

蚁群算法在求解旅行商问题中的应用综述^①

郭城成¹, 田立勤², 武文星³

¹(青海师范大学 计算机学院, 西宁 810016)

²(华北科技学院 计算机学院, 廊坊 065201)

³(华北科技学院 安全工程学院, 廊坊 065201)

通信作者: 郭城成, E-mail: 943628840@qq.com



摘要: 旅行商问题作为组合优化研究中最具挑战的问题之一, 自被提出以来就引起了学术界的广泛关注并提出了大量的方法来解决它. 蚁群算法是求解复杂组合优化问题的一种启发式仿生进化算法, 是求解旅行商问题的有效手段. 本文分别介绍蚁群算法中几个有代表性的算法, 综述了蚁群算法的改进、融合和应用的文献研究进展, 以评价近年来不同版本的蚁群算法为解决旅行商问题的发展和研究成果, 并针对改进蚁群算法结构框架、算法参数的设置及优化、信息素优化和混合算法等方面, 对现被提出的改进算法进行了分类综述. 对蚁群算法在未来对旅行商问题及其他不同领域的研究内容和研究热点的进一步发展提供了展望和依据.

关键词: 蚁群算法; 旅行商问题; 启发式算法; 信息素优化

引用格式: 郭城成, 田立勤, 武文星. 蚁群算法在求解旅行商问题中的应用综述. 计算机系统应用, 2023, 32(3): 1-14. <http://www.c-s-a.org.cn/1003-3254/8976.html>

Survey on Ant Colony Optimization for Solving Traveling Salesman Problem

GUO Cheng-Cheng¹, TIAN Li-Qin², WU Wen-Xing³

¹(School of Computer Science, Qinghai Normal University, Xining 810016, China)

²(School of Computer Science, North China Institute of Science and Technology, Langfang 065201, China)

³(School of Safety Engineering, North China Institute of Science and Technology, Langfang 065201, China)

Abstract: As one of the most challenging problems in combinatorial optimization, the traveling salesman problem has attracted extensive attention from the academic community since its birth, and a large number of methods have been proposed to solve it. The ant colony optimization (ACO) is a heuristic bionic evolutionary algorithm for solving complex combinatorial optimization problems, which is effective in solving the traveling salesman problem. This study introduces several representative ACOs and makes a literature review of the improvement, fusion, and application progress of ACOs to evaluate the development and research achievements of different versions of ACOs in solving the traveling salesman problem in recent years. Moreover, the improved ACOs are summarized in categories in terms of the framework structure, setting and optimization of algorithm parameters, pheromone optimization, and hybrid algorithms. The research provides an outlook and basis for the ACO application to solve the traveling salesman problem and further develop the research content and focuses of other fields.

Key words: ant colony optimization (ACO); traveling salesman problem; heuristic algorithm; pheromone optimization

① 基金项目: 河北省物联网监控技术创新中心 (21567693H); 青海省物联网重点实验室 (2017-ZJ-Y21); 中央高校基本科研业务费 (3142021009)

收稿时间: 2022-07-29; 修改时间: 2022-09-01; 采用时间: 2022-09-19; csa 在线出版时间: 2022-12-02

CNKI 网络首发时间: 2022-12-05

1 引言

旅行商问题 (traveling salesman problem, TSP)^[1] 是程序研究领域中以寻找通过一组城市的最短路径为目标的一个标准组合优化的问题, 是离散优化算法性能分析中常用的标准测试问题之一. 这意味着 TSP 属于一类 NP (non-deterministic polynomial) 问题, 即非确定性多项式时间困难问题. NP 完备问题的计算复杂度随着城市数量的增长而呈指数级增长. 因此, 近年来提出了一些启发式算法来解决这一问题, 在计算量和时间复杂度方面取得了更好的效果. 所以 TSP 问题的研究大多集中在启发式算法上. 根据文献提供的解决方案, 目前还没有有效的算法来解决 TSP 问题, 虽然 TSP 中对这个问题的表述很简单, 但它仍然是对计算机研究人员来说最具挑战性的难题之一, 因为很难找到最短的封闭行程来连接所有给定的城市. TSP 问题已被应用于许多的实际应用中, 如数据关联、车辆路径、图像处理 and 模式识别等.

群智能算法 (swarm intelligence, SI)^[2] 就是启发式算法, 其适用于用或多或少的性能来解决 NP 难题, 为处理大规模、非线性和复杂的问题在相对可接受的时间内返回了一个具有竞争力的解决方案. 然而, 启发式算法并不能保证问题的最优解, 只能保证次优解. 但在许多情况下, 仍可以使用启发式方法在可接受的时间内生成一个高质量的解决方案. 现代启发式算法采用一种“邻域搜索”结构, 在优化机制方面存在一定的差异, 但在优化流程上基本一致. 其步骤如下.

步骤 1. 算法构建一个或一组初始解.

步骤 2. 在关键参数控制下通过邻域函数产生若干邻域解.

步骤 3. 按照接受准则 (确定性、概率性或混沌方式) 更新当前状态.

步骤 4. 按关键参数修改准则来调整关键参数的值.

如此重复上述步骤直到满足算法的收敛准则, 最终得到问题的最优解. 启发式算法的优化流程如图 1 所示.

许多已知的方法已被应用于解决 TSP 问题, 如线性规划法、回溯算法、分支限界方法等精确算法. 精确的方法并不总是适用于较大的实例, 因此, 近几年对 TSP 的研究主要集中在使用先进的启发式和元启发式算法, 如遗传算法 (genetic algorithm, GA)^[3]、果蝇优化算法 (fruit fly optimization algorithm, FFOA)^[4]、黑洞算法 (black hole algorithm, BHA)^[5]、禁忌搜索算法 (tabu

search, TS)^[6] 和粒子群算法 (particle swarm optimization, PSO)^[7] 等. 其中, 许多是新提出的元启发式算法, 它们大多结构简单, 易于应用, 但是理论基础缺乏, 还没有得到大量实验验证. 蚁群算法 (ant colony optimization, ACO) 作为一种典型的启发式算法, 最初是于 20 世纪 90 年代由意大利学者 Dorigo 等人^[8] 首次提出并不断改进的一种应用于求解复杂组合优化问题的元启发式仿生进化算法. 它从一些蚂蚁物种的觅食行为中获取灵感, 受到群体行为的启发. 它是由许多个体组成的, 这些个体都是同质的, 其中所有的局部通信都基于简单的规则和良好的自组织机制. 它的自催化行为形成了一种有机的增强型学习系统, 具有很强的鲁棒性, 并且采用优良的分布式并行计算特性, 易于与其他优化算法相融合, 但是正反馈机制易于使算法过早的陷入全局最优解和收敛速度慢的问题, 也存在种群多样性和时间性能上的矛盾, 传统的蚁群算法仍不够完美.

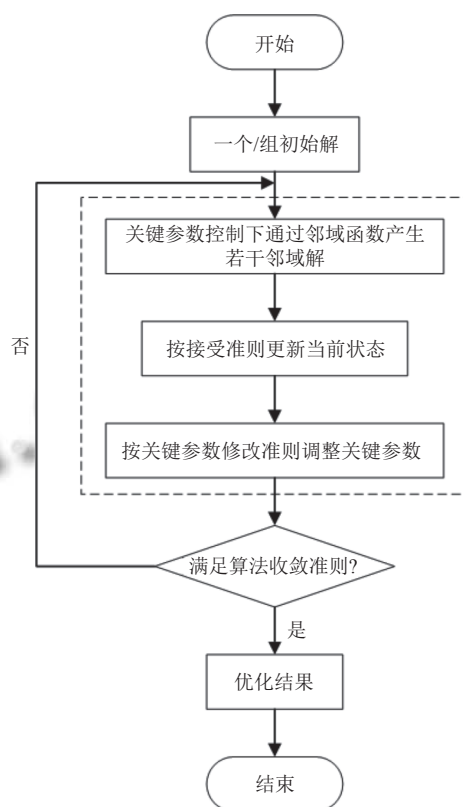


图 1 群智能算法优化流程图

蚁群算法自被提出以来就被成功应用, 许多研究者利用蚁群算法作为基准来寻找更好的解决方案. 本文概括地介绍了蚁群算法从诞生到成熟过程中几个代表性的算法, 给出了 TSP 问题的数学模型和蚁群算法

的数学背景;对解决 TSP 问题而提出的算法优化方案和一些最优化算法的实现方法进行了综述.以评价不同版本的蚁群算法在 TSP 问题领域中的应用.

2 旅行商问题数学模型

在优化问题中,我们的目标是遵守特定的约束来寻找最接近最优解的结论.TSP 问题是组合优化问题中最经典的问题之一,它可表述为在 n 个城市的集合中,一个商人从某一个城市出发,约束条件是只经过其他所有城市一次,最后再回到起始城市,也就是说它必须以最低的成本一次且仅一次的通过所有城市.一般来说,旅行长度代表解决方案的成本,路由目的是最终找到最小闭合路径 $R = [C_1, C_2, \dots, C_n]$,称之为最短路径.已知每条边对应的城市节点对 (i, j) 之间的距离为 d_{ij} ($i, j \in 1, 2, \dots, n$),若路径 R 满足下列目标函数 $f(R)$ 的最小值,则路径 R 就是所求的最优路径:

$$f(R) = \sum_{i=1}^{n-1} d_{(C_i, C_{i+1})} + d_{(C_1, C_n)} \quad (1)$$

按照图论方法对其解释, TSP 问题也意味着在赋权完全图中找到最小权重的哈密顿 (Hamilton) 回路.解决方法是将图中的每个节点表示为一个城市,每条路径都有一个对应于城市之间的距离的权重,优化目标是使总距离最小化(按距离计算达到最佳路径).搜索空间是 n 个点的所有排列的集合,大小为 $(n-1)!$ 记 $G=(V, E)$ 表示赋权完全图, V 为 n 个城市构成的顶点集, E 表示节点间边的集合,已知各点间的距离 $d_{ij} > 0$ ($i, j \in V, i \neq j$),设 x_{ij} 为问题的决策变量,其表达式如下:

$$x_{ij} = \begin{cases} 1, & \text{若}(i, j)\text{在最优路径上} \\ 0, & \text{其他} \end{cases} \quad (2)$$

当 x_{ij} 等于 1 时,就可以实现从节点 i 到节点 j 的路径;否则,在所考虑的行程中,节点之间没有直接连接.则该问题优化的目标函数为:

$$\min \sum_{i=1}^n \sum_{j=1}^n d_{ij} x_{ij} \quad (3)$$

3 蚁群算法求解 TSP 问题的工作原理

ACO 模仿了真实蚂蚁的觅食行为,利用人工蚂蚁通过信息素作为分布式传播的媒介.它的灵感来自于 Goss 等人使用真实蚂蚁进行的双桥实验^[9].当蚂蚁觅食时,在穿越路径上留下一一种被称为信息素的挥发性

化学物质,这种信息素可以在一定范围内被其他蚂蚁感知到,并影响他们的行为.同时信息素的浓度随着时间的推移是波动的,那些碰巧选择了最短的食物路线的蚂蚁将会提前返回巢穴,并通过在返回巢穴的路上放置食物追踪信息素来强化这条最短的路线.路径上高浓度的信息素会逐渐吸引其他蚂蚁跟随,随着越来越多的蚂蚁跟随这条路线,它的信息素不断得到强化,使其对其他蚂蚁的吸引力更大.这是一个自催化的或正反馈的过程.

3.1 基本蚁群算法-蚂蚁系统求解 TSP 问题

蚂蚁系统 (ant system, AS) 作为最初 Dorigo 等人^[8]的 ACO 算法,使用探索和开发阶段为解决 TSP 找到最短路径而被提出,为其他蚁群算法提供了基本框架.通常各种蚁群算法都是以 AS 作为他们的先驱和研究的基础,在其基础上提出了不同的改进和扩展.其基本结构如图 2 所示.

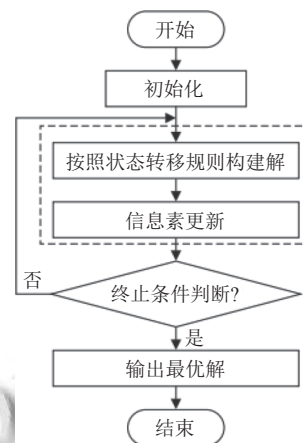


图 2 蚂蚁系统的基本结构

现从蚁周模型的角度说明基本蚁群算法求解 TSP 问题的具体步骤.

步骤 1. 初始化.将等于或小于 n 个城市数量的 m 个蚂蚁随机置于与城市相对应的顶点上作为起始位置.将各城市路径间的信息素初始化为一个常数 $\tau_{ij}(0)$.对每只蚂蚁 k ($k = 1, 2, \dots, m$),将一个记录其已走过城市的禁忌表 $tabu_k$ ($k = 1, 2, \dots, m$) 初始化长度为 1,即 $tabu_k$ 初始化状态中只包含蚂蚁的起始城市.

步骤 2. 路径构建.在每一次迭代循环中,每只蚂蚁 k 根据路径上的信息素和启发式信息(两个节点位置之间的距离)独立选择其需要访问的下一个城市.为避免多次访问同一个城市,将蚂蚁 k 当前所访问过的城市用禁忌表 $tabu_k$ 记录为禁忌城市.每只蚂蚁 k 在搜

索路径过程中,根据路径上的信息量采用轮盘赌原则来计算 t 时刻从当前城市 i 转移到下一个尚未访问的城市 j 的状态转移概率 $p_{ij}^k(t)$,如式(4)所示,并将城市 j 置于禁忌表 $tabu_k$ 中,直到所有蚂蚁都完成一次循环:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{s \in allowed_k} [\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}, & j \in allowed_k \\ 0, & \text{其他} \end{cases} \quad (4)$$

其中, $\tau_{ij}(t)$ 表示 t 时刻城市 i 和城市 j 之间即路径 (i, j) 上的信息素强度;为了保证解的合法性, $allowed_k = \{1, 2, \dots, n\} - tabu_k$ 表示蚂蚁 k 没有被访问过的下一步可选择的城市的集合。

η_{ij} 为启发式函数,表示 t 时刻路径 (i, j) 之间的能见度,与城市 i 和城市 j 之间的距离成反比,即路径距离越大,能见度越小.其取值为 $\eta_{ij} = 1/d_{ij}$, $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ 表示城市 i, j 之间路径的长度。

α 为信息素影响因子,反映了蚂蚁在选择路径时积累的信息素的影响程度. α 越大,蚂蚁在选择下一路径时,信息素的影响越大,即蚂蚁越倾向于选择以前走过的路径。

β 为启发式函数影响因子,反映了蚂蚁在运动过程中启发信息(两个节点位置之间的距离)在蚂蚁选择路径中的影响程度. β 越大,在计算下一路径的状态转移概率时,启发式的作用就越大,即尽管路径上信息素量很大,但蚂蚁 k 总会以很高的概率选择更近的城市。

步骤3. 更新信息素.为了防止剩余信息素过多而引起的残留信息淹没启发因子,当 m 个蚂蚁完成对所有 n 个城市的一次遍历都找到了一条合法路径后,对各个路径上残留信息素根据式(5)中的全局更新规则进行更新处理:

$$\begin{cases} \tau_{ij}(t+n) = (1-\rho) \times \tau_{ij}(t) + \Delta \tau_{ij}(t) \\ \Delta \tau_{ij}(t) = \sum_{k=1}^m \Delta \tau_{ij}^k(t) \end{cases} \quad (5)$$

其中, ρ 表示信息素挥发因子,取值范围通常为 $(0, 1)$; $1-\rho$ 表示信息素残留因子. $\Delta \tau_{ij}(t)$ 表示本次迭代中所有蚂蚁留在路径 (i, j) 上的信息素增量; $\Delta \tau_{ij}^k(t)$ 表示第 k 只蚂蚁在本次迭代中留在路径 (i, j) 上的信息素浓度,其值主要根据信息素的更新方式和释放量不同分为3种模型,分别是蚁周模型(ACS)、蚁密模型(AQS)、蚁量模型(ADS),如下:

ACS 模型:

$$\Delta \tau_{ij}^k(t) = \begin{cases} Q/L_k, & \text{当蚂蚁 } k \text{ 在本次循环中经过 } (i, j) \\ 0, & \text{其他} \end{cases} \quad (6)$$

AQS 模型:

$$\Delta \tau_{ij}^k(t) = \begin{cases} Q/d_{ij}, & \text{当蚂蚁 } k \text{ 在本次循环中经过 } (i, j) \\ 0, & \text{其他} \end{cases} \quad (7)$$

ADS 模型:

$$\Delta \tau_{ij}^k(t) = \begin{cases} Q, & \text{当蚂蚁 } k \text{ 在本次循环中经过 } (i, j) \\ 0, & \text{其他} \end{cases} \quad (8)$$

其中, Q 是一个常量,表示每只蚂蚁完成遍历后的总信息素强度, L_k 表示蚂蚁 k 完成本次遍历后所走路径的总长度。

AQS 模型和 ADS 模型采用局部更新的方式在每只蚂蚁遍历节点 (i, j) 之间的路径过程中对信息素进行更新. ACS 模型采用全局更新的方式在每只蚂蚁遍历了一条路径之后按式(6)更新各路径的信息素.通常, ACS 模型在解决 TSP 问题中具有更好的搜索性能,所以一般采用 ACS 模型更新信息素。

计算并比较所有 m 只蚂蚁的路径 L_k , 记录当前的最短路径。

步骤4. 判断是否满足终止条件.当迭代计数器 t 达到最大迭代次数时,循环结束,比较每次循环最短路径,输出为所求的最优解.否则,清空禁忌表,转至步骤2.

3.2 蚁群系统求解 TSP 问题

蚁群系统(ant colony system, ACS)是对蚂蚁系统的主要改进之一^[10], ACS 主要从3个方面对 AS 进行了改进。

(1) 状态转移规则的不同

在 ACS 算法中,位于城市 i 的蚂蚁 k 使用伪随机比率移动到下一个访问的城市 j 的选择规则如下:生成一个随机变量 $q \in (0, 1)$ 和一个常量 $q_0 \in (0, 1)$,若 $q \geq q_0$,则按 AS 中式(4)选择下一个城市 j ,以扩大搜索空间;若 $q < q_0$,则选择当前转移概率最大的城市,按式(9)选择城市 j :

$$j = \arg \max_{s \in allowed_k} \{(\tau_{is})^\alpha \cdot (\eta_{is})^\beta\} \quad (9)$$

(2) 更改信息素全局更新规则

与 AS 不同的是, ACS 只对当前循环中找到的最优路径的边根据式(10)进行信息素更新:

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \rho \Delta \tau_{ij}^{gb} \quad (10)$$

其中, $\Delta \tau_{ij}^{gb} = 1/L_{gb}$ 是全局最优路径长度的倒数。

(3) 引入信息素局部更新规则

当蚂蚁 k 从城市 i 移动到下一个城市 j 之后, 边 (i, j) 上的信息素按式 (11) 进行更新:

$$\tau_{ij}(t) = (1 - \gamma)\tau_{ij}(t) + \gamma\tau_0 \quad (11)$$

其中, $\gamma \in (0, 1)$ 是一个常数, τ_0 是各路径上信息素的初始值.

若 $\tau_{ij}(t) < \tau_0$, 则 $\tau_{ij}(t) = (1 - \gamma)\tau_{ij}(t) + \gamma\tau_0 > (1 - \gamma)\tau_{ij}(t) + \gamma\tau_{ij}(t) = \tau_{ij}(t)$, 即如果路径 (i, j) 上的信息素浓度已少于初始值时, 增强路径 (i, j) 上的信息素浓度, 增加后面蚂蚁选择路径 (i, j) 的可能性.

若 $\tau_{ij}(t) > \tau_0$, 则 $\tau_{ij}(t) = (1 - \gamma)\tau_{ij}(t) + \gamma\tau_0 < (1 - \gamma)\tau_{ij}(t) + \gamma\tau_{ij}(t) = \tau_{ij}(t)$, 即将会减少路径 (i, j) 上的信息素浓度, 降低了此路径对其他蚂蚁的影响力, 增强了蚂蚁对没有被选择过的路径的搜索能力, 从而实现搜索的多样化.

在之后的研究中, ACS 还引入了 3-opt 启发式策略, 明显地提高了算法的性能.

3.3 最大-最小蚁群系统求解 TSP 问题

根据信息素更新策略的差异, 德国学者 Stützle 等人^[11] 做出另一个改进, 提出了最大最小蚁群算法 (max-min ant system, MMAS). MMAS 对基本蚁群算法的改进主要在于以下 3 个方面.

(1) 与 ACS 一样, MMAS 在每次迭代完成后, 只允许本次迭代中最优蚂蚁 (在此迭代中构造最短路径的蚂蚁) 或迄今为止的最优蚂蚁按照式 (12) 更新信息素, 加快了算法的收敛速度. MMAS 取消了信息素的局部更新规则:

$$\tau_{ij}(t+1) = \rho\tau_{ij}(t) + \Delta\tau_{ij}^{\text{best}} \quad (12)$$

其中, $\Delta\tau_{ij}^{\text{best}} = 1/L^{\text{best}}$ 是要增加的信息素的量, 被定义为本次迭代最优解或全局最优解的倒数.

(2) MMAS 中约束路径信息素浓度值介于 $[\tau_{\min}, \tau_{\max}]$ 范围内, 如式 (13):

$$\tau_{ij}(t) = \begin{cases} \tau_{\min}, & \text{if } \tau_{ij}(t) \leq \tau_{\min} \\ \tau_{\max}, & \text{if } \tau_{ij}(t) \geq \tau_{\max} \end{cases} \quad (13)$$

既增加了对最优解探索的可能性, 又避免了信息素为零或持续积累的现象.

(3) 在算法开始时设置信息素初始化为其浓度上限 τ_{\max} , 以确保在算法的开始阶段增大解的搜索范围, 充分探索搜索空间, 避免早熟或过早停滞.

4 蚁群算法的改进研究

随着移动环境的复杂性和任务难度的增加, 传统的蚁群算法在很多应用场合已然达不到预期的效果. 随

后, 许多国内外专业研究人员对蚁群算法的缺陷问题进行了深入的研究分析. 为提高其稳定性, 旨在降低其复杂度及深度来优化算法, 对蚁群算法的发展做出了重大贡献. 针对蚁群算法在求解过程中具有收敛速度慢、难以扩展搜索空间、容易停滞和陷入局部最优等缺点, 本文从几个方面介绍了蚁群算法的改进形式, 主要从蚁群算法的路径构建和信息素更新两个主要阶段来构建最优路径进行切入, 改进策略主要从算法结构改进, 信息素初始化与更新策略的改进, 参数选择与优化策略, 融合蚁群算法等提高算法的优化能力方面来进行综述, 并介绍了几个蚁群算法在其他领域的扩展应用, 以期为之后蚁群算法的进一步改进优化提供理论依据.

4.1 蚁群算法结构和框架的改进

4.1.1 基于基本蚁群算法的扩展

许多学者将基本蚁群算法提供的基础框架作为研究基础, 在其框架和结构上提出了许多性能优异的改进措施, 目标是获得比 AS 更好的性能. 如 Dorigo 等人提出的蚁群系统 (ACS)^[10] 从 3 个方面改进基本蚁群算法. Stützle 等人提出的最大最小蚁群系统 (MMAS)^[11], 将信息素浓度控制在一定范围内, 防止信息素值溢出而导致算法停滞. Bullnheimer^[12] 提出的带精英策略的蚂蚁系统 (ant system with elitist strategy, ASelite), 仅采用最优解对信息素进行更新, 同时对搜索过程进行排序, 根据排序不同更新信息素; 和基于排序的蚂蚁系统 (ant system with elitist strategy and ranking, ASrank) 引入了一种排序策略, 通过加权对每个蚂蚁迭代后的路径长度按从小到大进行排序, 只有顶部的蚂蚁才能更新信息素, 可以使信息素的差距变大, 从而加快了收敛速度. 然而, 这些经典改进算法并没有很好的解决过早收敛和长搜索时间等问题, 针对这些问题, 大量学者借鉴已提出的改进策略提出了许多新的改进方案.

Pendharkar^[13] 给出了元启发式蚁群优化算法 (ACO-MH) 的定义, 为求解复杂组合优化问题给出了元启发式的应用范例, 提供了通用算法框架.

Escario 等人^[14] 在通用蚁群优化算法框架的基础上提出了一种蚁群扩展算法. 其并不采用经典蚁群算法通常采用的构造图, 而是使用了一种基于状态空间探索的方法来进行搜索. 为了促进探索, 算法中使用了可以选择启发式或信息素的探索蚁和只使用信息素的觅食蚂蚁以及在搜索过程中执行一种监督政策来控制每种蚂蚁的数量, 通过种群的平衡来控制搜索的平衡.

Duan 等人^[15] 在确定性选择和随机选择相结合的

基础上采用了动态蒸发因子策略提出了有利于全局搜索的动态运动概率规则. 并引入信息素自适应调整策略, 将 ACO 算法中的正反馈机制适当地进行抑制, 以减少局部最优解和最差解在相应路径上信息素的差值, 有效地解决了扩展搜索与寻找最优解之间的矛盾.

Rokbani 等人^[16]提出了一种针对 TSP 等 NP 组合优化问题的双启发式自适应模式的广义框架. 提出了 4 种新的融合模式, 包括两种启发式方法, 其中一个负责问题, 另一个负责参数拟合: 基于授粉蚂蚁监督的局部搜索算法 ASFPA-Ls, 基于局部搜索的认知蚂蚁粒子群监督的 Co-ASPSO-Ls, 其中认知粒子群监督蚁群算法参数的自适应; So-ASPSO-Ls 是一种混合算法, 其中社会粒子群被用于自适应蚁群算法和混沌 ASPSO-Ls, 而混沌粒子群被引入自适应蚁群算法中.

对于动态优化问题 (DOPs), 传统的 ACO 算法不能很好地适应动态变化, 可能会随着目标函数、决策变量、问题实例、约束时间等而发生变化, 这种不确定性可能会改变最优状态. Mavrovouniotis 等人^[17]提出了基于移民方案的 ACO 算法框架, 将新的个体通过移民引入到当前种群中, 用 3 种移民方案集成到 ACO 算法中来解决动态旅行商 (DTSPs) 问题. 通过对不同 DTSPs 情况下算法的实验研究. 移民提高了 DTSPs 的 ACO 表现, 在产生的多样性和转移的信息之间取得了良好的性能. Chowdhury 等人^[18]提出了一种新的蚁群优化框架来解决动态旅行商问题. 为了将知识转移到以往环境中的信息素路径中来保持多样性, 开发了一个基于 ACO 的增强元启发式算法, 此算法基于自适应大邻域搜索 (ALNS) 算法, 在 ACO 框架内生成移民, 目标是有效地解决 DTSP 问题.

4.1.2 利用机器学习方法优化蚁群算法结构

此外, 还有一些研究者利用机器学习的方法来优化蚁群算法的结构. 如王原等人^[19]基于深度学习方法, 提出了一种混合优化启发式算法框架. 该方法将问题实例中的特征进行提取之后, 来代替蚁群算法中的启发式信息矩阵, 利用蚁群算法在替换过的特征矩阵中搜索解空间. Erol 等人^[20]利用人工神经网络动态的适应蚁群算法的信息素和启发式信息这两个重要因素进行优化. 蚁群算法先以两个重要因子的不同值运行, 再用神经网络对蚁群算法得到的所用参数的结果对其进行更新, 根据解预测可能的最优参数值, 再作为反馈信息发送给蚁群算法, 提高了蚁群算法的求解质量.

反向学习是机器学习中的一种新概念, 其主要思

想是在当前迭代后计算所有相反的解, 然后在生成的解及其相反的解中, 选择最优解进行下一轮迭代. Zhang 等人^[21]基于反向学习的反向路径构造方法, 利用相反路径的信息, 提出了 3 种不同的基于相反路径的 ACO 框架, 使 ACO 不再局限于局部最优解, 避免了过早收敛, 并提高了其性能. 但是改进后的算法要求所有蚂蚁都参与更新信息素, 而目前许多算法使用最佳蚂蚁来更新信息素, 因此改进后的 3 种算法在运行时间上没有明显增加, 其与 AS 的时间复杂度相同, 推广到更多的蚁群算法时将存在一定的局限性.

如果采用单一的固定模式去改变蚁群算法的路径构建策略和信息素更新策略不具有灵活性, 不能很好地解决算法过早收敛等问题. 上述改进方法主要都是以基本蚁群算法为基础, 通过平衡求解效率和求解质量^[14]、避免陷入局部最优^[21]、提高搜索效率^[15]等几个方面对算法的结构进行扩展和改进, 能够有效地提高全局搜索能力、在一定程度上加快收敛速度, 提高解的质量. 但是蚁群算法的发展仍处于起步阶段, 目前还没有系统的理论分析, 对其有效性还没有给出合理的数学解释, 因此蚁群算法的改进总结模型有待进一步研究. 此外, 未来的研究方向还可以在扩大随机搜索程度、N-opt 局部随机搜索、设计分布式控制等机制来优化算法. 除了改进算法机制外, 合理的分配算法参数也是提高算法性能的另一有效途径.

4.2 算法参数的改进

蚁群算法对参数的变化很敏感, 蚂蚁通过重要参数值来扩展其路径, 表 1 给出了 ACO 算法的关键参数, 并做了简要介绍, 它们在算法中的功能和位置信息可在第 3 节中找到.

无论选择元启发式或者仿生技术来解决问题, 性能总是受算法参数的影响. 在蚁群算法中, 蚂蚁的行为主要由路径上的启发信息和信息素决定. 在优化过程中, 基本蚁群算法的参数值通常是固定的, 参数值的设置直接影响到了算法的收敛性. 如果参数值不协调的话, 蚂蚁要么依赖启发式信息, 要么依赖信息素. 在这种情况下, 算法通常会陷入局部极小值. 此外, 对于不同的问题, 甚至同一问题的不同实例, 参数值也往往有所不同. 通常给定一个问题, 为了确定算法参数的最佳组合, 需要进行大量的实验来选取.

近年来, 蚁群算法的参数设置已经被广泛研究. 吴春明等人^[22]通过实验分析了蚁群算法中的参数对实验结果的影响, 但并没有找到针对不同问题来设置最优

参数的方法。

Cheong 等人^[23]用 OpenCI 框架预先实现的蚁群算法在不同参数设置下的表现,讨论了不同参数的变化如何影响蚁群算法的优化过程,通过改变蚁群大小和 α 等参数,分析了 5 种不同规模 TSP 问题的最佳结果,并与其他两个算法的结果进行比较。然而,这些结果并没有显示出如何获得最佳结果的明确模式。

向永靖^[24]通过仿真实验得出针对不同规模的 TSP 问题,各个参数较为合理的取值范围。

Gan 等人^[25]为了寻找蚁群算法的参数配置关系,通过正交实验进行微生境蚁群优化算法得到的最优参数配置显示了优良的性能。

Peker 等人^[26]用方差分析确定了蚁群算法中参数对系统的影响,并利用 Taguchi 方法来确定蚁群算法中较好的参数配置,此方法可以用较少的实验次数得到算法的最优或接近最优的结果,节省算法的时间和成本。

Blagoveshchenskaya 等人^[27]致力于蚁群算法参数的动态自适应,提出了使用遗传算法进行参数更新的蚁群优化方法,并定义了一个新的参数 $P(M_k)$ 代表蚂蚁 k 选择参数的概率,用交叉算子实现所选参数的位交叉,变异算子随机改变每个参数,并实时找到其中最好的参数对其更新,使得该算法不太容易受到参数的影响,并对解决任何配置的组合问题更加通用。

针对参数对蚁群算法性能的影响,Wang 等人^[28]

提出了针对 TSP 的混合共生生物搜索算法 (SOS) 和蚁群算法。该算法在 ACO 分配了一定的参数后,剩余的参数通过 SOS 进行自适应优化,使 ACO 找到最优解或接近最优解,大大降低了分配 ACO 参数的复杂性。

Salem 等人^[29]对蚁群算法中蚂蚁数量、城市数量、迭代次数和蒸发速率这几个参数对算法开销的影响进行了重点研究,基于调整多个不同的实现算法的主要参数来获得更好的性能并得出各不同参数对算法的不同影响。

上述研究的一般都是将算法中的参数由固定不变模式改为动态调整模式的改进策略,采用参数自适应的优化方法来影响蚁群算法的行为,用较少的实验次数和成本来获取比较好的参数配置。在考虑蚁群算法对不同问题的不同变化时,参数值自适应的主要方法有预先设定参数的变化^[30],基于信息素轨迹^[31]或解质量的自适应参数变化^[32],基于搜索的参数自适应^[33]等。每个参数以不同的方式影响算法的执行效果。由于各参数之间的耦合关系尚不清楚,因此现阶段的研究没有分配 ACO 参数的理论依据,确定 ACO 参数的最佳组合是对算法性能的关键影响。未来的工作需要更深入的研究如何通过改变不同的参数来实现更好的解决方案、结构参数如何变化以及它们对算法性能的影响。例如,预测大多数蚂蚁的构造参数值是否在增加,而不是只有少数蚂蚁在终止时在增加;此外,所提出的算法还可以通过系统的参数变化和控制策略来增强。

表 1 ACO 的关键参数

参数	定义	描述
η_{ij}	启发式函数	蚂蚁在每一步移动的最佳路径,计算为从节点 i 到节点 j 的路径上的距离(成本)的倒数
$\tau_{ij}(t)$	信息素强度	节点 (i, j) 间路径上的信息素水平
α	信息素影响因子	信息素轨迹的相对重要程度,反映了蚂蚁在选择路径时积累的信息素的影响程度。 α 越大,蚂蚁在选择下一路径时,信息素的影响越大,即蚂蚁越倾向于选择以前走过的路径
β	启发式函数影响因子	能见度的相对重要程度,反映了蚂蚁在运动过程中启发信息(两个节点位置之间的距离)在蚂蚁选择路径中的影响程度。 β 越大,计算下一路径的状态转移概率时,启发式的作用就越大,即尽管路径上信息素量很大,但蚂蚁总会以很高的概率选择更近的城市
ρ	信息素挥发系数	这个系数控制信息素挥发的速度。 ρ 过小时,各路径残留信息素过多,导致无效路径被继续搜索; ρ 过大时,信息素低的路径不被选择,影响算法全局搜索。对收敛速度和搜索能力有很大的影响

4.3 基于信息素的改进策略

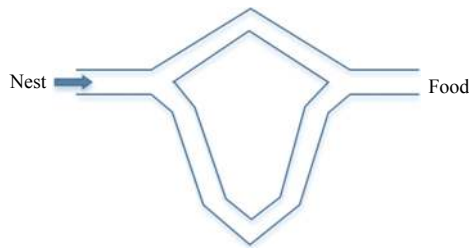
首先通过介绍一个著名的双桥实验,来说明信息素的作用机制和对路径选择的影响。这个实验是 Goss 等人^[9]在 20 世纪 80 年代末通过一个连接蚂蚁巢穴和食物来源的双桥来完成的。该实验做了两个方案,方案 1 中如图 3 所示,从实验开始之前就给出了一个长路径和一个短路径,长路径是短路径长度的两倍,如图 3(a)。

从实验中得出,90% 以上的实验运行中,大部分蚂

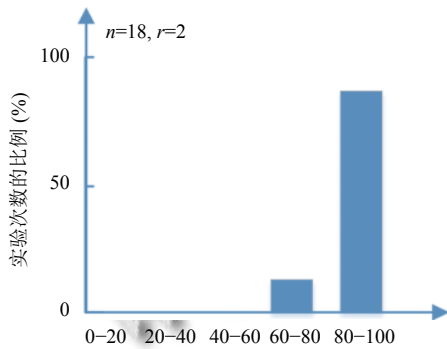
蚁(80%–100%)的流量最终集中在短分支上,数据如图 3(b) 所示,其中 r 是两个分支之间的比值。

方案 2 中如图 4 所示,开始时只有较长路径存在,在一段时间(30 min)该路径上形成了稳定的信息素轨迹后提供一个较短路径,如图 4(a)。此方案是用来分析当蚁群在收敛后,提供一条更好(即更短)的路径时,会发生什么。通过实验数据观察到,在几乎 50% 的实验中,只有 0–20% 的蚂蚁选择了新提供的较短路径。较

短路径并不经常被选择,因此蚁群大部分仍被困在最初提供的长路径上,如图4(b)所示。

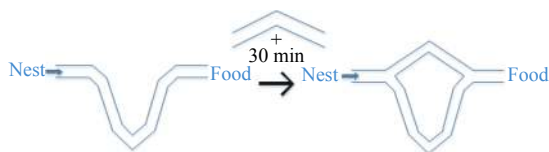


(a) 实验开始前给出的两个路径

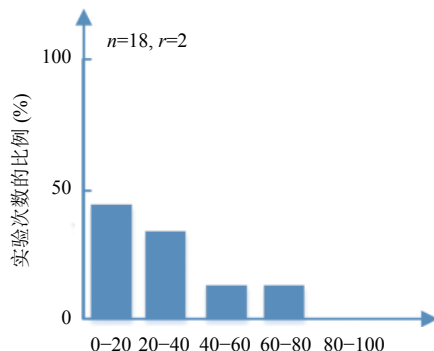


(b) n 个实验中选择短路径的蚂蚁的比例

图3 方案1



(a) 实验开始前只有长路径, 30 min 后加入较短路径



(b) n 个实验中选择短路径的蚂蚁的比例

图4 方案2

大多数蚂蚁继续选择长路径的现象可以用两个原因来解释:长路径上信息素浓度高和信息素蒸发缓慢。首先,即使提供了一个更短的路径,长路径上的高水平信息素浓度(与短分支上的信息素浓度相比)导致了一

种自催化行为,会继续强化长路径。其次,缓慢的信息素蒸发速率不会使蚁群忘记他们最初选择的次优路径,从而阻止了新的和更短的路径被发现和学习。

因此,信息素浓度和信息素蒸发速率是这个过程中的关键参数,因为它们控制了新的(希望是更好的)路径的路径探索和已经建立的路径的路径开发之间的权衡。

4.3.1 信息素初始化策略的改进

传统的 ACO 算法中,初始阶段主要使用固定数量的信息素来更新,这就造成了蚂蚁在对初始路径的搜索中缺乏信息素有一定的盲目性,忽略了解分布特征,从而使算法收敛缓慢。

为使算法在初始阶段能够具有良好的搜索性能,乔东平等人^[34]将城市间的距离信息引入到初始信息素中,与信息素总量构成初始信息素分布矩阵,减少了非最优路径被选择的概率。

陈颖杰等人^[35]采用贪婪策略在算法的初始化阶段构建次优路径并增加信息素浓度,使信息素在不同路径上的初始分布在搜索初期就起到指导作用,并对每次迭代后的最优路径利用遗传突变操作,找到一条更优的路径之后对其自适应的调整信息素增量。

Ning 等人^[36]针对算法陷入局部最优而接近搜索停滞状态时,提出了一种新的信息素平滑机制来重新初始化信息素矩阵,以平衡信息素在每条路径上的差异,有助于蚂蚁在接近停滞时期继续搜索路径,并在一定程度上提高了停滞期后算法的质量。

Luo 等人^[37]提出了一种改进初始信息素的方法,该方法基于当前位置和起点,终点连接的相对距离,来计算出非均匀分布的初始信息素值。解决了在初始阶段进行盲目搜索以提高收敛速度的问题。

针对 ACO 初始阶段缺乏信息素和收敛缓慢的缺点,Fei 等人^[38]引入图卷积网络生成更好的解,并通过信息素转换策略将其转换为 ACO 初始路径上的信息素提高算法的初始求解速度。

4.3.2 信息素更新策略的改进

传统的 ACO 算法主要使用固定数量来更新信息素,正反馈会加速算法的收敛速度,使蚂蚁沿着特定的路径移动,失去探索其他潜在路径的能力,使算法易于陷入局部最优,从而导致过早收敛。

Gao^[39]提出了一种极化所有路径的信息素密度的新的信息素更新策略。通过增加良好路径和普通路径的信息素多样性,对所有路径的信息素密度进行了极化,来解决 ACO 的计算效率低和过早收敛的问题。

顾竞豪等人^[40]在蚁群系统的基础上引入了3种改进策略,添加了带导向的信息素来获取全局的导向信息,加强了算法的全局寻优能力和算法的稳定性。

Ekmecki^[41]提出了蚁群记忆优化解(ACO-MBS)方法,一种根据记忆代价更新信息素轨迹的新ACO模型。根据组合优化问题的复杂性、问题参数的特征和估计的解空间,在信息素更新中可以使用如图5所示线性、S型(Sigmoid函数)和平方3种不同的传递函数来更新信息素的新ACO模型。线性函数用于将信息素值与解的成功与否直接联系起来;S型函数用于对好解和坏解进行分组;平方函数的使用是为了得到高质量的解决方案。

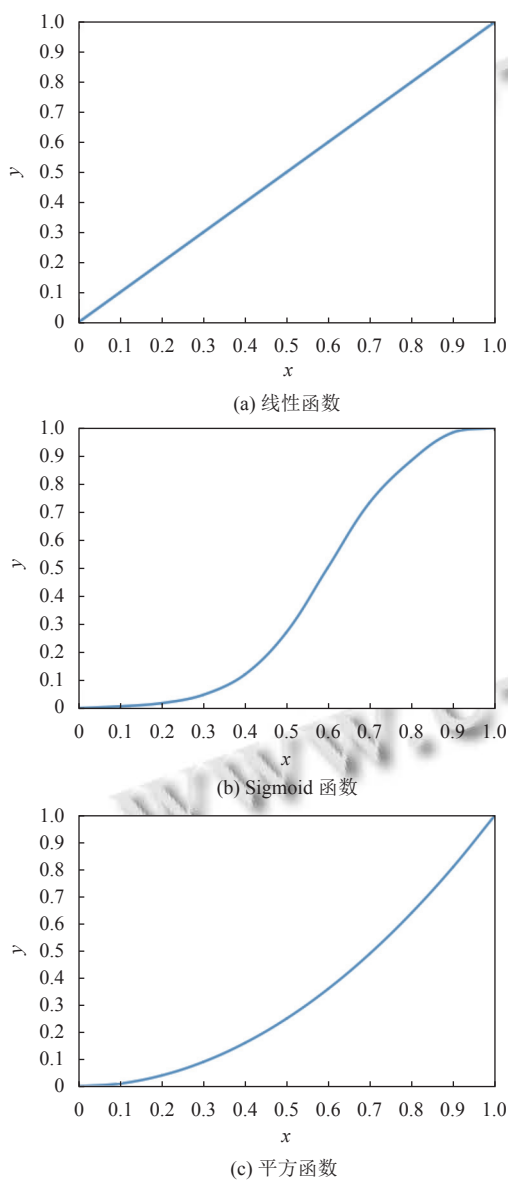


图5 传递函数

4.3.3 基于混合反馈机制策略

上述文献大多数都是对信息素更新进行动态调整或在信息素初始分布阶段进行调整,仍然属于单一正反馈机制下的改进。然而过度使用正反馈机制来提高收敛速度仍然存在容易获得局部最优解而不易跳出的问题,如果正反馈信息素过度集中在良好路径上,来引导大量蚂蚁选择高信息素路径,导致其他信息素边缘被选择的概率降低,即探索未知区域的概率可以忽略不计。

针对正反馈机制对算法的影响,Ye等人^[42]扩展了蚁群优化算法设计了一种改进的具有增强负反馈机制的ACO算法来解决随机二元组合问题,算法的出发点是利用负反馈来提高求解的多样性,它在每一代结束时都基于原始的正反馈机制更新负反馈信息素矩阵,不仅将信息素沉淀在良好路径的顶点上还使人工蚂蚁将信息素放置在当前一代最差解决方案的路径上从而加强对未知区域的探索,可以有效地抑制算法向局部最优解的快速收敛。虽然Ye等人^[42]提出的改进算法中的转移规则在搜索过程中将负反馈信息素浓度纳入基本蚁群算法中,但它仍然仅根据信息素浓度选择下一个城市,如果信息素在某些路径上始终保持较高浓度,则该算法还是很容易陷入局部最优。针对这一问题,Li等人^[43]提出了一种新的状态转换规则的伪动态搜索机制,该机制在信息素传递规则中引入一个角度,使多个角度较小的城市也被包含在下一个候选城市的列表中,同时更新最差路径和最优路径上的信息素浓度,并增强最优路径上信息素浓度的权重。通过实验数据验证了该算法可以更好地提高收敛精度和收敛速度。

冯振辉等人^[44]在传统蚁群算法的基础上设计了基于刺激-响应模型的负反馈平衡机制,利用补充负反馈分工行为来平衡正反馈机制下的全局搜索能力,并通过引入了不依赖于信息素搜索路径的扩展蚂蚁,扩展蚂蚁基于已知路径的信息进行组合交换来构造新的路径,对个体分工的信息素更新策略进行了调整。

上述文献都在一定程度上避免算法陷入局部最优解等缺点上进行了相关改进,主要是通过改进信息素更新规则和算法的启发式信息来有效地提升算法的收敛速度等,但引入启发式信息可能会造成算法规模和复杂度的增加。单策略的改进模式不够满足现在问题对算法的要求,大部分学者对路径构建和信息素进行多策略的改进方法,通过对初始信息素分布、路径选择概率、信息素更新等几个方面的改进来提高算法的求解性能和求解效率。此外还有众多针对信息素调整策略的改进算法被不断提出。如采用信息素二次更新

策略^[45], 引入梯度下降法选择路径^[46], 将二维凸包信息融入信息素更新中^[47]等改进策略都在一定程度上提高了算法的寻优性能。

4.4 混合蚁群算法的研究发展

另一种算法的最近的趋势是蚁群算法与其他优化算法的结合。由于蚁群算法优良的分布式并行计算特性, 使其易于与其他优化方法相结合, 他们都试图将原始算法组合后的最佳特征进行挖掘, 融合后的优化算法在寻找最佳路径的过程中能够同时兼顾全局和局部, 增强了算法对解决优化问题的整体性能, 研究者们对蚁群算法与其他算法的融合方面展开了大量的研究工作。

遗传算法和蚁群算法可以在可接受的时间内找到高质量的解决方案, 很多研究者利用遗传算法交叉、变异和较强的全局寻优能力和蚁群算法的正反馈、分布式、并行计算机制进行结合来克服各自缺陷进行优势互补, 融合成新的混合启发式算法。Aydoğan 等人^[48]基于遗传算法的生物操作和 MMAS 中蚂蚁的觅食行为设计了一种新的 HGAMMAS 算法。为了将 MMAS 结果引入到遗传算法过程中, 将 MMAS 算法的一次寻优结果转化为基因-染色体的形式, 生成遗传算法的初始种群, 如图 6 所示。

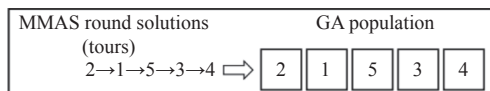


图 6 每只蚂蚁获得的旅行染色体演示

但是这种先蚁群后遗传的静态融合算法中具有 MMAS 对种群的影响会随遗传代数增加而减少的局限性。Chen 等人^[49]利用遗传算法, 通过选择、交叉和变异演化操作, 根据路径节点的分布将种群进化信息转换为蚁群算法所需的初始信息素值。实验测试表明, 融合算法的迭代次数较少, 降低了计算时间成本。但是这种先遗传后蚁群的静态融合算法将遗传算法的种群进化输出作为先验信息输入给 ACO 算法, 一旦遗传算法在前期进化时陷入局部最优, 那么将种群进化信息转换为初始信息素的蚁群算法在之后的运行过程中, 也很难跳出局部最优。针对静态融合算法中存在的缺陷, 陶丽华等人^[50]在蚁群算法内部嵌入遗传因子, 将遗传算法的重插入子代进行改进之后, 用种群进化后的最优解来更新信息素, 实现了与蚁群算法的动态融合, 提高了此动态融合算法的全局寻优能力和搜索效率。但是此算法对于较大规模的问题时, 整体性能和寻优效果仍待提高。

虽然粒子群算法 (PSO) 和蚁群算法通常能找到较

好的 TSP 的解, 但它们的解仍然受到随机初始化种群和参数配置等因素的影响。Mahi 等人^[51]利用 PSO 来优化影响 ACO 性能的参数, 检测蚁群算法中用于城市选择操作的参数 α 和 β 的最优值, 确定城市间信息素和距离的重要性, 并引入了 3-opt 启发式方法以改进局部解。从实验结果中看出, 随着蚂蚁数量的减少, 该方法的性能会有所提高。但由于 ACO 算法降低了局部最小值, 因此 3-opt 算法无法对城市选择操作进行改进。Rokbani 等人^[52]结合了引力粒子群优化 (PSOGSA) 和蚁群算法, 称为引力粒子群优化监督下的蚂蚁与局部搜索算法 (PSOGSA-ACO-LS)。其中蚁群算法采用 PSOGSA 来优化蚁群设置, 并采用 2-opt 局部搜索机制来改进蚁群的局部解, 实现了 ACO 参数的有效自适应。

Stodola 等人^[53]结合了蚁群算法和模拟退火算法的原理, 提出了一种新的混合元启发式算法, 通过每一代蚂蚁找到的最佳路径 r^{best} 作为模拟退火的初始解 R^{SA} 输入算法。如果 SA 算法对初始解进行改进, 则用新的改进解替换 r^{best} ; 否则, SA 算法返回的解与初始解相同。再用改进后的解更新信息素矩阵。通过对比实验可以得出混合算法的改进效果, 但是代价是需要更长的执行时间。

由于某些边缘上的高信息素轨迹可能会导致算法的快速收敛并进入局部最优, Zhang 等人^[54]将云模型 (CM) 嵌入到 ACO 算法的框架结构中, 得到一种新的 ACO & CM 混合算法。该算法采用基于 CM 的信息素更新策略, 通过调整边缘上的信息素轨迹来寻找高质量的区域, 提高 ACO 算法的性能。

由于 MMAS 的性能很大程度上受相关参数的影响, 而共生生物搜索算法 (SOS) 本身不包含参数, 因此, Sheng^[55]提出了一种混合 SOS-MMAS 方法, 首先用 MMAS 作为提高任务调度效率的基本算法, 同时在 MMAS 中引入 SOS 来对 MMAS 中的两个关键参数 α 和 β 进行优化。对求解 TSP 问题具有较好的优化效果和较强的鲁棒性。

虽然一般的 ACO 算法能够给出高质量的解, 但是在某些情况下, 混合算法被证明也是必要的, 能够与蚁群算法融合的算法还包括萤火虫算法^[56]、人工势场法^[57]、布谷鸟搜索算法^[58]等。这些融合策略能够有效提升算法的寻优能力, 但是在一定程度上增加了算法的复杂度, 同时也需要更多的优化时间。

4.5 蚁群算法的扩展应用

蚁群算法在解决实际优化问题中的改进策略在不断被研究者们提出, 使算法的模型越来越丰富和完善。

并且随着对蚁群算法的不断深入研究和改进,使其从最开始应用于经典的 TSP 求解最短路径问题,到求解各领域的优化问题以及具有不同边界条件的优化问题,使其功能更强大、应用领域更广泛。而不同的应用场合各具有不同的特点和需求,

在企业生产中常遇到的车间调度问题 (job-shop scheduling problem, JSP)^[59], 是一种典型的 NP 困难问题通常将其转化为寻找最佳路径问题, 其最优解的求法随复杂度的增加而增加。它在调度和 CIMS 领域受到了广泛的关注。Deepalakshmi 等人^[60] 介绍了 ACO 对 JSP 问题的适用性并详细分析了在 JSP 问题中的作用和影响, 之后引入了 Sigmoid 函数极限作为一种与伪神经框架融合的极限建立了新信息素更新模型, 获得的结果支持了使用所提出的算法来解决 JSP 的充分性。

车辆路径问题 (vehicle routing problem, VRP)^[61] 中, 蚁群算法对合理分配车辆资源, 提高工作效率以及缓解交通压力有重要的研究意义。刘紫玉等人^[62] 引入了节约矩阵和分段函数来提高 ACO 算法的全局搜索能力和收敛速度, 综合了算法流程和参数设置对蚁群算法在 VRP 问题中进行了改进研究, 为解决 VPR 问题提供了新思路。

机器人路径规划问题^[37,56,57] 中, 寻找从起始位置到目标位置的最优或次优路径, 蚁群算法是一种常用的解决方案。Miao 等人^[63] 在 ACO 的状态转移概率中

引入了角度引导系数和障碍物排除系数, 并将启发式信息自适应调整因子和自适应信息素挥发因子引入了信息素更新规则中, 提出了一种改进的自适应蚁群算法 (IAACO), 实现了机器人路径规划的全面全局优化, 具有较高的路径规划实时性和稳定性。

目前蚁群算法的研究大多针对与二维平面的路径规划, 对三维空间的研究相对较少, 如将蚁群算法应用于无人机三维路径规划^[64]、水下机器人三维路径规划^[65] 等。

5 总结与展望

虽然在上述文献中针对 TSP 问题提出了诸多不同版本的改进蚁群算法, 但这些改进算法主要包括 3 个方面: 首先, 路径选择是蚁群算法的一个重要组成部分, 因此许多研究者关注蚂蚁的状态转移规则设计^[42,43,46], 研究了不同转移规则对蚁群算法的影响。第二, 信息素更新策略是 ACO 算法性能的决定性因素, 信息素更新问题是蚂蚁如何更新其路径的信息素强度^[21,34-41]。第三, 蚁群算法的发展已经有了相当大的进步, 但也有其优点和缺点。因此, 许多研究试图提高混合算法^[48-58] 和新的元启发式算法的性能, 如蜂群算法等。为了比较分析, 将上述内容进行了简单总结, 来评价和总结蚁群算法改进后的优劣, 如表 2 所示。

表 2 改进蚁群算法的方法及优缺点

改进类型	改进措施	文献	优点	缺点
状态转移规则	动态运动概率规则	[15]	提高全局寻优能力, 扩大搜索空间	搜索时间较长, 优化效率不高
	伪动态搜索机制	[43]		
路径构建	基于状态空间探索	[14]	通过种群的平衡控制搜索的平衡	信息素更新复杂, 适应性比较局限
	反向路径构造	[21]	不局限于局部最优解, 避免过早收敛	
参数选择与优化	枚举法实验分析不同参数配置	[22-24,29]	得到不同参数配置对算法结果的影响	没有找到设置最优参数的方法
	结合其他方法得到最优参数配置	[20,25,26]	用较少的实验次数和成本获取比较好的参数配置	没有分配参数的理论依据
	参数动态自适应	[27,28,30-32]		
信息素初始化	非均匀分布初始信息素浓度	[34,37]	使信息素的分布在搜索初期就起到指导作用	不利于搜索多样性, 易限于局部最优
	次优路径构建初始分布	[35,38]		
信息素更新策略	信息素自适应调整	[15,20,63]	解决了扩展搜索与寻找最优解之间的矛盾	算法结构复杂, 长时间搜索, 降低了收敛速度
	信息素平滑机制	[36]	解决算法的停滞状态	
	增加信息素多样性	[18,39]	避免过早收敛	
	带导向信息素	[40]	加强全局寻优能力	
	用不同传递函数更新信息素	[41]	提高探索能力和开发能力	
	增强负反馈机制	[42,44]	抑制算法向局部最优的收敛速度	
融合蚁群算法	增强最优路径信息素浓度	[13,43]	提高收敛速度和精度	易限于局部最优
	初始解构建初始路径分布	[48,49,53]	使算法在初始阶段就具有良好的搜索性能	增加了算法的复杂度, 需要更长的执行时间
	改变信息素分布或更新规则	[50,54]	提高了寻优能力和搜索效率	
	优化参数	[51,52,55]	使算法具有较好的优化效果和较强的鲁棒性	参数变化规律缺乏通用性

由于文献中通过蚁群算法解决 NP 困难问题的初步结果令人鼓舞, 尽管蚁群算法没有为这类问题提供最佳的解决方案, 但它仍然在算法和应用学科的进一步研究中发挥着重要作用。

改进传统算法或者探索新的融合算法已成为当前研究的重点, 未来的研究还应包括对最近的启发式技术和仿生技术等算法的融合等。其他未来工作可能包括调查蚂蚁的数量和最近邻集如何影响算法的性能; 预测蚂蚁行为从好到差的波动对算法性能的影响等都是值得研究的, 并将所提出的算法扩展到 TSP 问题的其他变化。

参考文献

- 1 Campbell PJ. The traveling salesman problem: A computational study. *Mathematics Magazine*, 2007, 80(3): 238.
- 2 Beni G. Swarm intelligence. In: Sotomayor M, Pérez-Castrillo D, Castiglione F, eds. *Complex Social and Behavioral Systems: Game Theory and Agent-based Models*. New York: Springer, 2020. 791–818.
- 3 Razali NM, Geraghty J. Genetic algorithm performance with different selection strategies in solving TSP. *Proceedings of the World Congress on Engineering*. London: International Association of Engineers, 2011. 1–6.
- 4 Huang L, Wang GC, Bai T, *et al.* An improved fruit fly optimization algorithm for solving traveling salesman problem. *Frontiers of Information Technology & Electronic Engineering*, 2017, 18(10): 1525–1533.
- 5 Hatamlou A. Solving travelling salesman problem using black hole algorithm. *Soft Computing*, 2018, 22(24): 8167–8175.
- 6 Laguna M. Tabu search. In: Martí R, Pardalos PM, Resende MGC, eds. *Handbook of Heuristics*. Cham: Springer, 2018. 741–758.
- 7 Kennedy J, Eberhart R. Particle swarm optimization. *Proceedings of 1995 International Conference on Neural Networks*. Perth: IEEE, 1995. 1942–1948.
- 8 Dorigo M, Birattari M, Stutzle T. Ant colony optimization. *IEEE Computational Intelligence Magazine*, 2006, 1(4): 28–39. [doi: [10.1109/MCI.2006.329691](https://doi.org/10.1109/MCI.2006.329691)]
- 9 Goss S, Aron S, Deneubourg JL, *et al.* Self-organized shortcuts in the Argentine ant. *Naturwissenschaften*, 1989, 76(12): 579–581. [doi: [10.1007/BF00462870](https://doi.org/10.1007/BF00462870)]
- 10 Dorigo M, Maniezzo V, Colomi A. Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 1996, 26(1): 29–41. [doi: [10.1109/3477.484436](https://doi.org/10.1109/3477.484436)]
- 11 Stützle T, Hoos HH. MAX-MIN ant system. *Future Generation Computer Systems*, 2000, 16(8): 889–914. [doi: [10.1016/S0167-739X\(00\)00043-1](https://doi.org/10.1016/S0167-739X(00)00043-1)]
- 12 Bullnheimer B, Hartl RF, Strauss C. An improved ant system algorithm for the vehicle routing problem. *Annals of Operations Research*, 1999, 89: 319–328. [doi: [10.1023/A:1018940026670](https://doi.org/10.1023/A:1018940026670)]
- 13 Pendharkar PC. An ant colony optimization heuristic for constrained task allocation problem. *Journal of Computational Science*, 2015, 7: 37–47.
- 14 Escario JB, Jimenez JF, Giron-Sierra JM. Ant colony extended: Experiments on the travelling salesman problem. *Expert Systems with Applications*, 2015, 42(1): 390–410. [doi: [10.1016/j.eswa.2014.07.054](https://doi.org/10.1016/j.eswa.2014.07.054)]
- 15 Duan P, Yong AI. Research on an improved ant colony optimization algorithm and its application. *International Journal of Hybrid Information Technology*, 2016, 9(4): 223–234. [doi: [10.14257/ijhit.2016.9.4.20](https://doi.org/10.14257/ijhit.2016.9.4.20)]
- 16 Rokbani N, Kumar R, Abraham A, *et al.* Bi-heuristic ant colony optimization-based approaches for traveling salesman problem. *Soft Computing*, 2021, 25(5): 3775–3794. [doi: [10.1007/s00500-020-05406-5](https://doi.org/10.1007/s00500-020-05406-5)]
- 17 Mavrovouniotis M, Yang SX. Ant colony optimization with immigrants schemes for the dynamic travelling salesman problem with traffic factors. *Applied Soft Computing*, 2013, 13(10): 4023–4037. [doi: [10.1016/j.asoc.2013.05.022](https://doi.org/10.1016/j.asoc.2013.05.022)]
- 18 Chowdhury S, Marufuzzaman M, Tunc H, *et al.* A modified ant colony optimization algorithm to solve a dynamic traveling salesman problem: A case study with drones for wildlife surveillance. *Journal of Computational Design and Engineering*, 2019, 6(3): 368–386. [doi: [10.1016/j.jcde.2018.10.004](https://doi.org/10.1016/j.jcde.2018.10.004)]
- 19 王原, 陈名, 邢立宁, 等. 用于求解旅行商问题的深度智慧型蚁群优化算法. *计算机研究与发展*, 2021, 58(8): 1586–1598. [doi: [10.7544/issn1000-1239.2021.20210320](https://doi.org/10.7544/issn1000-1239.2021.20210320)]
- 20 Erol AH, Er M, Bulkan S. Optimizing the ant colony optimization algorithm using neural network for the traveling salesman problem. *Proceedings of the 2012 International Conference on Industrial Engineering and Operations Management*. Istanbul, 2012. 1695–1701.
- 21 Zhang ZJ, Xu ZX, Luan SY, *et al.* Opposition-based ant colony optimization algorithm for the traveling salesman problem. *Mathematics*, 2020, 8(10): 1650. [doi: [10.3390/math8101650](https://doi.org/10.3390/math8101650)]

- 22 吴春明, 陈治, 姜明. 蚁群算法中系统初始化及系统参数的研究. 电子学报, 2006, 34(8): 1530–1533. [doi: 10.3321/j.issn.0372-2112.2006.08.035]
- 23 Cheong PY, Aggarwal D, Hanne T, *et al.* Variation of ant colony optimization parameters for solving the travelling salesman problem. Proceedings of the 2017 IEEE 4th International Conference on Soft Computing & Machine Intelligence (ISCM). Mauritius: IEEE, 2017. 60–65.
- 24 向永靖. 蚁群算法中参数设置的研究——以 TSP 为例. 现代信息科技, 2020, 4(22): 95–98, 102. [doi: 10.19850/j.cnki.2096-4706.2020.22.025]
- 25 Gan Y, Lan LW, Li S. Study on parameters configuration for ant colony optimization. Advanced Materials Research, 2011, 279: 371–376. [doi: 10.4028/www.scientific.net/AMR.279.371]
- 26 Peker M, Şen B, Kumru PY. An efficient solving of the traveling salesman problem: The ant colony system having parameters optimized by the Taguchi method. Turkish Journal of Electrical Engineering and Computer Sciences, 2013, 21(7): 2015–2036.
- 27 Blagoveshchenskaya EA, Mikulik II, Strüngmann LH. Ant colony optimization with parameter update using a genetic algorithm for travelling salesman problem. Models and Methods for Researching Information Systems in Transport 2020, 2020, (1): 20–25. [doi: 10.24412/1613-0073-2803-20-25]
- 28 Wang Y, Han ZP. Ant colony optimization for traveling salesman problem based on parameters optimization. Applied Soft Computing, 2021, 107: 107439. [doi: 10.1016/j.asoc.2021.107439]
- 29 Salem A, Sleit A. Analysis of ant colony optimization algorithm solutions for travelling salesman problem. International Journal of Scientific & Engineering Research, 2018, 9(2): 570–575.
- 30 Merkle D, Middendorf M, Schmeck H. Ant colony optimization for resource-constrained project scheduling. IEEE Transactions on Evolutionary Computation, 2002, 6(4): 333–346. [doi: 10.1109/TEVC.2002.802450]
- 31 李万庆, 李彦苍. 求解复杂优化问题的基于信息熵的自适应蚁群算法. 数学的实践与认识, 2005, 35(2): 134–139. [doi: 10.3969/j.issn.1000-0984.2005.02.024]
- 32 Cai ZQ, Huang H, Qin Y, *et al.* Ant colony optimization based on adaptive volatility rate of pheromone trail. International Journal of Communications, 2009, 2(8): 792–796.
- 33 Anghinolfi D, Boccialatte A, Paolucci M, *et al.* Performance evaluation of an adaptive ant colony optimization applied to single machine scheduling. Proceedings of the 7th Asia-Pacific Conference on Simulated Evolution and Learning. Melbourne: Springer, 2008. 411–420.
- 34 乔东平, 裴杰, 李浩, 等. 改进蚁群算法求解 TSP 问题研究. 机械设计与制造, 2019, (10): 144–149. [doi: 10.3969/j.issn.1001-3997.2019.10.036]
- 35 陈颖杰, 高茂庭. 基于信息素初始分配和动态更新的蚁群算法. 计算机工程与应用, 2022, 58(2): 95–101. [doi: 10.3778/j.issn.1002-8331.2012-0569]
- 36 Ning JX, Zhang Q, Zhang CS, *et al.* A best-path-updating information-guided ant colony optimization algorithm. Information Sciences, 2018, 433–434: 142–162.
- 37 Luo Q, Wang HB, Zheng Y, *et al.* Research on path planning of mobile robot based on improved ant colony algorithm. Neural Computing and Applications, 2020, 32(6): 1555–1566. [doi: 10.1007/s00521-019-04172-2]
- 38 Fei T, Wu XX, Zhang LY, *et al.* Research on improved ant colony optimization for traveling salesman problem. Mathematical Biosciences and Engineering, 2022, 19(8): 8152–8186. [doi: 10.3934/mbe.2022381]
- 39 Gao W. Modified ant colony optimization with improved tour construction and pheromone updating strategies for traveling salesman problem. Soft Computing, 2021, 25(4): 3263–3289. [doi: 10.1007/s00500-020-05376-8]
- 40 顾竞豪, 王晓丹, 贾琪. 求解大规模 TSP 问题的带导向信息素蚁群算法. 火力与指挥控制, 2018, 43(8): 111–115. [doi: 10.3969/j.issn.1002-0640.2018.08.023]
- 41 Ekmekci D. An ant colony optimization memorizing better solutions (ACO-MBS) for traveling salesman problem. Proceedings of the 2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT). Ankara: IEEE, 2019. 1–5.
- 42 Ye K, Zhang CS, Ning JX, *et al.* Ant-colony algorithm with a strengthened negative-feedback mechanism for constraint-satisfaction problems. Information Sciences, 2017, 406–407: 29–41.
- 43 Li J, Xia Y, Li B, *et al.* A pseudo-dynamic search ant colony optimization algorithm with improved negative feedback mechanism. Cognitive Systems Research, 2020, 62: 1–9. [doi: 10.1016/j.cogsys.2020.03.001]
- 44 冯振辉, 肖人彬. 基于混合反馈机制的扩展蚁群算法. 控制与决策, 2022, 37(12): 3160–3170.
- 45 许凯波, 鲁海燕, 程毕芸, 等. 求解 TSP 的改进信息素二次更新与局部优化蚁群算法. 计算机应用, 2017, 37(6): 1686–1691. [doi: 10.11772/j.issn.1001-9081.2017.06.1686]

- 46 Liang BL, Yang ZM, Qin Y, *et al.* Ant colony algorithm with gradient descent for solving multi-constrained quality of service routing. *Journal of Computer Applications*, 2017, 37(3): 722–729.
- 47 马学森, 宫帅, 朱建, 等. 动态凸包引导的偏优规划蚁群算法求解 TSP 问题. *通信学报*, 2018, 39(10): 59–71. [doi: [10.11959/j.issn.1000-436x.2018218](https://doi.org/10.11959/j.issn.1000-436x.2018218)]
- 48 Aydoğan T, Al-Badri R. A hybrid model of max-min ant system with genetic algorithm for improved to travelling salesman problem. *International Journal of Engineering and Technology*, 2018, 10(3): 254–258. [doi: [10.7763/IJET.2018.V10.1069](https://doi.org/10.7763/IJET.2018.V10.1069)]
- 49 Chen XY, Dai YH. Research on an improved ant colony algorithm fusion with genetic algorithm for route planning. *Proceedings of the 2020 4th IEEE Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*. Chongqing: IEEE, 2020. 1273–1278.
- 50 陶丽华, 马振楠, 史朋涛, 等. 基于 TSP 问题的动态蚁群遗传算法. *机械设计与制造*, 2019, (12): 147–149, 154. [doi: [10.3969/j.issn.1001-3997.2019.12.037](https://doi.org/10.3969/j.issn.1001-3997.2019.12.037)]
- 51 Mahi M, Baykan ÖK, Kodaz H. A new hybrid method based on particle swarm optimization, ant colony optimization and 3-opt algorithms for traveling salesman problem. *Applied Soft Computing*, 2015, 30: 484–490. [doi: [10.1016/j.asoc.2015.01.068](https://doi.org/10.1016/j.asoc.2015.01.068)]
- 52 Rokbani N, Kromer P, Twir I, *et al.* A new hybrid gravitational particle swarm optimization-ACO with local search mechanism, PSOGSA-ACO-Ls for TSP. *International Journal of Intelligent Engineering Informatics*, 2019, 7(4): 384–398. [doi: [10.1504/IJIEI.2019.101565](https://doi.org/10.1504/IJIEI.2019.101565)]
- 53 Stodola P, Michenka K, Nohel J, *et al.* Hybrid algorithm based on ant colony optimization and simulated annealing applied to the dynamic traveling salesman problem. *Entropy*, 2020, 22(8): 884. [doi: [10.3390/e22080884](https://doi.org/10.3390/e22080884)]
- 54 Zhang XX, Shen X, Yu ZQ. A novel hybrid ant colony optimization for a multicast routing problem. *Algorithms*, 2019, 12(1): 18. [doi: [10.3390/a12010018](https://doi.org/10.3390/a12010018)]
- 55 Sheng YF. A computational optimization research on ant colony optimization for the traveling salesman problem. *Journal of Physics: Conference Series*, 2022, 2258: 012006. [doi: [10.1088/1742-6596/2258/1/012006](https://doi.org/10.1088/1742-6596/2258/1/012006)]
- 56 杜力, 徐光辉, 汪繁荣. 自适应萤火虫算法改进蚁群算法的移动机器人路径规划. *河南理工大学学报(自然科学版)*, 2022, 41(2): 124–130. [doi: [10.16186/j.cnki.1673-9787.2021010073](https://doi.org/10.16186/j.cnki.1673-9787.2021010073)]
- 57 李志锟, 赵倩楠. 融合人工势场蚁群算法的移动机器人路径规划. *电光与控制*, 2022, 29(11): 118–124.
- 58 张烈平, 何佳洁, 于滢琳, 等. 基于蚁群算法优化的布谷鸟搜索算法. *微电子学与计算机*, 2018, 35(12): 21–26. [doi: [10.19304/j.cnki.issn1000-7180.2018.12.005](https://doi.org/10.19304/j.cnki.issn1000-7180.2018.12.005)]
- 59 Nazif H. Solving job shop scheduling problem using an ant colony algorithm. *Journal of Asian Scientific Research*, 2015, 5(5): 261–268. [doi: [10.18488/journal.2/2015.5.5/2.5.261.268](https://doi.org/10.18488/journal.2/2015.5.5/2.5.261.268)]
- 60 Deepalakshmi P, Shankar K. Role and impacts of ant colony optimization in job shop scheduling problems: A detailed analysis. *Evolutionary Computation in Scheduling*, 2020: 11–35.
- 61 蒋华伟, 郭陶, 杨震. 车辆路径问题研究进展. *电子学报*, 2022, 50(2): 480–492. [doi: [10.12263/DZXB.20201154](https://doi.org/10.12263/DZXB.20201154)]
- 62 刘紫玉, 赵丽霞, 薛建越, 等. 面向车辆路径问题的改进蚁群算法研究. *河北科技大学学报*, 2022, 43(1): 80–89. [doi: [10.7535/hbkd.2022yx01009](https://doi.org/10.7535/hbkd.2022yx01009)]
- 63 Miao CW, Chen GZ, Yan CL, *et al.* Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm. *Computers & Industrial Engineering*, 2021, 156: 107230.
- 64 孔维立, 王峰, 周平华, 等. 改进蚁群算法的无人机三维路径规划. *电光与控制*, 1–8. <http://kns.cnki.net/kcms/detail/41.1227.tn.20220517.1822.002.html>. [2022-09-06].
- 65 陈超, 张莉. 基于改进蚁群算法的三维路径规划. *计算机工程与应用*, 2019, 55(20): 192–196. [doi: [10.3778/j.issn.1002-8331.1904-0212](https://doi.org/10.3778/j.issn.1002-8331.1904-0212)]

(校对责编: 孙君艳)