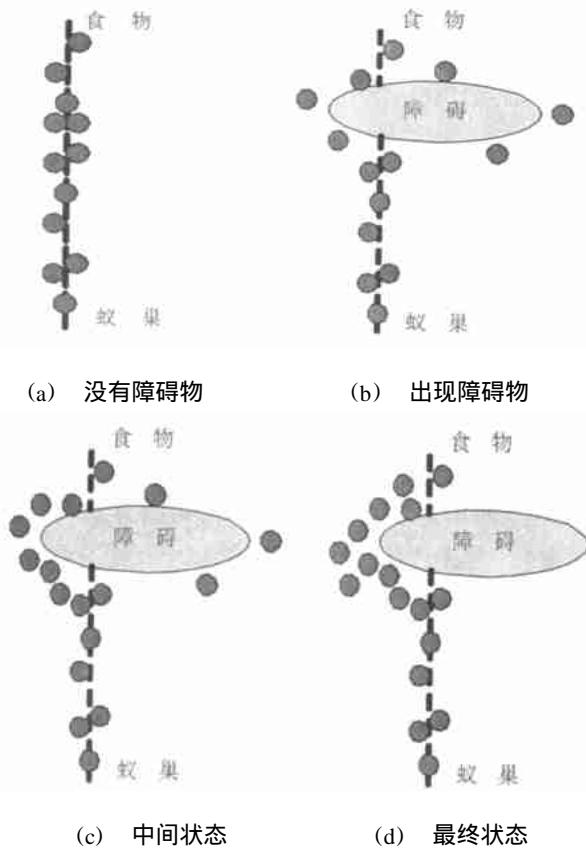


图 1 蚁群寻找最短觅食路径

通过分析上面的寻路过程可以看出, 蚂蚁的行为实质上是一种通过简单个体的自组织行为所体现出来的群体行为。ACO 算法有两个重要的特征: 其一是蚂蚁群体行为中所展现出来的正反馈过程。正是有了这个正反馈, 才使得整个系统得以演化发展。在此过程中, 反馈的信息是全局信息, 通过反馈机制进行调整, 可对系统中的较优解起到一个自增强的作用。从而使得问题的解向着全局最优的方向不断演变, 最终能有效地获得全局的相对较优解。这是 ACO 算法最重要的一个特征。其二是具有的分布式

并行计算能力。它可使算法在全局的多点同时进行解的搜索, 有效地避免了陷入局部最优解的可能性。从而使得 ACO 算法在求解非线性问题方面, 具有其他算法无法比拟的优良效果和高度的鲁棒性。

ACO 算法特别适合求解状态随时间变化的大规模组合优化问题, 如著名的旅行商问题、二次分配问题和作业调度问题等^[3,5]。除离散型组合优化问题外, ACO 算法也可用于求解连续型的函数优化问题, 即一般的函数优化问题^[6,7], 但它主要针对无约束的情形。对于有约束的情形, 由于条件的加强以及缺乏 ACO 算法的数学理论基础, 目前尚无相应的 ACO 算法^[6]。

目前, 对 ACO 算法的研究多集中于 ACO 算法中最优参数的选取, ACO 算法与其他优化搜索算法的融合利用, 某些经典问题求解, ACO 算法与现有算法的比较等。在形成系统的分析方法与成熟的数学理论上, 算法应用领域的拓展及与其他相关学科的交叉, 以及更深层次的应用(如基于 ACO 算法的蚂蚁智能体硬件)等方面, 还需要做大量的工作。

3 基于 ACO 算法的机器人路径规划

复杂工作环境中的机器人(最优时间)路径规划问题, 严格地说, 是一个带约束条件的连续函数优化问题, 因此目前还没有相应的 ACO 算法。设计解决此类问题的 ACO 算法, 是一个很有意义又极具挑战性的课题。

3.1 预备工作

在求解的过程中, 为了对蚁群的行为进行仿真, 引入以下描述符号: W 为蚁群中蚂蚁的个数; $d_{i,j}$ ($i = 1, 2, \dots, n_1; j = 1, 2, \dots, n_2; n_1$ 和 n_2 分别是对平面工作环境的 2 维划分维数) 为平面环境中位置点 i 与 j 之间的距离; $b_i(t)$ 为 t 时刻位于位置点 i 处的蚂蚁数目; $\tau_{i,j}(t)$ 表示 t 时刻在路径 i,j 上残留的信息素轨迹的量。显然, 有等式 $W = \sum_{i=1}^n b_i(t)$ 。因为在初始时刻, 每条路径的信息素轨迹的量都是相等的, 所以有预设条件 $\tau_{i,j}(0) = C$ (C 是一定常数)。

蚂蚁 k ($k = 1, 2, \dots, W$) 在运动过程中, 会根据各条路径上的信息素轨迹量决定其下一步的转移方向。在时刻 t 时, 蚂蚁 k 要从位置点 i 向 j 转移, 其对应的转移概率可定义为

$$p_{i,j}^k(t) = \begin{cases} \frac{\tau_{i,j}^k(t) \eta_{i,j}^k(t)}{\sum_r \tau_{i,r}^k(t) \eta_{i,r}^k(t)}, & j \in S_i^k; \\ 0, & \text{否则} \end{cases} \quad (1)$$

式中: $\eta_j(t)$ 是能见度的局部启发式函数(在该问题中定义为 $1/d_{i,j}$); 参数 α 和 β 分别表示 $\tau_{i,j}(t)$ 和 $\eta_j(t)$ 对整个转移概率的影响权值; S^k 表示蚂蚁 k 在位置点 i 处的可行邻域(即与点 i 相邻且尚未被蚂蚁 k 访问过的位置点的集合), 借助于种群的记忆功能, 这个集合在进化过程中将会不断地动态调整

随着时间的推移, 信息素轨迹将会逐渐地挥发 这里, 用 ρ 表示在某条路径上信息素轨迹挥发后的剩余度 在经过 h 个时刻后, 蚁群会完成一个循环的移动 此时, 各条路径上信息素轨迹的量将按照

$$\tau_{i,j}(t+h) = \rho\tau_{i,j}(t) + \sum_{k=1}^W \Delta\tau_{i,j}^k \quad (2)$$

全局调整准则进行调整 式中: $\Delta\tau_{i,j}^k$ 表示蚂蚁 k 在本次循环中留在路径 i,j 上的信息素轨迹的量, 可基于局部调整准则对其定义为

$$\Delta\tau_{i,j}^k = \begin{cases} Q/L_k, & \text{如果蚂蚁 } k \text{ 在本次循环} \\ & \text{中经过了路径 } i,j; \\ 0, & \text{否则} \end{cases} \quad (3)$$

式中: Q 为表示信息素轨迹强度的一定常数; L_k 表示蚂蚁 k 在本次循环中经过的所有路径的长度

在初始时刻, 有 $\tau_{ij}(0) = C, \Delta\tau_{i,j}^k = 0 (i = 1, 2, \dots, n_1; j = 1, 2, \dots, n_2; k = 1, 2, \dots, W)$. 此外, $\tau_{i,j}(t), \Delta\tau_{i,j}^k$ 和 $p_{i,j}^k(t)$ 的表达式也可根据算法的具体应用而有所调整

3.2 具体算法

整个求解算法的步骤为:

Step 1: 产生初始时刻的蚂蚁种群移动路径 根据移动过程中途经各点周围的距离启发式信息概率, 产生多条从起点到终点的可行移动路径, 每一条路径代表了一只蚂蚁的爬行轨迹

Step 2: 信息素的调整 对所产生的每一条可行移动路径, 分别计算路径的长度和所对应信息素的增量, 再采用设计的信息素轨迹更新函数(如式(2))对路径上各点所对应的信息素进行更新

Step 3: 对产生的每一可行路径进行一定的修正处理 在这里, 是将蚂蚁所走的弯曲路径逐段拉直为一条由直线段连接的可行路径(即成为一折线). 将此可行路径与记录的目前最短路径进行比较, 如果路径长度更小, 则用该路径替换最短路径 对路径上的所有点的信息素也根据 Step 2 中的方法进行更新 如果当前时刻已达到预先设定的终止时刻, 则转 Step 5

Step 4: 下一时刻蚂蚁路径的产生 综合使用当前点周围的距离启发式信息概率和基于信息素轨迹

的转移概率(如式(1)), 产生由起点到终点的可行路径, 并转 Step 2

Step 5: 算法结束 将当前路径作为最短路径输出

注 1(距离启发式信息概率的定义) Step 1 与 Step 4 中出现的距离启发式信息概率定义为

$$\varphi_{ij} = \begin{cases} \frac{(\text{MaxDistance}_{A(i),e} - \text{Distance}_{j,e})(\omega + \mu)^\lambda}{(\text{MaxDistance}_{A(i),e} - \text{Distance}_{j,e})(\omega + \mu)^{\lambda'}} & j \in A \text{ available}(i) \\ j \in A \text{ available}(i); & \\ 0, & \text{否则} \end{cases} \quad (4)$$

式(4) 决定了只依赖于距离信息时, 编号为 k 的蚂蚁从当前点 i 向其周围一点的移动概率 $A \text{ available}(i)$ 是点 i 周围 1 个单位距离内非障碍区中点的集合, 算法中蚂蚁只能向前、后、左、右 4 个方向移动, 因此 $0 < |A \text{ available}(i)| \leq 4$ $\text{Distance}_{j,e}$ 是从 j 点到终点的距离, 其值在算法执行前被预先计算出 $\text{MaxDistance}_{A(i),e}$ 是 $\text{Distance}_{j,e}$ 中的最大值 因为 $A \text{ available}(i)$ 的各个 $\text{Distance}_{j,e}$ 之间相差不超过 2, 所以需要对 $\text{Distance}_{j,e}$ 重新定标以体现它们之间的差别, 否则启发式概率将变成一个随机函数, 达不到应有的启发效果 这里引入 3 个标定系数 ω, μ 和 λ 根据多次试验, 算法中取 $\omega = 10, \mu = 2, \lambda = 2$

注 2(权值系数 α 和 β 的确定) 如前所述, α 和 β 两个参数分别决定了信息素轨迹和启发式函数(能见度)的相对重要性 在一般的蚂蚁算法中, α 和 β 是常数, 在算法执行过程中不作改变 但在路径规划问题中, 由于蚂蚁可经过的点太多(在一个 400×200 位图表示的环境中有 80 000 个点, 而研究旅行商问题一般达到几百规模就已经非常大了), 很难确保每个点都获得信息素 这样将带来一个严重的问题, 即如果获得信息素的点越少, 那么结果陷入局部解的可能性就越大 仿真实验发现将 α 和 β 设为常数后, 通常情况下不能找到较好的解 因此在本文设计的ACO 算法中, α 和 β 将随时间变化而调整^[8], 即

$$\alpha = \begin{cases} 4q/m, & 0 < q < m; \\ 4, & m \leq q \leq u; \\ (3m - 1.5q)/m, & 0 < q < m; \end{cases} \quad (5)$$

$$\beta = \begin{cases} 1.5, & m \leq q \leq u. \end{cases} \quad (6)$$

式中: m 为临界时刻, 在 m 时刻前, 由于各点上的信息量较少, 蚂蚁寻路过程中的主导因素为启发式因素(即基于式(4) 所表示的概率), 这样能使更多的点获得信息素; 在 m 时刻后, 蚂蚁寻路过程中的主导因素变为信息素因素(即基于式(1) 所表示的概

率). 从初始时刻 0 到临界时刻 m , α 值随时间线性递增, β 值随时间线性递减; 从临界时刻 m 到终止时刻 n , α 和 β 值均为常数, 且 $\alpha > \beta$

注 3(对可行路径的修正处理) 在 Step3 中, 对可行路径进行了一定的修正处理, 目的是基于以下考虑: 因为算法中的蚂蚁是在所设定的由像素点构成的位图环境中爬行, 当位图环境中的像素点过多(密)时, 蚂蚁在某段特定距离内的爬行路径可能变成曲线段, 从而人为地造成移动路径长度的增加. 为避免这种情况的出现, 可人为地将蚂蚁在某段特定距离内弯曲的爬行路径“拉直”为一直线段, 从而减少移动路径的长度.

注 4(距离启发式信息概率和转移概率的综合使用) 在 Step4 中提到从当前点到下一个可行点的转移是由距离启发式信息概率和基于信息素轨迹的转移概率综合决定的. 这里所使用的综合决定方法是基于比例选择(即赌盘)策略. 通过比例选择, 可交替使用 3 种概率: 式(1)所示的基于信息素轨迹的转移概率, 式(4)所示的距离启发式信息概率和综合考虑上述两种概率所产生的转移概率, 以决定下一个具体的可行移动点.

4 仿真研究

下面给出一个仿真示例. 取蚂蚁的数目 $w = 10$, 临界时刻 $m = 10$, 信息素挥发的剩余度 $\rho = 0.7$. 图 2 是一个给定的大小为 400×200 像素的环境位图(图中有大量的不规则障碍物, 左上角和右下角的小圈分别表示起点和终点), 并表示 10 只蚂蚁在第 10 时刻累积的移动路径曲线. 从图 2 可以发现, 大多数蚂蚁所选择的路径都位于连接起点和终点线段两侧的一个限定范围内, 这使得该区域内的信息量高于其他区域. 图 3 是第 50 时刻累积的蚁群移动路径曲线. 与图 2 相比, 越来越多的蚂蚁的移动路径落入了连接起点和终点线段两侧的限定区域内, 这导致其中所含的信息量渐渐高于其他区域. 图 4 表示在第 120 时刻蚁群最终的收敛移动路径. 所有蚂蚁经过的点所对应的信息素轨迹用灰度表示, 而实线表示一条已被修正处理过的最短移动路径. 图 5 表示最短移动路径长度的收敛曲线. 由该图可以发现, 通过使用所设计的 ACO 算法, 在第 70 时刻就能得到收敛的路径解, 而从第 70 ~ 120 时刻没有得到更好的解, 这表明算法的收敛效果趋于稳定.

5 结 语

ACO 算法是一种极有发展前途的智能仿生类

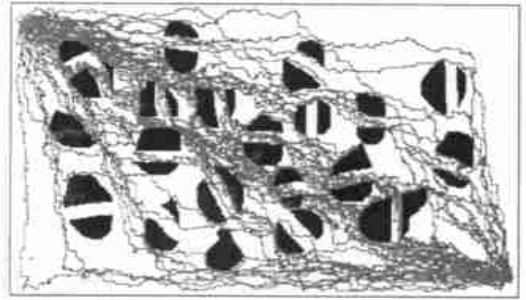


图 2 在第 10 个时刻累积的蚁群移动路径曲线

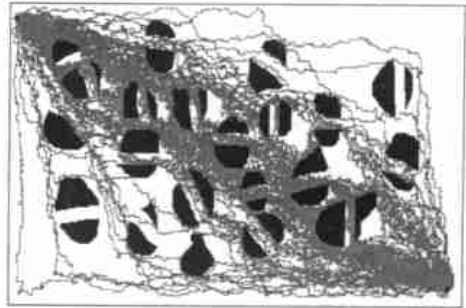


图 3 在第 50 个时刻累积的蚁群移动路径曲线

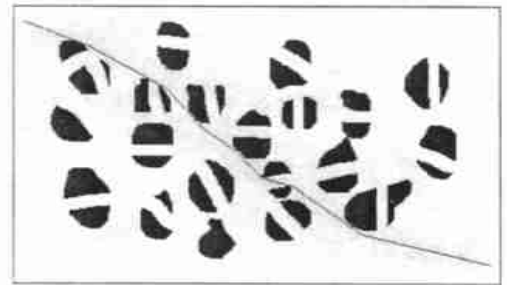


图 4 在第 120 个时刻蚁群最终的收敛移动路径曲线

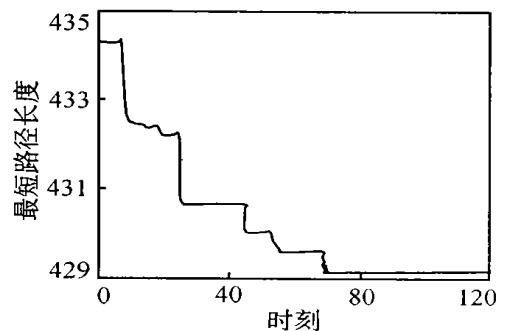


图 5 最短移动路径长度的收敛曲线图

优化算法, 已在组合优化领域中得到了广泛的应用^[9]. ACO 算法除了具有一般模拟演化算法所具有的全局最优性、并行性等普遍的特点之外, 还具有个体间高效的通信协作方式以及与问题实例信息的有效耦合等独特的特点^[10]. 在分析了 ACO 算法后, 本文将 ACO 算法引入机器人路径规划这一新的应用领域, 设计了相应的 ACO 算法, 解决了以前尚未涉

足的带约束条件的连续函数优化问题 本文设计的算法,在距离启发式信息概率的构造、重要权值系数的动态确定、可行解的有效构造以及在移动过程中为蚂蚁选定转移概率等方面,均做了一些创造性的工作 仿真实验结果验证了所设计算法的实用性和有效性

参考文献(References):

- [1] Abe Y, Shikann M, Fokuda T, et al Vision based navigation system by variable template matching for autonomous mobile robot[A]. *Proc of IEEE Int Conf on Robotics & Automation*[C]. Leaven, 1998 952-957.
- [2] Colnri A, Dorigo M, Maniezzo V. Distributed optimization by ant colonies[A]. *Proc of the ECAL-91* [C]. Paris, 1991. 134-142
- [3] Dorigo M, Maniezzo V, Colnri A. The ant system: Optimization by a colony of cooperating agents [J]. *IEEE Trans on Systems, Man & Cybernetics-B*, 1996, 26(2): 29-41
- [4] Dorigo M, Gambardella L M. Ant colony system: A cooperative learning approach to the travelling salesman problem [J]. *IEEE Trans on Evolutionary Computation*, 1997, 1(1): 53-66
- [5] 马良 来自昆虫世界的寻优策略——蚂蚁算法[J]. 自然杂志, 1999, 21(3): 161-163
- (Ma Liang Ant algorithm——Optimization strategy from the insect colony [J]. *Nature Magazine*, 1999, 21(3): 161-163)
- [6] 马良 全局优化的一种新方法[J]. 系统工程与电子技术, 2000, 22(9): 61-62
- (Ma Liang A new method for global optimization [J]. *Systems Engineering and Electronics*, 2000, 22(9): 61-62)
- [7] 魏平,熊伟清 用于一般函数优化的蚁群算法[J]. 宁波大学学报, 2001, 14(4): 52-55
- (Wei Ping, Xiong W Q. Ant colony algorithm for general function optimization problems [J]. *J of Ningbo University*, 2001, 14(4): 52-55)
- [8] 易晟 平面机器人路径规划研究[D]. 长沙: 中南大学, 2002
- [9] Bonabeau E, Dorigo M, Theraulaz G. Inspiration for optimization from social insect behaviour [J]. *Nature*, 2000, 406(6791): 39-42
- [10] 李有梅,王文剑,徐宗本 关于求解难组合优化问题的蚁群优化算法[J]. 计算机科学, 2002, 29(3): 115-118
- (Li Y M, Wang W J, Xu Z B. About ACO algorithm for hard combinatorial optimization problem [J]. *Computer Science*, 2002, 29(3): 115-118)

(上接第 165 页)

参考文献(References):

- [1] Shaukat A B, Luan L L. Heuristics for scheduling in a flow shop with multiple processors [J]. *European J of Operational Research*, 1999, 113(1): 113-122
- [2] Kochlar S, Morris R J T. Heuristic methods for flexible flow line scheduling [J]. *J of Manufacturing Systems*, 1987, 6(4): 299-314
- [3] Wittrock R J. Scheduling algorithms for flexible flow lines [J]. *IBM J of Research and Development*, 1985, 29(4): 401-412
- [4] 方述诚,汪定伟 模糊数学与模糊优化[M]. 北京: 科学出版社, 1997.
- [5] Grzegorzewski Nearest interval approximation of a fuzzy number [J]. *Fuzzy Sets and System*, 2002, 130(3): 321-330
- [6] Lai L J, Huang C L. *Fuzzy Mathematical Programming* [M]. Berlin: Springer-Verlag, 1992