

5.1 蚁群算法起源

5.1.1 蚁群算法生物学基础

蚁群算法是一种模拟蚂蚁群体智能行为的随机优化算法,它试图模仿蚂蚁在觅食活动中找到最短路径的过程。

蚂蚁是自然界的一种群居动物,设想一个蚂蚁觅食的情景,假设蚁巢到食物是一条直线,即如图 5.1 所示的情况,那么蚂蚁将在这条直线上来回移动搬运食物。假设此时这条直线路径上出现了一个障碍,那么蚂蚁将会产生如图 5.2 所示的移动轨迹。

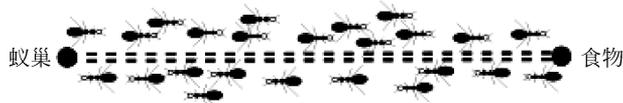


图 5.1 蚂蚁觅食路径示意图

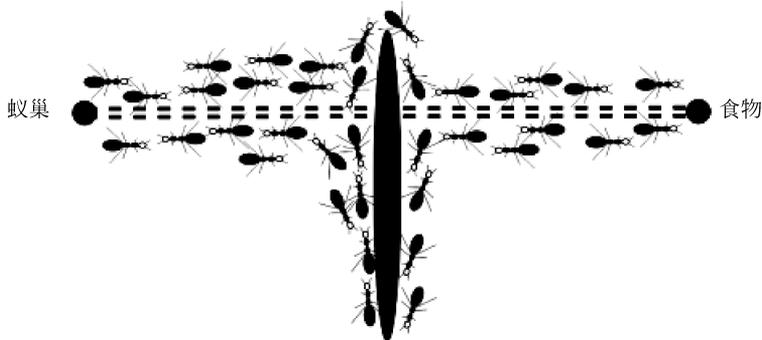


图 5.2 放置障碍后蚂蚁觅食路径示意图

Deneubourg 等在 1989 年就进行过这样一个实验,将蚁巢和食物置于两端,中间设置不同长度的路径或障碍,观察蚁群的行为路线。实验结果表明,在经过长时间的移动之后,蚁群会在最短的路径上持续进行食物的搬运,而较长的路径将不会有蚂蚁移动,大致情形如图 5.3 所示。



图 5.3 一段时间后蚂蚁觅食路径示意图

众所周知,两点间直线距离最短,但蚂蚁们显然不具备这样的视力和智慧。它们无法从远处看到食物源,也无法计划一个合适的路径来搬运食物。蚂蚁们采用的方法是全体在周围区域进行地毯式搜索,而它们之间的联系方式是在爬过的路径上分泌化学物质,这种化学物质叫信息素。大量研究表明,蚂蚁在寻找食物的过程中,会在所经过的路径上释放信息素,当它们碰到一个还没有走过的路口时,会随机地挑选一条路径前行,与此同时释放出与路径长度有关的信息素。路径越长,释放的激素浓度越低。后来的蚂蚁通过感知这种物质以及强度,会倾向于朝信息素浓度高的方向移动,同时移动留下的信息素会对原有的信息素进行加强。由于较短路径积累的信息素快、浓度值高,这样经过一段时间后,利用整个群体的自组织就能够发现最短的路径。蚂蚁个体之间就是通过这种间接的通信机制达到协同搜索食物最短路径的目的。

5.1.2 蚁群算法发展历程

蚂蚁算法^[1]是由意大利学者 M. Dorigo 在蚂蚁寻找从巢穴到食物源的路径的现象中受到启发而提出的。1992 年, M. Dorigo 在博士学位论文中首次提出了蚁群优化算法模型: 蚂蚁系统 (Ant System, AS)。但是在随后的几年时间里, 蚂蚁算法并没有得到学术界的广泛关注。直到 1996 年, M. Dorigo 等发表了一篇名为 *Ant system: Optimization by a colony of cooperating agents* 的论文^[2], 详细介绍了蚁群算法的基本原理以及数学模型, 并且对算法参数的影响做了简单介绍, 拓展了求解问题的领域。自此以后, 该算法受到越来越多的学者和专家的关注和研究, 理论基础不断完善, 并逐渐渗透到其他领域问题的解决上, 例如 TSP、加工调度问题、车辆路由、图像处理、数据挖掘等。

随着各界学者对蚁群算法不断高涨的研究热情, 算法研究也广泛开展, 有一些突破性的研究成果有力地推动了蚁群算法的迅速发展。Gambardella 和 Dorigo 为了进一步改善 AS 算法的性能, 克服其搜索时间长、收敛速度慢、容易出现停滞现象等缺点, 提出了 Ant-Q 算法^[3]; 之后 Dorigo 等提出了带有精英策略的蚂蚁系统 (Ant Colony System, ACS)^[4]; 德国学者 Stutzle 和 Hoos 提出了最大最小蚂蚁系统 (Max-Min Ant System, MMAS)^[5]; Bullnheimer 等提出了基于排序的蚂蚁系统 (Rank-based Ant System, RAS)^[6]; Gambardella 等提出了混合型蚁群算法^[7]; Cordon 等提出了最优-最差蚂蚁系统^[8]。

国内学者对蚁群算法的研究起步较晚, 大多都是在现有的成果基础上进行改进, 提出一些修正方案来弥补蚁群算法自身的不足, 或者是将其他先进算法的思想融合进来。吴庆洪

等提出了具有变异特征的蚁群算法^[9],有机地结合了 2-opt 方法,提高了算法的搜索速度。陈焯将遗传算法中的杂交算子引入蚁群算法^[10],有效地改善了算法的收敛速度慢、易陷入局部最优的缺陷。陈峻等提出了一种基于分布均匀度的自适应蚁群算法^[11],可以在加速收敛、早熟以及停滞现象之间取得平衡。黄国锐等提出了信息素扩散模型^[12],并且仿真表明该算法取得了一定的效果。秦海生等提出了一种动态局部搜索蚁群优化算法^[13],局部搜索技术提高解的质量,动态策略用于更新信息素,不同的路径使用不同的信息素更新策略。

蚁群算法作为一种智能模拟进化算法,具有正反馈性、鲁棒性和分布式等优点,在复杂的组合优化问题的求解上表现出了一定的优越性。最初该算法用来解决经典的组合优化-旅行商问题,并且取得了较为显著的效果。由于蚁群算法具有强大的优势,因此对其的研究已由当初单一的 TSP 领域渗透到多个应用领域,从解决一维静态优化问题到多维组合优化问题,由解决离散域范围的问题到连续域范围的问题,并在蚁群算法的模型改进及与其他启发式随机优化算法的结合等方面也获得了突破性进展^[14-17]。

目前,蚁群算法已成为国际智能计算领域关注的热点和前沿课题。1998 年在布鲁塞尔专门召开了第一届 ACO 国际研讨会,以后每两年召开一次。应当指出,现阶段对蚁群算法的研究大多还停留在仿真阶段,尚未能提出一个完善的理论分析,对它的有效性也没有给出严格的数学解释。尽管蚁群算法的严格理论基础尚未奠定,但这种新兴的智能进化仿生算法已展现出勃勃生机。

5.2 蚁群算法实现

5.2.1 蚁群算法流程

蚁群算法开始随机地选择搜索路径,在寻优过程中通过增强较好解,使搜索过程逐渐变得规律,从而逐渐逼近直至最终达到全局最优解。蚁群算法通过适应阶段和协作阶段进行寻优。在初步的适应阶段,一群人工蚁按照虚拟信息素和启发式信息的指引,在解空间不断地寻求外界信息来构造问题的解,同时它们根据解的质量在其路径上留下相应浓度的信息素。在后期的协作阶段,其他人工蚁选择信息素浓度高的路径前进,同时在这段路径上留下自己的信息素,通过这种正反馈的信息交流机制找到解决问题的高质量的解。

为了避免蚁群算法出现早熟现象,算法利用信息素的挥发实现负反馈机制。但是挥发的强度不能太弱,否则无法防止早熟现象的产生;挥发强度也不能太强,否则将抑制个体间的协作过程。

蚁群算法作为一种新的启发式算法具有以下特征^[18]。

(1) 通用的(versatile)。它可以应用于求解多种组合优化问题。对不同问题,不需要或仅需做少量的改动就可直接应用,例如,此算法既可以直接应用于 TSP,也可不需改动地应用于不对称 TSP(ATSP)。

(2) 鲁棒的(robust)。它的性能不因组合优化问题的不同而不同。

(3) 一种基于群体的方法(a population based approach)。它由多个个体(multiagent)组成,个体之间可以互相交流信息,并互相影响,使整体达到某个目的。这个特征使它可以使正反馈作为一种搜索机制,也适合应用于并行系统。

蚁群优化算法已被成功地用于求解组合优化的近似最优解^[19]。为了说明蚁群系统模型,引入典型的组合优化-旅行商问题。假设有 n 座城市,并且知道城市两两之间的距离,要求确定一条经过各个城市当且仅有一次并回到出发城市的最短路线。

蚁群优化算法解决 TSP 的基本要素包括路径构建和信息素更新两部分。首先,在构建路径的过程中,每只蚂蚁都随机选择一个城市作为其出发城市,并维护一个路径记忆向量,用来存放该蚂蚁依次经过的城市,蚂蚁在构建路径的每一步中,按照随机比例规则选择一个将要到达的城市。其次,在信息素更新的过程中,当所有的蚂蚁构建完路径后,算法会对所有的路径进行全局信息素的更新,此次更新是在全部蚂蚁均完成路径的构造后才进行的,信息素浓度的变化与蚂蚁在这轮中构建的路径长度有关。

1. 路径构建

设蚂蚁当前所在城市为 i ,则其选择城市 j 作为下一个访问对象的概率为:

$$p_k(i, j) = \begin{cases} \frac{[\tau(i, j)]^\alpha [\eta(i, j)]^\beta}{\sum_{u \in J_k(i)} [\tau(i, u)]^\alpha [\eta(i, u)]^\beta}, & j \in J_k(i) \\ 0, & \text{其他} \end{cases} \quad (5-1)$$

其中, k 表示第 k 只蚂蚁, $k \in (1, 2, \dots, m)$, 对于每只蚂蚁 k , 路径记忆向量 \mathbf{R}^k 按照访问顺序记录了所有蚂蚁 k 经过的城市序号; $J_k(i)$ 表示从城市 i 可以直接到达的且又不在蚂蚁访问过的城市序列 \mathbf{R}^k 中的城市集合; $\tau(i, j)$ 表示信息素浓度值; $\eta(i, j)$ 表示启发式信息, 通常由 $\eta(i, j) = \frac{1}{d_{ij}}$ 直接计算得到, 表示蚂蚁从城市 i 到城市 j 的期望程度; d_{ij} 表示从城市 i 到城市 j 之间的距离, 可以看出长度越短、信息素浓度越大的路径被蚂蚁选择的概率越大; α 和 β 是两个预先设置的参数, 用来控制启发式信息和信息素浓度作用的权重关系, 当 $\alpha=0$ 时, 算法演变成传统的随机贪心算法, 最邻近城市被选中的概率最大, 当 $\beta=0$ 时, 蚂蚁完全根据信息素浓度确定路径, 算法将很快收敛, 这样构建出来的最优路径往往与实际目标有较大的差异, 算法性能比较糟糕。

2. 信息素更新规则

蚂蚁释放信息素和各个城市之间链接路径上的信息素挥发是同时进行的, 当蚁群完成一次搜索时, 每条路径上的信息素浓度就要更新一次, 信息素更新公式为:

$$\tau(i, j) = (1 - \rho) * \tau(i, j) + \sum_{k=1}^m \Delta\tau_k(i, j), \Delta\tau_k(i, j) = \begin{cases} (C_k)^{-1}, & (i, j) \in \mathbf{R}^k \\ 0, & \text{其他} \end{cases} \quad (5-2)$$

其中, m 是蚂蚁个数; ρ 是信息素的蒸发率, 规定 $0 < \rho \leq 1$; $\Delta\tau_k(i, j)$ 是第 k 只蚂蚁在经过城市 i 到城市 j 之间的路径上释放的信息素量, 它等于蚂蚁 k 本轮构建路径长度的倒数; C_k 表示路径长度, 是 \mathbf{R}^k 中所有边的长度和。

3. 蚁群算法解决 TSP

应用蚁群算法解决 TSP 的步骤如下。

步骤 1: 设置初始化蚂蚁种群规模 m , 城市节点个数 n , 蚂蚁个体 $k=1$, 当前迭代次数 $N_c=0$, 最大迭代次数 N_{c_max} 以及相应的参数 α, β, ρ 等。

步骤 2: 如果 $N_c < N_{c_max}$ 则转步骤 3, 否则输出最优解。

步骤 3: 如果蚂蚁个体 $k > m$ 转步骤 5, 否则蚂蚁个体 k 依据式(5-1)选择下一个城市节

点 j , 并修改蚂蚁 k 的禁忌表, 将城市节点 j 放入禁忌表中, 利用式 (5-2) 更新局部信息素值, 令蚂蚁 k 禁忌表中城市节点的个数更新为 $\text{num} = \text{num} + 1$ 转步骤 4。

步骤 4: 如果蚂蚁 k 禁忌表中的城市节点的个数 $\text{num} \geq n$, 则令 $k = k + 1$ 转步骤 3, 否则直接转步骤 3。

步骤 5: 所有蚂蚁完成一次遍历后, 令 $N_c = N_c + 1$ 转步骤 1。

5.2.2 离散域和连续域蚁群算法

传统的蚁群算法具有很强的全局搜索能力, 但是也存在一些缺陷, 如搜索时间过长, 收敛速度较慢, 在执行过程中容易出现停滞现象等, 因此为了提高性能, 研究者对传统的蚁群算法进行了改造。

1. 离散域蚁群算法

1992 年, Dorigo 第一次对 ACO 算法进行改进, 提出了精华蚂蚁系统 (Elitist Ant System, EAS)^[20], 在每次迭代之后, 记录历史中的全局最优解, 利用历史信息进行信息素的正反馈来找到最优解。

针对蚁群算法收敛速度慢的缺点, Bullnheimer 等将排序思想扩展到蚁群算法中, 提出了基于排序的蚂蚁系统^[21]。RAS 按照蚂蚁搜索的路径长度以递增的方式进行排序, 并根据此排列顺序对其赋予不同的信息素更新权值。蚂蚁构建的路径越短, 信息素的更新权值就越大, 就可以增加最优解被选择的概率。

针对蚁群算法停滞的缺点, Stutzle 和 Hoos 提出了最大最小蚂蚁系统^[22], 其基本思想就是将每条路径上的信息量限制在 $[\tau_{\min}, \tau_{\max}]$, 若超出这个范围则把信息素分别调整为 τ_{\min} 或者 τ_{\max} , 避免单一路径信息素的积累而使蚂蚁过于集中, 导致算法停滞现象的产生。在每一次循环中, 只对最好路径的信息素进行更新, 可以更好地利用历史信息以加快收敛速度。MMAS 是解决 TSP 等离散优化问题的较好模型。

针对传统蚁群算法容易出现早熟和停滞的现象, 李开荣等改变了传统的 MMAS 算法更新信息量的策略, 提出了一种动态自适应蚁群算法^[23]。其做法就是采用不同的信息素更新公式, 根据解的分布情况动态地调整各条路径上的信息量强度, 提高算法全局搜索能力, 避免早熟和局部收敛的情况。

为了克服蚁群算法计算时间较长的缺点, 吴庆洪等提出了一种具有变异特征的蚁群算法^[24]。在基本的蚁群算法中引入变异机制, 使得该方法具有较快的收敛速度, 节省了计算时间, 通过仿真实验证明了该算法的有效性。

蚁群算法有较强的正反馈能力, 通过信息素的累积能够快速收敛到最优解, 但是如果初始信息素不足, 则求解速度慢。通过将蚁群算法与其他优化算法相结合的方式, 可以弥补其在某些方面的不足。张岩岩等提出了一种改进的人工免疫-蚁群混合算法应用于机器人路径规划问题^[25], 前期利用人工免疫算法自适应能力强、具有记忆功能以及全局收敛性的特点, 找出不同路径环境下的最优参数组合, 后期利用蚁群算法正反馈收敛于最优路径, 大大地提高了算法的运行效率。陈云飞等提出将小生境遗传算法与蚁群算法进行结合^[26], 避免了算法易陷入局部最优的缺点。钟一文等基于免疫机理提出一种免疫蚁群算法来解决有约束关系的多任务调度问题^[27], 该算法能够很好地保持蚁群的多样性。值得一提的是将基本遗传算法与蚁群算法相结合时, 基本遗传算法容易陷入局部最优, 且在初始种群选取数量不

足的情况下,种群的多样性会受到限制。

2. 连续域蚁群算法

在连续域寻优问题的求解中,解空间是以区域性的方式表示的,通过将传统蚁群算法中信息量留存过程拓展为连续空间中信息量分布函数,可实现对连续空间问题的求解。

Bilchev 等首次提出了蚁群优化算法 CACO^[28],将离散域中的一个区域划分为随机分布的区域,通过局部搜索和全局搜索进行解的构建。Dorigo 等提出了 ACOR(Algorithm for Continuous Optimization Problems)算法^[29],该算法利用加权高斯核函数作为信息素概率分布函数,将蚂蚁搜索到的至今最优解作为信息素保存在档案里,并通过更新档案解来更新信息素。Coello 等根据 ACOR 的规则提出了 DACO^[30],它通过一种替代机制改善蚁群多样性以探索更多区域来摆脱局部最优,更适合搜索空间大的连续域优化问题。Xiao J. 提出了一种混合的连续域蚁群优化(Hybrid Ant Colony Optimization for continuous domains, HACO)算法^[31],该算法结合了差分进化和基于种群增长式学习动态的高斯概率密度函数来产生新解。同年,Liao 提出了一种连续优化的局部搜索增量蚁群(Incremental Ant Colony Algorithm with Local Search for continuous optimization, IACOR-LS)算法^[32],该算法使用了三种类型的局部搜索策略,并且随着时间的推移动态调整解档案的大小。Chen 提出一种不同交叉方案的蚁群算法优化(Ant Colony Optimization with Different Crossover Schemes for Continuous Optimization, ACO-MV)^[33],通过将解方案扩展为连续变量、序数变量和分类变量三部分,不仅可以处理连续优化问题,而且还可以处理混合变量优化问题。Yang 提出了一种自适应多模态连续蚁群(Adaptive Multimodal continuous Ant Colony Optimization, AM-ACO)算法^[34],在更新过程中分别采用自适应参数调整策略、差分进化算子和局部搜索的方式,在求解多峰连续优化问题的过程中,能够很好地平衡局部最优和全局最优。

5.3 基于蚁群算法的路径规划

5.3.1 蚁群算法的路径规划中的优势

随着移动机器人的发展,路径规划的研究受到越来越多专家学者的关注。路径规划的目标就是在有障碍物的环境中按照某些性能指标,寻找一条从起点到终点的最优或者近似最优的无碰撞路径。在路径规划问题中,性能指标需要考虑能量消耗、时间和距离等相关因素,并且根据这些因素制定最佳移动准则。根据环境信息的已知程度,移动机器人路径规划的方法可以分为两种类型:基于模型的全局路径规划和基于传感器的局部路径规划。

全局路径规划是指掌握全部环境信息的前提下,对机器人的路径进行规划,又称为静态路径规划。常用的适用于全局路径规划的算法有启发式搜索方法以及蚁群算法、粒子群优化算法、遗传算法和模拟退火算法等各种智能算法。与全局路径规划方法不同,局部路径规划方法对环境中的障碍物的信息是未知的,需要通过传感器实时感知周围环境信息与自身状态,规划出一条从起点到目标点的安全运动路径,又称为动态路径规划。局部规划更具实时性和实用性,对动态环境有较强的适应能力。由于机器人在路径规划中仅仅依靠局部信息,容易产生局部极值点或者出现振荡的情况,造成机器人无法顺利达到目的地的情况。常用的适用于局部路径规划的方法有事例学习法、滚动窗口法、人工势场法以及行为分解法。

蚁群算法作为一种随机搜索的全局优化算法,具有强鲁棒性、隐含并行性、易与其他算法相结合等优点,可以用来解决移动机器人的全局路径规划问题。前面提到蚁群算法的生物基础就是蚁群在蚁巢与食物之间寻找到一条最短的路径,与移动机器人的路径规划的过程相似,二者在内部的机理上存在着天然的联系,为蚁群算法求解路径规划问题提供了可靠的依据。但是蚁群算法本身不可避免地带有搜索速度慢、易陷入局部最优的缺点。

5.3.2 算法描述以及实现

全局路径规划依据完全已知环境中障碍物的位置和形状给移动机器人规划出一条从起点至终点的安全运动路径^[35],规划路径的精确程度取决于获取的环境信息的质量。全局规划路径方法通常计算量大,实时性较差。当环境中出现移动的障碍物时,此方法的效果可能不尽如人意。全局路径规划涉及两部分,包括环境模型的建立和路径搜索策略。

现有的构建环境模型方法主要有栅格分解法、单元树法、可视图法和 Voronoi 法。栅格分解法是研究路径规划问题时常用的建模方法,它能够对复杂环境进行单元分割,将环境空间划分成若干个相同大小的栅格,并用栅格数组表示环境。每个栅格点可能是自由或者被占据,对于混合栅格点(即一部分是自由空间另一部分是障碍物),依据其在自由空间或障碍物空间的比例,将其归属于自由或者被占据。栅格之间相互邻接,这样路径规划问题简化为路径所包含的栅格之间的连接。

首先基于基本蚁群算法的路径规划模拟蚂蚁蚁群的觅食行为,机器人的出发点为蚁巢的位置,最终目标地点为食物源的位置,蚂蚁觅食的过程就是从蚁巢出发,在搜索空间内寻找食物源的过程,蚁群在觅食过程中,通过信息素的正反馈作用找到一条最短的安全路径。算法的实现过程如下所述。

步骤 1: 利用栅格法建立环境模型,以矩阵形式描述承载环境信息的栅格信息,以单元数组的形式描述栅格的连通信息。

步骤 2: 初始化蚁群算法搜索路径的参数,设置信息素的初始分布,将蚂蚁置于初始位置(即初始栅格)。

步骤 3: 提取蚂蚁所在栅格的连通信息,计算蚂蚁移动到与其所在栅格连通栅格的转移概率,并根据概率移动蚂蚁,局部更新两栅格之间的信息素。

步骤 4: 判断蚂蚁是否移动到目标栅格,如果没有则转到步骤 3,否则进入步骤 5。

步骤 5: 比较判断选出本次迭代中的当前最优路径,对较优的路径进行信息素的全局更新。判断是否求得最优路径或满足终止条件,如果没有则返回步骤 2; 否则程序结束,进入步骤 6。

步骤 6: 对最优路径进行可行性处理,输出最优路径。

5.3.3 全局路径规划方法

针对蚁群算法本身存在的缺点,在解决路径规划问题时从算法本身机制以及算法与其他算法结合的角度,对算法进行合理的改进。朱庆保提出一种滚动规划蚂蚁算法^[36],由两组蚂蚁采用最邻近搜索策略相互协作完成机器人每一步局部最优路径的搜索,不断动态修改前进路径从而找到一条全局优化的路径。Yang J. 等提出一种改进的蚁群优化算法^[37],通过引入遗传算子以及对全局更新规则的修改,使算法成功跳出局部最小值并提高了算法

的收敛速度。赵娟平等提出了一种基于参数模糊自适应窗口蚁群优化算法^[38],首先利用模糊控制优化参数 α 、 β 和 ρ ,并引入城市节点活跃度的概念将其作为未来信息,指导蚂蚁进行解的构造和信息素的更新,仿真结果证明,即便在复杂环境中该算法仍然能够快速规划路径。他们还针对蚁群算法易陷入局部最优的缺点,提出了一种复杂静态环境下移动机器人路径规划的改进蚁群优化算法——差分演化混沌蚁群算法^[39]。该算法利用差分演化算法进行信息素的更新,同时针对可能出现的停滞现象,在信息素更新时加入了混沌扰动因子,算法还采用了一个新的评价函数增强了算法的逃逸能力,避免了路径死锁现象,也提高了最优路径的搜索效率,仿真结果表明,即使在障碍物非常复杂的环境,算法仍能快速规划出安全的优化路径。左大利等提出一种改进的蚁群算法^[40],以栅格法建立机器人工作环境,改进信息素的更新方式,设置信息素浓度的阈值,引入死锁处理策略,改进状态转移概率,增加解的多样性。值得注意的是以上方法均使用栅格法对环境进行建模,当地图中空白栅格数量较多时,需要使用蚁群算法频繁地对路径进行更新,因此获得最优路径的过程耗时较长。

5.4 基于蚁群算法的社区检测

5.4.1 多目标蚁群算法

多目标优化问题通常描述为:

$$\begin{cases} \min F(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})] \\ \text{subject to } \mathbf{x} = (x_1, x_2, \dots, x_m) \in \Omega \end{cases} \quad (5-3)$$

其中, Ω 表示的是决策空间的可行域, \mathbf{x} 为决策向量, $k \geq 2$ 表示目标函数的个数。 $F: \Omega \rightarrow \mathcal{R}^m$ 表示从决策向量空间到 k 个目标函数空间的映射。

下面对多目标优化问题的最优解进行定义。

定义 5-1(x_1 优于 x_2) 一个解 x_1 比另一个解 x_2 占优(记作 $x_1 > x_2$)当且仅当

$$f_i(x_1) \leq f_i(x_2) (\forall i = 1, 2, \dots, k), \quad f_i(x_1) < f_i(x_2) (\exists i = 1, 2, \dots, k) \quad (5-4)$$

定义 5-2(Pareto 最优解) $x^* \in X$ 是一个 Pareto 最优解或非占优解的定义是不存在另一个 x 会使得 x 比 x^* 更占优。

定义 5-3(Pareto 最优解集) 一个多目标优化问题会产生很多 Pareto 解,而所有的 Pareto 解构成 Pareto 最优解集合。

$$p^* = \{x^* \in X \mid \neg \exists x \in X, x > x^*\} \quad (5-5)$$

定义 5-4(Pareto 最优前沿) 把非占优解映射到目标空间得到的目标函数值形成的区域称为 Pareto 最优前沿(Pareto Optimal Front, POF)。

$$\text{POF} = \{F(x^*) = (f_1(x^*), f_2(x^*), \dots, f_k(x^*))^T \mid x^* \in p^*\} \quad (5-6)$$

对于一个网络 $G = (V, E)$,其中 V 表示节点的集合, E 表示节点之间连接的边的集合。使用邻接矩阵 A 来存储图中节点与节点之间的连接,如果节点 v_i 和 v_j 之间存在一条边,那么 $A_{ij} = 1$,否则 $A_{ij} = 0$ 。如果 V_1 和 V_2 是 V 的两个互不相交的子集,定义 $L(V_1, V_2) = \sum_{i \in V_1, j \in V_2} A_{ij}$ 和 $L(V_1, \bar{V}_2) = \sum_{i \in V_1, j \in \bar{V}_2} A_{ij}$,其中 $\bar{V}_2 = V - V_2$ 。给定一个图 G 的划分 $S = (V_1, V_2, \dots, V_s)$,其中 V_i 是图 G 的子图 G_i 对应的节点集 $i = 1, 2, \dots, s$ 。模块度密度 D 的定义为

$$D = \sum_{i=1}^s \frac{L(V_i, V_i) - L(V_i, \bar{V}_i)}{|V_i|} \quad (5-7)$$

式(5-7)中,每一个求和项的分子 $L(V_i, V_i) - L(V_i, \bar{V}_i)$ 表示的是子图节点的内度之和减去其节点的外度之和,分母 $|V_i|$ 表示该子图的节点数目。社区检测问题就可以看作是找到一个划分,使得模块度密度 D 的值最大的优化问题。

基于多目标蚁群算法的社区检测中涉及 NRA 和 RC 两个函数:

$$\text{NRA} = \sum_{i=1}^s \frac{L(V_i, V_i)}{|V_i|} \quad (5-8)$$

$$\text{RC} = \sum_{i=1}^s \frac{L(V_i, \bar{V}_i)}{|V_i|} \quad (5-9)$$

NRA 反映的是在各个社区内节点彼此之间互相连接的边平均值求和,如果 NRA 越小,则将网络划分成为少量具有高内部连接密度的社团结构。RC 表示的是各个社区内节点和其他社区之间连边的平均值求和,最小化 RC 会导致网络中社团数目较少,每个社团的节点数增多,将一个网络划分成和图中其他部分连接较稀疏的大社团。

模块度密度 $D = \sum_{i=1}^s \frac{L(V_i, V_i) - L(V_i, \bar{V}_i)}{|V_i|} = -\left(-\sum_{i=1}^s \frac{L(V_i, V_i)}{|V_i|} + \sum_{i=1}^s \frac{L(V_i, \bar{V}_i)}{|V_i|}\right) = -\text{NRA} - \text{RC}$,所以要最大化模块度密度的值就是最小化 NRA 和 RC,目标函数可以转化为下面的多目标问题:

$$\min \begin{cases} f_1 = \text{NRA} = \sum_{i=1}^s \frac{L(V_i, V_i)}{|V_i|} \\ f_2 = \text{RC} = \sum_{i=1}^s \frac{L(V_i, \bar{V}_i)}{|V_i|} \end{cases} \quad (5-10)$$

蚁群算法作为一种基于种群的算法,是受蚂蚁觅食行为的启发而设计的一种仿生算法。可以将多目标算法和蚁群算法结合起来解决复杂网络的社区检测问题,采用分解机制可以将多目标问题转化为一系列单目标的子问题,每一只蚂蚁得到的解对应着 Pareto 前端特定的点。另外在算法设计过程中应考虑蚂蚁与子问题之间的关系、解的编码方式和构造、启发式信息矩阵的定义、信息素矩阵的定义和更新方式等问题。

5.4.2 社区检测问题的改进

社区结构的概念最早是由 Girvan 和 Newman 提出的,一个复杂网络可以划分成若干个社区,社区内部的节点连接密度高于社区间节点的连接密度^[41]。社区检测的算法大致可以分为基于划分的算法、基于模块度函数优化算法、基于标签传播的算法以及各种仿生计算算法等。由于蚁群算法采用分布式正反馈并行计算机制,有较强的稳定性和鲁棒性。2007年,Liu 等首次在发掘社区结构领域^[42]应用蚁群算法,随后其得到广泛应用,但是蚁群算法本身存在收敛速度慢、容易陷入局部最优解等缺点,影响了其应用效果。

针对以上问题,研究者在蚁群算法的基础上提出了许多改进算法。He 等提出将蚁群算法与模拟退火及标签传播算法结合形成 MABA 算法^[43],它以 SABA 作为子方法,通过优化局部模块度函数 Q ,在已获得的网络社区结构上构建一个上层网络,再将 SABA 作用于新的上层网络,如此迭代直到 Q 值不再增加。MABA 虽然可以缓解模块度优化带来的

分辨率限制问题,但是算法求解精度较低。Chang 等提出在蚁群算法的基础上结合遗传算法中染色体解的表达形式(MACO 算法)^[44],该算法中蚂蚁通过跳跃选择下一个要到达的点编号,以此来填充解向量。MACO 算法对社区有很好的划分结果,但是染色体解的解码步骤耗时严重。Mu 等对 MACO 算法进行改进并提出 IACO 算法^[45],采用基于学习的策略尽可能减少探索阶段的冗余计算,但是解码耗时问题仍然没有解决。顾军华等提出一种基于标签传播的蚁群优化算法 (BLPACO)^[46],该算法采用一种新的解向量表达方式,其中每个节点位置存放该节点所属社区的标签,在解的构造阶段提出基于节点凝聚性的蚂蚁转移策略,降低蚂蚁转移过程中的随机性,并将标签传播思想引入蚁群搜索过程,使算法快速收敛以及提高算法的精确度。现有的社区检测算法也只解决了现实系统中的简单问题,对复杂网络的社区检测问题的研究方兴未艾,许多具有研究前景的问题等待着研究者投入精力研究。

参考文献

- [1] Colorni A, Dorigo M, Maniezzo V. Distributed optimization by antcolonies[C]//Proceedings of the first European conference on artificial life, 1991, 142: 134-142.
- [2] Dorigo M, Maniezzo V, Colorni A. Ant system: optimization by a colony of cooperating agents[J]. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 1996, 26(1): 29-41.
- [3] Gambardella L M, Dorigo M. Ant-Q: A Reinforcement Learning Approach to the Traveling Salesman Problem[C]. Proceedings of the 12th International Conference on Machine Learning, 1995: 252-260.
- [4] Dorigo M, Gambardella L M. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem[J]. IEEE Transactions on Evolutionary Computation, 1997, 1(1): 53-56.
- [5] Stützle T, Hoos H H. Max-min ant system[J]. Future Generation Computer Systems, 2000, 16(8): 889-914.
- [6] Bullnheimer B, Hartl R F, and Strauss C. A new rank-based version of the Ant System: A computational study[J]. Central European Journal for Operations Research and Economics, 1999, 7(1): 25-38.
- [7] Gambardella L M, Dorigo M. An Hybrid Ant System for the Sequential Ordering Problem. Technical Report, IDSIA, Lugano, CH, 1997: 11-97.
- [8] Li K, Xu F, Huang P, et al. A New Best-Worst Ant System with Heuristic Crossover Operator for Solving TSP. [C]//2009 Fifth International Conference on Natural Computation. IEEE Computer Society, 2009: 92-97.
- [9] 吴庆洪, 张纪会, 徐心和. 具有变异特征的蚁群算法. 计算机研究与发展. 1999, 36(10): 1240-1245.
- [10] 陈焯. 带杂交算子的蚁群算法[J]. 计算机工程, 2001, 27(12): 74-76.
- [11] 陈峻, 沈洁, 秦玲, 等. 基于分布均匀度的自适应蚁群算法. 软件学报, 2003, 14(08): 1379-1387.
- [12] 黄国锐, 曹先彬, 王煦法. 基于信息素扩散的蚁群算法[J]. 电子学报, 2004, 32(5): 865-868.
- [13] Qin H, Zhou S, Huo L, et al. A New Ant Colony Algorithm Based on Dynamic Local Search for TSP [C]//Fifth International Conference on Communication Systems and Network Technologies. IEEE, 2015: 913-917.
- [14] 李德华, 王韶, 刘洋, 等. 模糊遗传算法和蚁群算法相结合的配电网重构[J]. 电力系统保护与控制, 2009, 37(17): 26-31.
- [15] 邓高峰, 张雪萍, 刘彦萍. 一种障碍环境下机器人路径规划的蚁群粒子群算法[J]. 控制理论与应用, 2009, 26(5): 579-553.

- [16] Shahla N, Mohammad E B, Nasser G A, et al. A novel ACO-GA hybrid algorithm for feature selection in Protein function Prediction [J]. *Expert System with Applications*, 2009, 36 (10): 12086-12094.
- [17] Lee Z J, Su S F, Chuang C C, et al. Genetic algorithm with ant colony optimization (GA-ACO) for multiple sequence alignment [J]. *Applied Soft Computing*, 2008, 8(1): 55-78.
- [18] Dorigo M, Maniezzo V, Colomi A. Ant system: optimization by a colony of cooperating agents [J]. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 1996, 26(1): 29-41.
- [19] Dorigo M, DiCaro G, Gambardella L M. Ant algorithms for discrete optimization [R]. Technical Report IRIDIA/98-10. *Artificial Life*, 1999, 5(2): 137-172.
- [20] Dorigo M. Optimization, Learning and Natural Algorithms [D]. Milano: Politecnico di Milano, 1992.
- [21] Bullnheimer B, Hartl R F, Strauss C. A new rank-based version of the Ant System: A computational study [J]. *Central European Journal for Operations Research and Economics*, 1999, 7(1): 25-38.
- [22] Stutzle T, Hoos H. Improvements on the ant system: Introducing MAX-MIN ant system [C]// *Proceeding of the Int Confon Artifieial Neural Networks and Genetie Algorithms*. Wien: Springer-Verbag, 1997: 245-249.
- [23] 李开荣, 陈宏建, 陈峻. 一种动态自适应蚁群算法 [J]. *计算机工程与应用*, 2004, 29(2): 152-155.
- [24] 吴庆洪, 张纪会, 徐心和. 具有变异特征的蚁群算法 [J]. *计算机研究与发展*, 1999, 36 (10): 1240-1245.
- [25] 张岩岩, 侯媛彬, 李晨. 基于人工免疫改进的搬运机器人蚁群路径规划 [J]. *计算机测量与控制*, 2015-23(12): 4124-4127.
- [26] 陈云飞, 刘玉树, 范洁, 等. 广义分配问题的一种小生境遗传蚁群优化算法 [J]. *北京理工大学学报*, 2005, 25(6): 490-494.
- [27] 钟一文, 杨建刚. 求解多任务调度问题的免疫蚁群算法 [J]. *模式识别与人工智能*, 2006, 19(1): 73-78.
- [28] Bilchev G, Parmee I C. The ant colony metaphor for searching continuous design spaces [C]// *AISB Workshop on Evolutionary Computing*. Springer Berlin Heidelberg, 1995: 25-39.
- [29] Socha K, Dorigo M. Ant colony optimization for continuous domains [J]. *European Journal of Operational Research*, 2008, 185(3): 1155-1173.
- [30] Coello C A C. An alternative ACOR, algorithm for continuous optimization problems [C]// *International Conference on Swarm Intelligence*. Springer-Verlag, 2010: 48-59.
- [31] Xiao J, Li L P. A hybrid ant colony optimization for continuous domains [J]. *Expert Systems with Applications*, 2011, 38(9): 11072-11077.
- [32] Liao T, Aydin D, et al. An incremental ant colony algorithm with local search for continuous optimization [C]// *Proceedings of Genetic and Evolutionary Computation Conference (GECCO)*, Dublin, Ireland, 2011: 125-132.
- [33] Chen Z, Jiang Y, Wang R. Ant Colony Optimization with Different Crossover Schemes for Continuous Optimization [C]// *Bio-Inspired Computing—Theories and Applications*. Springer, Berlin, Heidelberg, 2015: 56-62.
- [34] Yang Q, Chen W N, Yu Z, et al. Adaptive Multimodal Continuous Ant Colony Optimization [J]. *IEEE Transactions on Evolutionary Computation*, 2017, 21(2): 191-205.
- [35] 孟偲, 王田苗. 一种移动机器人全局最优路径规划算法 [J]. *机器人*, 2008, 30(3): 217-222.
- [36] 朱庆保. 复杂环境下的机器人路径规划蚁群算法 [J]. *自动化学报*, 2006, 32(4): 586-593.
- [37] Yang J, Zhuang Y. An improved ant colony optimization algorithm for solving a complex combinatorial optimization problem [J]. *Applied Soft Computing*, 2010, 10(2): 653-660.
- [38] 赵娟平, 高宪文, 刘金刚, 等. 移动机器人路径规划的参数模糊自适应窗口蚁群优化算法 [J]. *控制与*

- 决策,2011,26(7): 1096-1100.
- [39] 赵娟平,高宪文,符秀辉. 移动机器人路径规划的改进蚁群优化算法[J]. 控制理论与应用,2011,28(4): 457-461.
- [40] 左大利,聂清彬,张莉萍,等. 移动机器人路径规划中的蚁群优化算法研究[J]. 现代制造工程,2017(05): 44-48.
- [41] Girvan M, Newman M E. Community structure in social and biological networks[J]. Proceeding of National Academy of Science,2002,9(12): 7821-7826.
- [42] Liu Y,Wang Q X, Wang Q, et al. Email Community Detection Using Artificial Ant Colony Clustering[M]//Advances in Web and Network Technologies, and Information Management. Springer Berlin Heidelberg,2007: 287-298.
- [43] He D,Liu J,Liu D,et al. Ant colony optimization for community detection in large-scale complex networks[C]//7th International Conference on Natural Computation, IEEE,2011.
- [44] Chang H,Feng Z,Ren Z. Community detection using Ant Colony Optimization[C]//IEEE Congress on Evolutionary Computation,2013.
- [45] Zhou X,Liu Y,Zhang J,et al. An ant colony based algorithm for overlapping community detection in complex networks[J]. Physica A: Statistical Mechanics and its Applications,2015,427: 289-301.
- [46] 顾军华,江帆,武君艳,等. 基于标签传播的蚁群优化算法求解社区发现问题[J]. 计算机应用与软件,2019,36(06): 233-242.