

# 第 3 章



## 蚁群算法

蚁群算法(ant colony algorithm, ACA)算法是 1991 年由意大利 M. Dorigo 博士等提出的一种群智能优化算法,它模拟了蚁群能找到一条由蚁穴到食物源的最短路径的觅食行为,并成功用于求解旅行商问题。后来,一些研究者把蚁群优化算法改进并应用于连续优化问题。

2008 年, Dorigo 等又提出了一种求解连续空间优化问题的扩展蚁群优化(extension of ant colony optimization,  $ACO_R$ )算法,通过引入解存储器作为信息素模型,使用连续概率分布取代 ACO 算法中的离散概率分布,将基本蚁群算法的离散概率选择方式连续化,从而将其拓展到求解连续空间优化问题。

本章首先介绍了蚁群算法的基本原理,然后分析了蚁群算法的几个典型应用。

### 3.1 蚁群算法思想及特点

#### 3.1.1 算法思想

蚂蚁作为一个生物个体,其自身的能力是十分有限的,如蚂蚁个体是没有视觉的,蚂蚁自身体积又是那么渺小,但是由这些能力有限的蚂蚁组成的蚁群却可以做出超越个体蚂蚁能力的超常行为。蚂蚁没有视觉却可以寻觅食物,蚂蚁体积渺小而蚁群却可以搬运比它们个体大十倍甚至百倍的昆虫。这些都说明蚂蚁群体内部的某种机制使得它们具有群体智能,可以做到蚂蚁个体无法实现的事情。

生物学家经过长时间的观察发现,蚂蚁是通过分泌于空间中的信息素进行信息交流,进而实现群体行为的。

##### 1. 蚂蚁的习性与蚁群社会

蚂蚁是一种社会性昆虫,起源约在一亿年前。蚂蚁种类为 9000~15000 种,但无一独居,都是群体生活,建立了独特的蚂蚁社会。之所以说蚂蚁是一种社会性昆虫,是因为蚂蚁不但有组织、有分工,还有相互的信息的传递。蚂蚁有着独特的信息系统:视觉信号、声音通信和更为独特的无声语言——分泌化学物质信息素(pheromone)。

蚂蚁王国分工细致,职责分明。有专门产卵的蚁后;有为数众多,从事觅食打猎、屋穴兴建、后代抚育的工蚁;有负责守卫门户、对敌作战的兵蚁;还有专备蚁后招婚纳赘的雄蚁。蚁后产下的受精卵发育成工蚁或新的蚁后,而未受精的卵发育成雄蚁。

## 2. 蚂蚁觅食行为与信息素

昆虫学家研究发现蚂蚁有能力在没有任何可见提示下找出由蚁穴到食物源的最短路径,并能随环境变化而自适应地搜索新的路径。蚂蚁在从食物源到蚁穴并返回的过程中,能在走过的路径上分泌一种化学物质——信息素,通过这种方式形成信息素轨迹(或踪迹),蚂蚁在运动中能感知这种物质的存在及其强度,以此指导自己的运动方向。

蚂蚁之间通过接触提供的信息传递来协调其行动,并通过组队相互支援,当聚集的蚂蚁数量达到某一临界数量时,就会涌现出有条理的大军。蚂蚁的觅食行为完全是一种自组织行为,自组织地选择去往食物源的路径。

大量蚂蚁组成的集体觅食表现为一种对信息素的正反馈现象。某一条路径越短,路径上经过的蚂蚁越多,遗留的信息素越多,信息素浓度就越高,蚂蚁选择这条路径的概率也就越高,由此构成正反馈过程,从而逐渐地逼近最优路径,找到最优路径。

蚂蚁觅食的运行轨迹模式如图 3-1 所示,蚂蚁以信息素作为媒介间接进行信息交流,判断蚁穴到食物源的最佳路径。

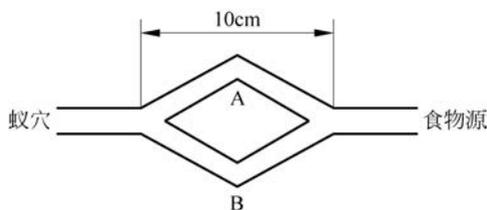


图 3-1 蚂蚁觅食的运动轨迹模式

当蚂蚁从食物源走到蚁穴,或者从蚁穴走到食物源时,都会在经过的路径上释放信息素,从而形成了一条含有信息素的路径,蚂蚁可以感觉出路径上信息素浓度的大小,并且以较高的概率选择信息素浓度较高的路径。

人工蚂蚁的搜索主要包括三种智能行为。

(1) 蚂蚁利用信息素进行相互通信。蚂蚁在所选择的路径上会释放一种叫作信息素的物质,当其他蚂蚁进行路径选择时,会根据路径上的信息素浓度进行选择,这样信息素就成为蚂蚁之间进行通信的媒介。

(2) 蚂蚁的记忆行为。一只蚂蚁搜索过的路径在下次搜索时就不再被该蚂蚁选择,因此在蚁群算法中建立禁忌表进行模拟。

(3) 蚂蚁的集群活动。通过一只蚂蚁的运动很难达到食物源,但整个蚁群进行搜索就完全不同。当某些路径上通过的蚂蚁越来越多时,路径上留下的信息素数量也就越多,导致信息素强度增大,蚂蚁选择该路径的概率随之增加,进一步增加该路径的信息素强度;而通过的蚂蚁比较少的路径上的信息素会随着时间的推移而挥发,从而变得越来越少。

蚁群优化算法是模拟蚂蚁觅食的原理,设计出的一种群集智能算法。算法寻优的快速性是通过正反馈式的信息传递和积累来保证的,而算法的早熟性收敛又可以通过其分布式计算特征加以避免,同时,具有贪婪启发式搜索特征的蚁群系统又能在搜索过程的早期找到可以接受的问题解。这种优越的问题分布式求解模式经过相关领域研究者的关注和努力,已经在最初的算法模型基础上得到了很大的改进和拓展。Dorigo 博士在 1993 年给出改进模型蚁群系统(ACS),其中改进了转移概率模型,并且应用了全局搜索与局部搜索策略来进行深度搜索。Hoos 给出了最大-最小蚂蚁系统(MMAS)。所谓最大-最小,就是为信息素设定上限与下限,设定上限避免搜索陷入局部最优,设定下限鼓励深度搜索。

### 3.1.2 算法特点

蚁群算法实现的重要原则包括以下几种。

#### 1. 避障原则

如果蚂蚁要移动的方向有障碍物挡住,它会随机地选择另一个方向,并且有信息素指引的话,它会按照觅食的规则选择方向。

#### 2. 播撒信息素原则

每只蚂蚁在刚找到食物或者蚁穴时散发的信息素最多,随着它走的距离越来越远,播撒的信息素越来越少。

#### 3. 范围原则

蚂蚁观察到的范围是一个方格世界,蚂蚁有一个参数为速度半径(一般是3),那么它能观察到的范围就是 $3 \times 3$ 个方格世界,并且能移动的距离也在这个范围之内。

#### 4. 移动原则

每只蚂蚁都朝信息素最多的方向移动,当周围没有信息素指引时,蚂蚁会按照自己原来运动的方向惯性地运动下去,并且在运动的方向有一个随机的小的扰动。为了防止原地转圈,蚂蚁会记住最近刚走过了哪些点,如果发现要走的下一点已经在最近走过了,它就会尽量避开。

#### 5. 觅食原则

在每只蚂蚁能感知的范围内寻找是否有食物,如果有就直接过去;否则看是否有信息素,并且比较在能感知的范围内哪一点的信息素最多,这样,它就朝信息素多的地方走,并且每只蚂蚁都会以小概率犯错误,从而并不是往信息素最多的点移动。蚂蚁找蚁穴的规则和上面一样,只不过它对蚁穴的信息素做出反应,而对食物信息素没反应。

#### 6. 环境

蚂蚁所在的环境是一个虚拟的世界,其中有障碍物,有别的蚂蚁,还有信息素。信息素有两种,一种是找到食物的蚂蚁撒下的信息素,每只蚂蚁仅能感知它本身撒下的食物信息素;另一种是找到蚁穴的蚂蚁撒下的蚁穴的信息素,每只蚂蚁仅能感知它范围内的环境信息,环境以一定的速率让信息素挥发。

根据这几条原则,蚂蚁之间并没有直接的关系,但是每只蚂蚁都和环境发生交互,从而通过信息素把每只蚂蚁关联起来。例如,一只蚂蚁找到了食物,它并没有直接告诉其他蚂蚁这里有食物,而是向环境播撒信息素,当其他蚂蚁经过它附近时,就会感觉到信息素的存在,进而根据信息素的指引找到食物。

蚁群算法是通过对生物特征的模拟得到的一种计算算法,其本身具有很多特点。

(1) 蚁群算法是一种本质上并行的算法。每只蚂蚁搜索的过程彼此独立,仅通过信息素进行通信。因此,蚁群算法可以看作一个分布式的多智能体系统,它在问题空间的多点同时开始进行独立的解搜索,不仅增加了算法的可靠性,也使得算法具有较强的全局搜索能力。

(2) 蚁群算法是一种自组织的算法。所谓自组织,就是组织力或组织指令来自系统的内部,区别于其他组织。如果系统在获得空间、时间或功能结构的过程中,没有外界的特定制约,则该系统是自组织的。简单地说,就是系统从无序到有序的变化过程。以蚂蚁群体优化为例,在算法开始初期,单个的人工蚂蚁无序地寻找解,算法经过一段时间的演化,人工蚂蚁间通过信息素的作用,自发地越来越趋向于寻找到接近最优解的一些解,这就是无序到有序的过程。

(3) 蚁群算法具有较强的鲁棒性。相对于其他算法,蚁群算法对初始路线要求不高,即蚁群算法的求解结果不依赖初始路线的选择,而且在搜索过程中不需要进行人工的调整。其次,蚁群算法的参数数目少,设置简单,易于蚁群算法应用到其他组合优化问题的求解。

(4) 蚁群算法是一种正反馈的算法。从真实蚂蚁的觅食过程中不难看出,蚂蚁能够最终找到最短路径,直接依赖最短路径上信息素的堆积,而信息素的堆积却是一个正反馈的过程。正反馈是蚁群算法的重要特征,它使得算法演化过程得以进行。

## 3.2 蚁群算法的应用

随着群智能理论和应用算法研究的不断发展,研究者已尝试将其用于各种工程优化问题,并取得了意想不到的收获。多种研究表明,群智能在离散求解空间和连续求解空间中均表现出良好的搜索效果,并在组合优化问题中表现突出。

蚁群优化算法并不是旅行商问题的最佳解决方法,但是它却为解决组合优化问题提供了新思路,并很快被应用到其他组合优化问题中。比较典型的应用研究包括网络路径优化、数据挖掘及一些经典的组合优化问题。

蚁群算法在电信路径优化中已取得了一定的应用成果。惠普公司和英国电信公司在 20 世纪 90 年代中后期都开展了有关这方面的研究,设计了蚁群路径(ant colony routing, ACR)算法。

像蚁群优化算法中一样,每只蚂蚁根据它在网络上的经验与性能,动态更新路径表项。如果一只蚂蚁因为经过网络中堵塞的路径而导致了比较大的延迟,那么就对该表项做较大的增强。同时根据信息素挥发机制实现系统的信息更新,从而抛弃过期的路径信息。

这样,在当前最优路径出现拥堵现象时,ACR 算法就能迅速地搜寻另一条可替代的最优路径,从而提高网络的均衡性、负荷量和利用率。目前这方面的应用研究仍在升温,因为通信网络的分布式信息结构、非稳定随机动态特性及网络状态的异步演化与蚁群算法的本质和特性非常相似。

基于群智能的聚类算法源于对蚁群蚁卵的分类研究。Lumerfaieta 和 Deneubourg 提出将蚁穴分类模型应用于数据聚类分析。其基本思想是将待聚类数据随机地散布到一个二维平面内,然后将虚拟蚂蚁分布到这个空间内,并以随机方式移动,当一只蚂蚁遇到一个待聚类数据时将之拾起并继续随机运动,若运动路径附近的数据与背负的数据相似性高于设置的标准则将其放置在该位置,然后继续移动,重复上述数据搬运过程。按照这样的方法可实现对相似数据的聚类。

蚁群算法(ACA)还在许多经典组合优化问题中获得了成功的应用,如二次分配问题(quadratic assignment problem, QAP)、机器人路径规划、作业流程规划、图着色(graph coloring)等问题。经过多年的发展,ACA 已成为能够有效解决实际二次规划问题的几种重要算法之一。利用 ACA 实现对生产流程和物料管理的综合优化,并通过与遗传、模拟退火和禁忌搜索算法的比较证明了 ACA 的工程应用价值。

### 3.2.1 蚁群算法在旅行商问题中的应用

旅行商问题(travelling salesman problem, TSP)是物流领域中的典型问题。它的求解具有十分重要的理论和现实意义。采用一定的物流配送方式,可以大幅节省人力物力,完善整个物流系统。

被广泛采用的遗传算法是旅行商问题的传统求解方法,但遗传算法收敛速度慢,具有一定的缺陷。

本节采用蚁群算法,充分利用蚁群算法的智能性,求解旅行商问题,并进行实例仿真。进行仿真计算是为了该算法能获得旅行商问题的优化结果、平均距离和最短距离。

在求解旅行商问题的蚁群算法中,每只蚂蚁相互独立,用于构造不同的路线,若干蚂蚁之间通过自适应的信息素值来交换信息,合作求解并不断优化。

利用蚁群算法求解旅行商问题的过程如下:

(1) 初始化。设迭代的次数为  $NC$ ,初始化  $NC=0$ 。

(2) 将  $n$  只蚂蚁置于  $n$  个顶点上。

(3)  $m$  只蚂蚁按概率函数选择下一座城市,完成各自的周游。每只蚂蚁按照状态变化规则逐步地构造一个解,即生成一条回路。蚂蚁的任务是访问所有的城市后返回到起点,生成一条回路。设蚂蚁  $k$  当前所在的顶点为  $i$ ,那么蚂蚁  $k$  由点  $i$  向点  $j$  移动要遵循规则而不断迁移,按不同概率来选择下一点。

(4) 记录本次迭代最佳路线。

(5) 全局更新信息素值。应用全局信息素更新规则来改变信息素值。当所有  $m$  只蚂蚁生成了  $m$  个解,其中有一条最短路径是本代最优解,将属于这条路线上的所有弧相关联的信息素值进行更新。全局信息素更新的目的是在最短路线上注入额外的信息素,即只有位于最短路线的弧上的信息素才能得到加强,这是一个正反馈的过程,也是一个强化学习的过程。在各弧上,伴随着信息素的挥发,全局最短路线上各弧的信息素值得到增加。

(6) 终止。若终止条件满足,则结束;否则  $NC=NC+1$ ,转入步骤(2)进行下一代进化。终止条件可指定进化的代数,也可限定运行时间,或设定最短路长的下限。

(7) 输出结果。详细 MATLAB 代码见目录二维码中的附录 3。

蚁群算法优化路径和各代最短距离与平均距离对比分别如图 3-2 和图 3-3 所示。

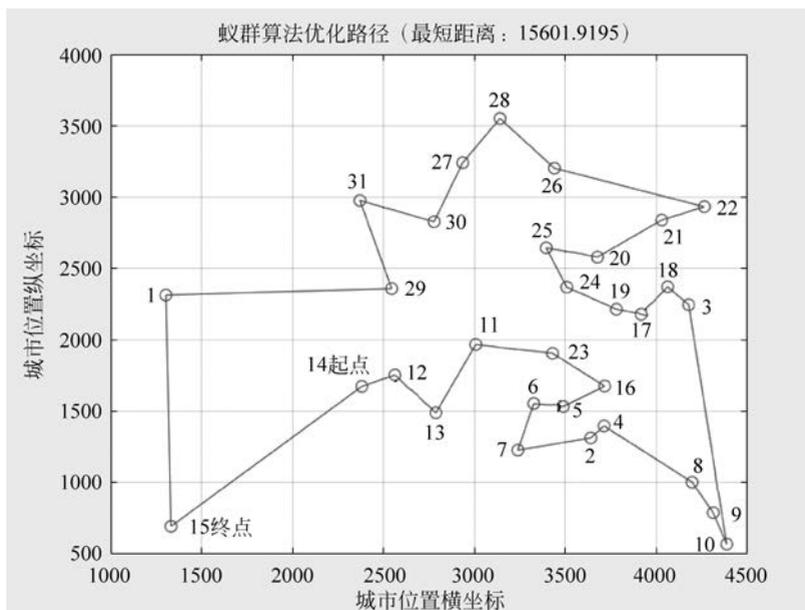


图 3-2 蚁群算法优化路径

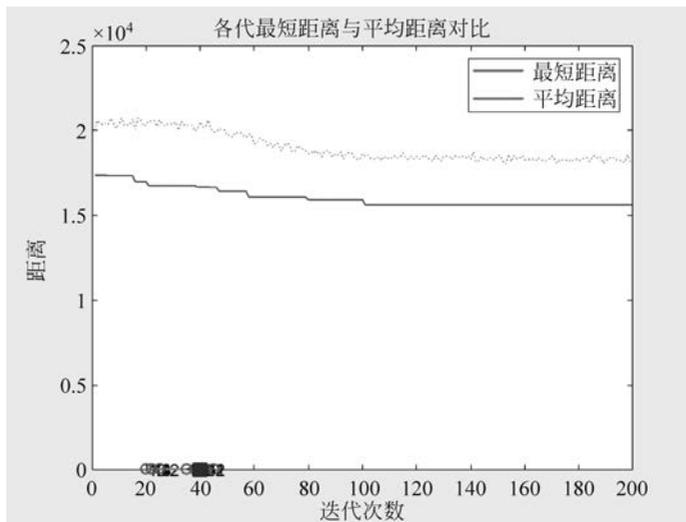


图 3-3 各代最短距离与平均距离对比(见彩插)

### 3.2.2 蚁群算法在函数极值问题中的应用

本案例的寻优函数如下:

$$f(x, y) = 20(x^2 - y^2)^2 - (1 - y^2)^2 - 3(1 + y)^2 = 0.3, \quad x \in [-5, 5], y \in [-5, 5]$$

求解步骤如下:

步骤 1: 初始化蚂蚁只数  $m = 300$ , 最大迭代次数  $\text{iter\_max} = 80$ , 信息素挥发因子  $\text{Rho} = 0.9$ , 转移概率常数  $P_0 = 0.2$ , 局部搜索步长  $\text{step} = 0.05$ 。

步骤 2: 随机产生蚂蚁初始位置, 计算适应度函数值, 设为初始信息素, 计算状态转移概率。其计算公式如下:

$$P(\text{iter}, i) = \frac{\max(\text{Tau}) - \text{Tau}(i)}{\max(\text{Tau})}$$

其中,  $\max(\text{Tau})$  表示信息素的最大值,  $\text{Tau}(i)$  表示蚂蚁  $i$  的信息素,  $P(\text{iter}, i)$  表示第  $\text{iter}$  次迭代蚂蚁  $i$  的转移概率值。

步骤 3: 进行位置更新, 当状态转移概率小于转移概率常数时, 进行局部搜索, 搜索公式为  $\text{new} = \text{old} + r_1 \cdot \text{step} \cdot \lambda$ , 其中  $\text{new}$  为待移动的位置,  $\text{old}$  为蚂蚁当前位置,  $r_1$  为  $[-1, 1]$  的随机数,  $\text{step}$  为局部搜索步长,  $\lambda$  为当前迭代次数的倒数; 当状态转移概率大于转移概率常数时, 进行全局搜索, 搜索公式为  $\text{new} = \text{old} + r_2 \cdot \text{range}$ , 其中  $r_2$  为  $[-0.5, 0.5]$  的随机数,  $\text{range}$  为自变量的区间大小。

通过判断待移动位置的函数值与当前位置函数值的大小来确定是否更新蚂蚁当前的位置, 并利用边界吸收方式进行边界条件处理, 将蚂蚁位置界定在取值范围内。

步骤 4: 计算新的蚂蚁位置的适应度值, 判断蚂蚁是否移动, 更新信息素, 更新公式为  $\text{Tau} = (1 - \text{Rho}) \cdot \text{Tau} + f$ , 其中  $\text{Rho}$  为信息素挥发因子,  $\text{Tau}$  为信息素,  $f$  为目标函数值。

步骤 5: 判断是否满足终止条件。若满足, 则结束搜索过程, 输出优化值; 若不满足, 则继续进行迭代优化。

具体 MATLAB 代码见目录二维码中的附录 4。

### 3.3 本章小结

蚁群算法是受自然界中蚁群搜索食物行为的启发而提出的一种智能优化算法。通过分析蚁群觅食过程中基于信息素的最短路径的搜索策略,本章首先介绍了蚁群算法的起源、基本原理等,然后介绍了如何用 MATLAB 实现蚁群算法编程,最后重点介绍了蚁群算法在旅行商问题和函数极值问题中的应用。

### 3.4 习题

1. 简述蚁群算法的概念。
2. 简述蚁群算法有哪些特点。
3. 简述蚁群算法求解过程。
4. 蚁群算法的设计原则有哪些?
5. 什么是蚁群算法的自组织性?
6. 信息素有何作用?