

基于灰狼优化器改进蚁群算法的物流配送路径优化算法

周子程, 梁景泉, 刘秀燕*, 黄毓培

青岛理工大学, 山东 青岛

Email: zzc1048576@163.com, ljingquan@outlook.com, hyp20211029@163.com, *liuxiuyan@qut.edu.cn

收稿日期: 2021年3月20日; 录用日期: 2021年4月14日; 发布日期: 2021年4月21日

摘要

蚁群算法在求解多任务物流配送路径优化中, 存在参数设定缺乏标准及不同情境下参数设定各不相同等问题, 本文提出一种基于灰狼优化器的改进的蚁群算法的路径优化算法。针对传统蚁群算法参数设定不确定等问题, 通过引入灰狼优化器, 借助灰狼优化器进行全局搜索的特点以及渐进式搜索的特性, 找到蚁群的最优参数, 从而自动获取最优参数, 解决蚁群算法的参数配置问题。最后, 将改进后的算法应用于多任务物流配送路径优化中。实验结果表明, 提出的算法能够自动得出更优的参数, 算法具有良好的求解精度和稳健的鲁棒性。

关键词

蚁群算法, 灰狼优化器, 参数设置, 路径优化

Logistics Distribution Route Optimization Algorithm Based on Improved Ant Colony Algorithm of Gray Wolf Optimizer

Zicheng Zhou, Jingquan Liang, Xiuyan Liu*, Yupei Huang

Qingdao University of Technology, Qingdao Shandong

Email: zzc1048576@163.com, ljingquan@outlook.com, hyp20211029@163.com, *liuxiuyan@qut.edu.cn

Received: Mar. 20th, 2021; accepted: Apr. 14th, 2021; published: Apr. 21st, 2021

*通讯作者。

文章引用: 周子程, 梁景泉, 刘秀燕, 黄毓培. 基于灰狼优化器改进蚁群算法的物流配送路径优化算法[J]. 计算机科学与应用, 2021, 11(4): 892-901. DOI: 10.12677/csa.2021.114092

Abstract

In solving multi task logistics distribution route optimization, ant colony algorithm has some problems, such as lack of standard parameter setting and different parameter setting in different situations. This paper proposes an improved ant colony algorithm based on gray wolf optimizer. In view of the uncertain parameter setting of traditional ant colony algorithm, the gray wolf optimizer is introduced to find the optimal parameters of ant colony by virtue of the global search and progressive search characteristics of gray wolf optimizer, so as to automatically obtain the optimal parameters and solve the parameter configuration problem of ant colony algorithm. Finally, the improved algorithm is applied to the multi task logistics distribution path optimization. Experimental results show that the proposed algorithm can automatically get better parameters, and the algorithm has good accuracy and robustness.

Keywords

Ant Colony Algorithm, Gray Wolf Optimizer, Parameter Setting, Path Optimization

Copyright © 2021 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着我国快递行业的快速崛起,人们对快递的需求与日俱增,而此时的快递行业却仍是低效率的配送,没有专业的多任务路径规划系统,很难提升送快递的效率。最近快递行业也曝光了很多问题,比如快递员违规将快递放入快递柜里等,其根本原因都是因为效率低,追求效率的提升所做出的这些行为。这种在快递运输速度的需求和实际快递行业运输效率不足的矛盾日益凸显,从而占用了大量的人力来配送快递,极大的提升了人力成本。此外,由于交通状况和需求的不确定性,人们每天都需要用自己的经验来做出一些判断。但是由于人的主观性以及经验的局限性导致我们的判断经常出错。在如今这个网购普及的时代下,物流配送需求急剧增加,然而由于缺乏专业的配送规划系统而增加了物流的成本,急需一个高效的解决方案来解决。

近年来,针对存在多约束条件和非线性等特点导致车辆路径难以优化的问题,国内外研究学者开展了大量的研究工作。各种智能算法应用于路径规划问题中,如典型的蚁群算法[1],粒子群算法[2],遗传算法[3]等。蚁群算法(Ant Colony System, ACO)的基本模型最早是由意大利学者 Dorigo M. 等于 1991 年提出的,最先被用来求解 TSP 问题,但当时并没有立即带动其他研究者也投身于对此思想的讨论中。直到五年之后, Dorigo 发表文章并在文章内更加详细的描述了蚁群算法的工作原理和数学模型,才引起了更多学者的关注,之后又慢慢涌现出更多相关研究,最终蚁群算法才逐渐走向学术最前沿。在蚁群算法的迅速成长期,在国际上有一些极具突破性的成果有力的推动了算法发展,如 Dorigo 等人提出的带精英策略的蚂蚁系统和 Ant-Q 蚂蚁系统,德国学者 Stutzle 和 Hoos 提出的最大最小蚂蚁系统(MAX-MIN Ant System, MMAS), Cordon 等人提出的最优-最差蚂蚁系统,近期外国也对蚁群算法有最新动态,比如将蚁群算法应用在机器人的路径规划和抗网络攻击方面。国内学者对蚁群算法的研究起步较晚,且因为有国外研究成果作为基础,基础领域的探索性工作偏少,大部分都是在已有研究基础上进行优化改进,提

出修正方案, 或者将其他更先进的思想融入到蚁群算法上。例如李伟等人[4]提出在蚁群算法的参数设置过程中结合遗传算法的交叉变异特性来进行调整。刘玉霞[5]等通过在蚁群算法中引入逆向蚂蚁, 同时结合模拟退火加快了整体收敛速度。文献[6]提出一种基于长短记忆网络和自适应的通信策略的多优化算法, 加速了收敛, 提高了收敛精度, 帮助算法跳出局部最优解。改进的蚁群算法可以在一定程度上弥补自身缺陷, 适应某一些特定领域的问题。文献[7]在蚁群算法的基础上, 为了解决约束满足(CSP)问题而提出了自动蚁群算法(AU-ACO), 该算法会保留优秀的变量, 而不会对所有变量进行赋值, 这样在一定程度上降低了算法的赋值度, 提高了收敛速度, 并且会有更多的机会接近最优解, 极大的提高了算法的性能。在文献[8]中, 就考虑到了蚁群算法的一些缺陷, 对它进行了一定改进后, 应用到船舶路径规划中。蚁群算法的本质是一种遗传算法, 它具有自然界的一些特征, 比如弱肉强食, 弱者淘汰的自然法则。种类算法的搜索代理往往具有一定的随机性, 借助这种性质, 可以让它解决一些传统算法难以解决的问题。如文献[9]中提到的, 比如 TSP 问题、生产调度以及聚类问题等。如果直接搜索或者动态规划计算, 问题的计算量会非常大。蚁群算法则可以省略了很多搜索的路径并且得到一个不是很差的结果。但是蚁群算法在不同的环境模型中的最优参数大不相同, 且使用者在使用的过程中往往要花费大量的精力来调整参数, 而且会很大程度上受到主观性影响。目前, 蚁群算法在多任务路径规划系统方面, 参数的选取在在业界内缺乏明确的标准。

Mirjalili 等人[10]受灰狼群体捕猎行为的启发, 于 2014 年提出一种灰狼优化算法(Grey Wolf Optimization, GWO)算法, 该算法作为一种新兴的群智能元启发式算法, 通过借鉴灰狼的社会组织与狩猎时的行为模式, 使得算法具有很强的搜索能力。目前, 该算法在众多工程领域获得了广发应用, 包括车间调度、数值优化、图像融合以及电力系统配置等。文献[11]中, 借助 A*算法来初始化头狼, 让灰狼优化器能有一个更好的起点。文献[12]在 GWO 的基础上提出了 HGWO 算法, 借助反向学习的方法, 提升初始种群的质量, 增强算法的勘探能力。文献[13]中提出了使用 HGWO 自动适应 VMD 的方法, 提高了原有模型的精度。文献[14]通过引入了正余双弦函数, 来控制其收敛速度, 提高了原有模型的收敛精度, 一定程度上避开了过拟合。文献[15]中, 通过引入非线性惯性权重和交叉变异策略, 强化全局搜索能力和多约束条件搜索能力。文献[16]是基于传统的算法提出的 AGWO 算法, 该算法引入了弹性机制、循环机制和攻击机制, 改进后的算法在局部最优回避和计算精度上明显由于其它启发式算法。在软件方面, 文献[17]提出了基于 GWO 的 ARP-GWO 算法, 对风险进行优先级排序, 使用集成指标和可用性指标, 让软件开发可以节约更多成本, 短时间内交付出更高质量的产品。

针对蚁群算法在物流配送车辆路径优化中的参数难以优化问题, 为提高路径规划问题的收敛速度, 结合不同元启发式算法的全局或局部搜索能力, 本文提出一种基于灰狼优化器的改进蚁群算法, 该算法通过使用灰狼优化器得到一组较优参数, 实现蚁群算法在路径规划问题上摆脱参数依赖, 实现自动取参。

2. 环境建模

在远距离物流运输中, 我们往往可以将两地之间的距离近似的看作成直线距离, 这种估算方法随着两地之间的距离增大而变得更有效果。常用的环境建模的方法有可视图法、格栅法以及拓扑法, 由于拓扑法在使用上简单, 环境建模容易, 并且比较适合用在城市之间的转移上, 因此使用拓扑法对环境进行建模模拟。可以将随机分布的 N 个城市放在二维连续的平面 G 中, 构建无向完全图。如图 1 所示, 用户从 0 地经过 1 地之后抵达 2 地, 那么用户剩下的选择区间为 $[3, N]$, 已经经过的地方将无法转移。

可以通过这些信息, 经过一定大计算得到如图 2 所示大邻接矩阵。这 N 个城市的坐标通过计算二维平面中两点之间的距离来估算城市之间的距离, 从而得到这 N 个城市之间的距离矩阵。

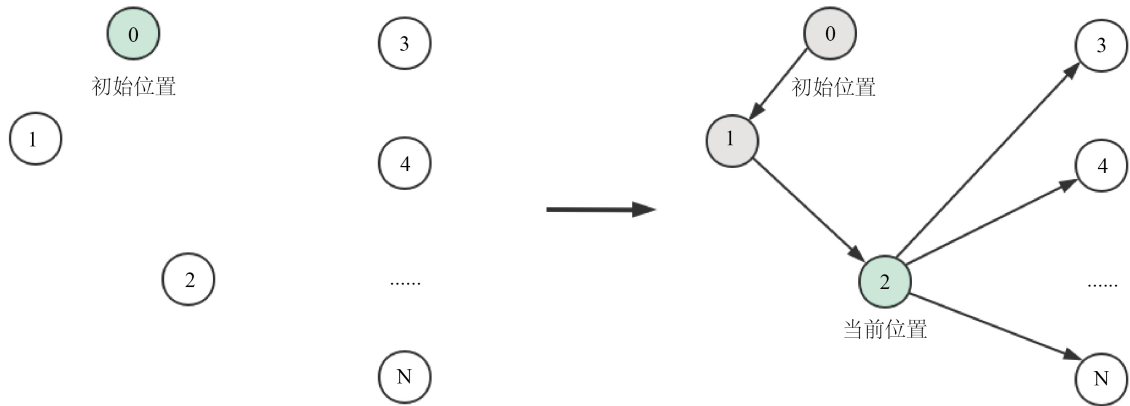


Figure 1. User simulation mobile model
图 1. 用户模拟移动模型

	1	2	3	...	N
1	d_{11}	d_{21}	d_{31}	...	d_{N1}
3	d_{12}	d_{22}	d_{32}	...	d_{N2}
3	d_{13}	d_{23}	d_{33}	...	d_{N3}
\vdots	\vdots	\vdots	\vdots	\ddots	\vdots
N	d_{1N}	d_{2N}	d_{3N}	...	d_{NN}

Figure 2. Distance relationship among N cities
图 2. N 个城市之间的距离关系

其中, d_{ij} 表示从城市 i 到城市 j 的直接距离。在距离集合 d 中, 满足 $\forall i \in N, j \in N, d_{ij} = d_{ji}$, 表示从城市 i 到城市 j 和从城市 j 到城市 i 的距离相等。由此也可以看出, 该矩阵为对称矩阵。将拓扑图构造为邻接矩阵可以在算法设计时, 更贴近计算机语言, 方便后续计算。

3. 相关算法

3.1. 蚁群算法

蚂蚁是一种团结协作的动物, 它们觅食的时候, 如果某一只蚂蚁找到了食物, 它会在路上留下信息素以供同伴寻路。在寻路的过程中, 每只蚂蚁遵循轮盘赌法的移动规则:

$$P_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha(t) * \eta_{ij}^\beta(t)}{\sum_{s \in allowed_k} \tau_{ij}^\alpha(t) * \eta_{ij}^\beta(t)}, & j \in allowed_k \\ 0, & j \notin allowed_k \end{cases} \quad (1)$$

其中 i 是蚂蚁目前所在的城市的编号, j 是蚂蚁能够到达的城市的编号, $\tau(t)$ 是时间 t 时由 i 到 j 的信息素浓度, η 是 i 到 j 的距离的倒数, 表示 i 与 j 两点间的能见度。allowed 是蚂蚁能够到达的位置的集合。为了让蚁群能够更加灵活, 还引入了信息素因子 α 和启发因子 β , 分别表示信息素的重要程度和蚂蚁拒绝探索新区域的可能性。

算法初期, 所有的位置都含有信息素浓度 Q 。每次迭代中, 都有 m 只蚂蚁在搜索。每完成一次迭代后, 本次最优的蚂蚁会留下信息素, 但每个位置也只会保留 $\tau(1-\rho)$ 的信息素, 其运算公式为:

$$\tau_{ij}(t+1) = \tau_{ij}(t) * (1 - \rho) + \Delta\tau_{ij} \quad (2)$$

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}(k) \quad (3)$$

其中， ρ 是信息素挥发因子。这个过程在最大迭代次数 t 之后停止。

蚁群算法流程图如图 3 所示，算法先获得用户的输入的数据，根据用户参数初始化蚂蚁和城市的位置以及蚁群算法的基本参数。完成初始操作后，蚂蚁开始逐个活动，将当前的位置添加到禁忌表中，再选择下一处位置，直到这只蚂蚁完成搜索，下一只蚂蚁再重复进行。当所有蚂蚁都完成搜索后，完成本次迭代，更新信息素。重复此过程直到蚂蚁完成最后一次迭代，输出最后一次迭代最优解：

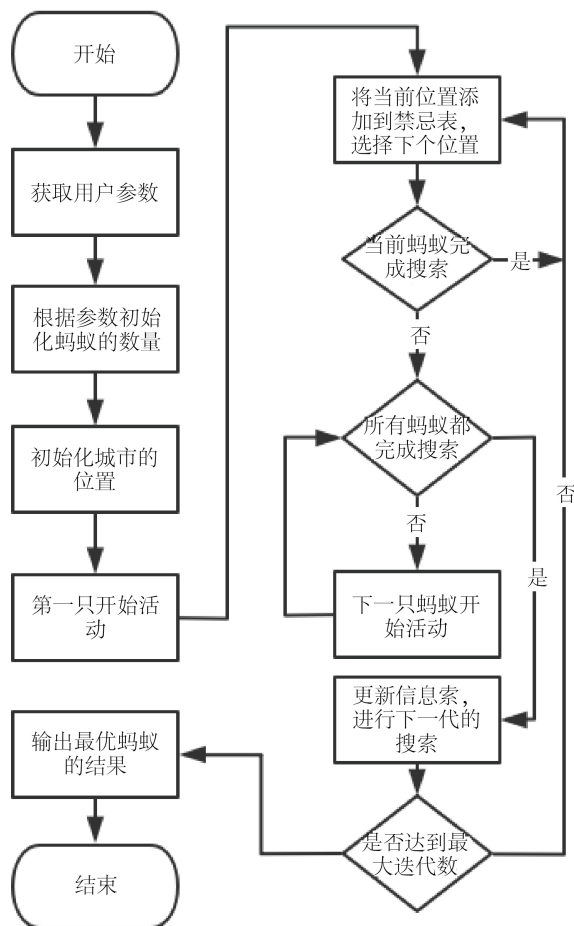


Figure 3. Flow chart of ant colony algorithm

图 3. 蚁群算法流程图

3.2. 灰狼算法

灰狼优化算法也是一种遗传算法。由于其具有的全局搜索能力以及良好的局部处理能力，使它能够在众多的优化器中脱颖而出。这个算法执行的过程类似于灰狼狩猎的过程。在狼群中，灰狼之间有着严格的等级制度。最高等级的三匹灰狼分别是 α 狼、 β 狼以及 δ 狼，其它的灰狼统称为 ω 狼。前三匹头狼是根据对猎物的感知能力进行设定，结果最好的三头狼为头狼。当它们包围猎物时，满足行为公式：

$$\bar{D} = \left| \bar{C} * \bar{X}_p(t) - \bar{X}(t) \right| \quad (3)$$

$$\bar{X}(t+1) = \bar{X}_p(t) - \bar{A} * \bar{D} \quad (4)$$

公式(3)中, 向量 \bar{D} 是头狼 X_p 领导下的位置转移函数, 其中的 p 可以被替换为 α 、 β 或者 δ 。将公式(3)的函数代入到公式(4)中, 即可完成灰狼的迭代。 X 是灰狼当前的位置, 受迭代的向量系数 A 的影响, $\bar{X}(t+1)$ 会随着迭代次数增加而趋近于 $\bar{X}(t)$ 。公式(3)、(4)中的 A 、 D 满足下式:

$$\bar{A} = 2\bar{a} * \bar{r}_1 - \bar{a} \quad (5)$$

$$\bar{C} = 2 * \bar{r}_2 \quad (6)$$

$$\bar{D}_\alpha = \left| \bar{C}_1 * \bar{X}_\alpha - \bar{X} \right|, \bar{D}_\beta = \left| \bar{C}_1 * \bar{X}_\beta - \bar{X} \right|, \bar{D}_\delta = \left| \bar{C}_1 * \bar{X}_\delta - \bar{X} \right| \quad (7)$$

公式(5)中, \bar{a} 是由2到1线性递减的变量, \bar{r}_1 和 \bar{r}_2 都是0到1的随机变量。使用这两个随机变量来生成 \bar{A} 变量和变量 \bar{C} , \bar{C} 作为公式(7)的参数, 是公式(3)的具体化, \bar{A} 是迭代公式(4)中的一个系数。虽然 ω 狼处于最低位置, 但是它们却是狼群中不可或缺的一部分。最高等级的前三匹狼通过下述公式指导着 ω 狼们的行动:

$$\bar{X}(t+1) = \frac{\bar{X}_1 + \bar{X}_2 + \bar{X}_3}{3} \quad (8)$$

公式(8)是公式(4)的具体化。每次迭代后, 新一代的灰狼 ω 会根据上一代最优秀的三匹狼调整它们狩猎的位置。经过不断地迭代更新狼群的位置, 最终会逼近猎物, 达到猎物的位置。

4. 基于灰狼优化器改进的蚁群算法

在高维度的参数空间中, 每一匹灰狼都携带着一组蚁群算法的参数, 它们将会把携带的参数传给蚁群算法中, 并让蚁群算法在这组参数下运行。经过算法对蚁群算法进行一定的评价之后, 再将最优的三组参数进行杂交。在初代中, 由于最开始的灰狼可能是凭运气而存活下来的, 随着算法进行, 再存留下来的灰狼就最有可能是凭借它出色的求解能力存活下来。所以灰狼算法会随着时间的推移, 父本杂交的效果会逐渐提升。但是传统的灰狼优化器是通过线性函数提升杂交效果, 一定程度上可能发生过拟合现象, 且蚁群算法的结果本身就具有随机性。因此对参数 a 进行了适应性调整:

$$a = 2 \sin \left(\frac{\pi}{2} \left(1 - \frac{t}{\text{Max_iter}} \right) \right) \quad (9)$$

其中, t 为当前迭代数, Max_iter 为最大迭代数。其它的算法及参数不做改动。

蚁群算法中, 我们关心的是该算法能否得到最优解。在得到的解不同的时候, 算法会优先选择得到的结果最优的参数组合。携带这组参数组合的灰狼在迭代中最有可能活下去。而当不同的参数得到最优解相同的时候, 会优先选择执行效率最高的参数组合。

$$\text{fitness} = \begin{cases} \min(\text{consuming}); \text{路径}\square\text{相等} \\ \min(\text{route}); \text{others} \end{cases} \quad (10)$$

在考虑运行效率时, 会优先考虑该参数组合下的时间效率。比如能在最少的迭代次数内完成任务。因为在最短的迭代次数内完成搜索, 可以有更多的机会探索新的可能的最优解。因此, 在得到最优迭代数最小的参数组合下, 算法的开发潜力最大。蚁群算法的蚂蚁数是一个很矛盾的参数, 如果该参数过大, 会导致算法复杂度过高, 而参数过小会使算法难以收敛, 因此提出如下判断标准:

$$m(t+1) = \begin{cases} \text{Min}(m(t)), & \text{所有灰狼结果相等} \\ \text{Mincons}(m(t)), & \text{出现更快收敛解} \\ \text{fitness}(m(t)), & \text{出现新的最优解} \end{cases} \quad (11)$$

虽然 m 属于灰狼优化器要优化的一部分，但是要单独提出来。传统的灰狼优化器只会一味的寻求最优解，但是不会考虑这个解的代价，因此引入公式(11)来控制算法的代价。

基于灰狼算法对蚁群算法进行优化的算法如图 4 所示。该算法主要分为两个部分，一个是用于求解问题的蚁群算法，另一个是用于优化蚁群算法参数的灰狼算法。算法开始时，先对算法进行初始化，在地图中加入 N 个城市，并随机分布它们的位置；对这 N 个城市进行距离的计算，得出这 N 个城市的邻接矩阵，并构建出 N 个顶点的无向完全图；初始化灰狼，让狼群均匀地分布在蚁群算法的张量空间中；每只灰狼再根据自己携带的参数运行蚁群算法，再根据灰狼的得分进行优化，最终得到最优的参数组合。

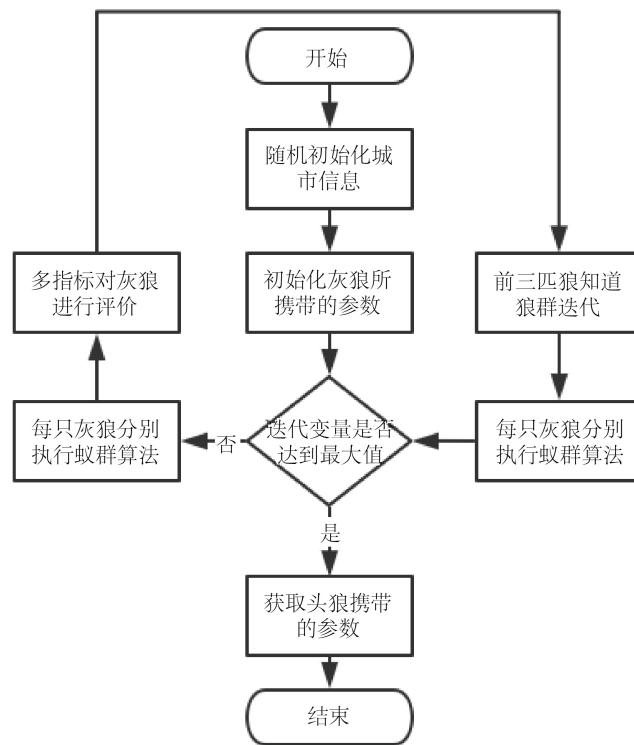


Figure 4. Algorithm flow chart
图 4. 算法流程图

改进算法的流程图如图 4 所示，算法详细步骤如下：

Step1: 随机初始化每个城市的位置，计算出每个城市之间的距离，并存入到邻接矩阵中，方便后续的计算。

Step2: 初始化灰狼，让狼群均匀分布在解的空间内，设置预定的最大迭代数。

Step3: 使用每只灰狼携带的参数分别运行蚁群算法，运行时要使用在 Step(1)中预设的城市数据。

Step4: 分析每只灰狼的运行结果，让结果最优，收敛最快的灰狼成为首领，指导狼群更新位置，继续搜索猎物。

Step5: 当狼群达到预定的迭代数后停止继续迭代，获取头狼的坐标信息，从而得到最优的参数信息。

5. 实验结果

为验证该算法的有效性，评估得到的参数的实用性，使用 MATLAB 进行仿真实验。该算法的具体参数设置为：模拟城市数为 20，灰狼数量 $s = 5$ ，最大迭代数 $T = 100$ 。蚁群算法的参数张量空间的四个维度分别是蚂蚁数 m ，信息因子 α ，启发因子 β 以及信息素挥发因子 ρ ；它们的下界为 $[0.6*n, 1, 4, 0.5]$ ，它们的上界为 $[0.9*n, 2, 6, 0.8]$ 。经过灰狼优化器优化的参数运行蚁群算法结果如图 5 所示，经过灰狼优化器优化后的蚁群算法参数有着明显的改善效果。

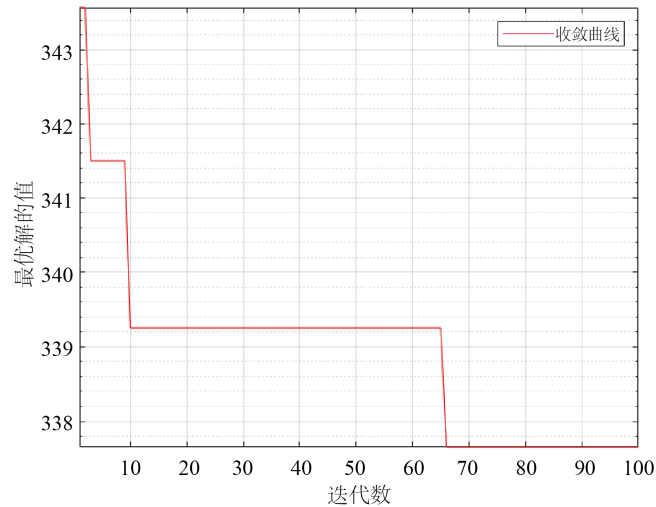


Figure 5. The results of the improved algorithm
图 5. 改进算法运行的结果

未经过参数优化的蚁群算法和经过参数优化后的蚁群算法得到的城市结果如图 6 所示，图 6(a)图中未经参数优化过的蚁群算法很明显存在问题，在处理一些比较靠近的节点的时候容易出现一些问题，虽然问题不大，但是却在一定程度上影响到了算法的性能；而图 6(b)图中的参数优化后的结果，不存在很明显的肉眼可见问题。未经优化时的蚁群算法在参数设计上存在很明显的的问题，有待优化。

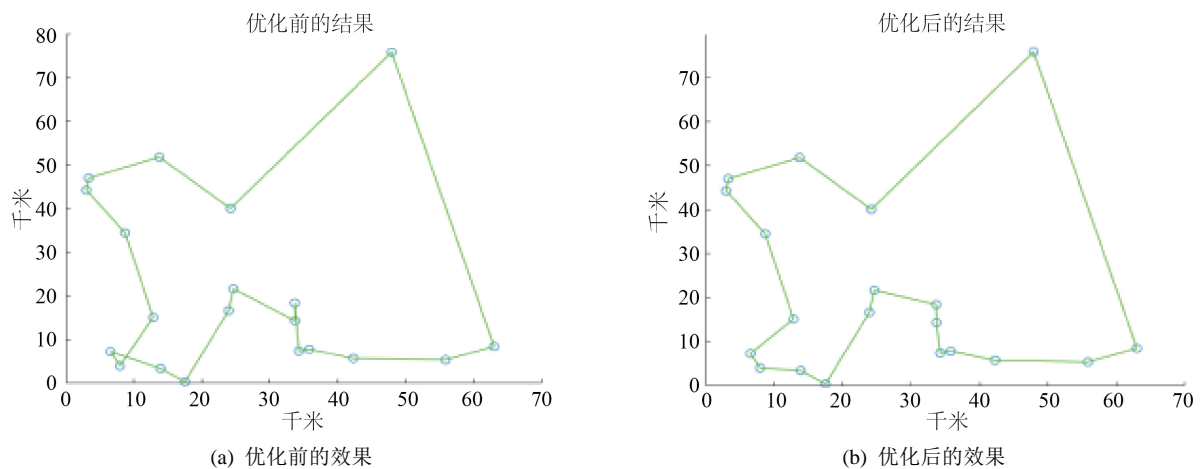


Figure 6. Operation results before and after optimization
图 6. 优化前后的运行结果

通过使用优化后的参数和优化前的参数对比搜索后的结果如图 7 所示，蓝实线代表优化前的收敛曲线，红实线代表优化后的收敛曲线，通过收敛曲线的比较可以更直观地表现出优化后的效果。可以明显看出优化后的初始结果要优于优化前的初始结果。此外优化后的算法还有着收敛速度快，收敛结果更好等优势特点。

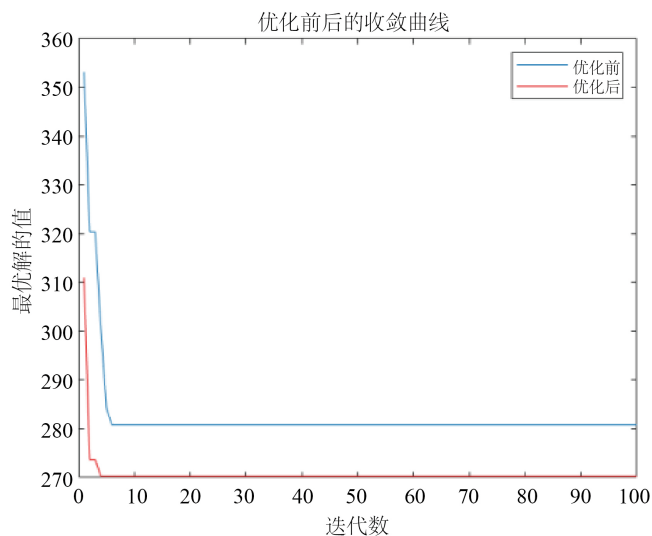


Figure 7. Comparison of convergence curves

图 7. 优化前后收敛曲线对比

重复上述实验五次，得到如表一所示的实验结果如表 1 所示，根据表格结果可以看出，经过灰狼算法优化后的参数具有很强的适应性，在大约 80% 的情况下能够优于随机选择参数的蚁群算法。在少数情况下，随机选择的参数和优化后的参数，有着相同的运行结果。但是在大多数随机运行结果较好的情况下，其参数是一个比较优的参数，而在大约 20% 的情况下会和优化后的参数相差无几，这同样可以验证优化后的运行结果是一个很好的结果。通过优化结果可以看出，优化前的结果比优化后的结果整体上劣势了大约 3.9%。所以可以观察出，优化后对原结果有一定程度上的改善。

Table 1. Results of five experiments

表 1. 五次实验得到的结果

实验次数	优化前	优化后	优化效果
实验 1	284	278	6
实验 2	273	273	0
实验 3	276	268	8
实验 4	298	276	22
实验 5	261	245	16
平均值	278.4	268	10.4

6. 结语

本文针对多任务物流配送路径优化问题，首先建立数学模型，针对蚁群算法中选择参数困难等问题，选取灰狼优化器来实现对蚁群算法参数的优化。其次，为了适应元启发式算法需要更长的收敛时间来稳

定数学期望, 故在灰狼优化器中, 引入正弦函数修正控制收敛的变量 a , 使其可以更加广泛地搜索。蚁群算法是一种元启发式算法, 在路径规划问题上具有很强的计算能力, 但是及其依赖参数的设置。而灰狼优化器本身也具有很强的全局搜索能力, 可以适应很多复杂的搜索地形, 可以用来搜索蚁群算法的参数设置。最后, 通过仿真实验与对比实验的结果, 证明该算法可以有效的优化物流配送车辆配送路径, 降低了配送成本, 提高了配送效率。但是该算法过度依赖路程长度, 而在实际工程中往往需要考虑各种问题, 未来将向引入道路复杂因子进行更加细致的研究。

基金项目

本文受山东省自然科学基金项目 (ZR2020QF008) 和山东省大学生创新创业训练计划项目 (S202010429208, S202010429018) 资助。

参考文献

- [1] 李锋源, 许艳萍, 王武. 多策略蚁群算法求解机器人路径规划[J]. 福州大学学报(自然科学版), 2011, 39(3): 385-391.
- [2] 刘关俊. 基于粒子群算法的移动机器人路径规划研究[D]: [硕士学位论文]. 长沙: 中南大学, 2007.
- [3] 陈刚, 沈林成. 复杂环境下路径规划问题的遗传路径规划方法[J]. 机器人, 2001, 23(1): 40-44.
- [4] 张江霄. 基于混合扫描法的优化蚁群算法的物流配送系统研究与应用[Z]. 河北省, 邢台学院, 2019-07-18.
- [5] 刘玉霞, 王萍, 修春波. 基于模拟退火策略的逆向蚁群算法[J]. 微计算机信息, 2006, 22(34): 265-267.
- [6] Li, S., You, X. and Liu, S. (2021) Multiple Ant Colony Optimization Using Both Novel LSTM Network and Adaptive Tanimoto Communication Strategy. *Applied Intelligence*, No. 9, 1-21. <https://doi.org/10.1007/s10489-020-02099-z>
- [7] Guan, B., Zhao, Y. and Li, Y. (2021) An Improved Ant Colony Optimization with an Automatic Updating Mechanism for Constraint Satisfaction Problems. *Expert Systems with Applications*, **164**, Article ID: 114021. <https://doi.org/10.1016/j.eswa.2020.114021>
- [8] 龚铭凡, 徐海祥, 冯辉, 薛学华. 基于改进蚁群算法的智能船舶路径规划[J]. 武汉理工大学学报(交通科学与工程版), 2020, 44(6): 130-134.
- [9] 肖艳秋, 焦建强, 乔东平, 等. 蚁群算法的基本原理及应用综述[J]. 轻工科技, 2018(3): 69-72.
- [10] Mirjalili, S., Mirjalili, S.M. and Lewis, A. (2014) Grey Wolf Optimizer. *Advances in Engineering Software*, **69**, 46-61. <https://doi.org/10.1016/j.advengsoft.2013.12.007>
- [11] 曹建秋, 张广言, 徐鹏. A*初始化的变异灰狼优化的无人机路径规划[J/OL]. 计算机工程与应用, 2021: 1-12. <http://kns.cnki.net/kcms/detail/11.2127.TP.20201225.0932.014.html>, 2021-04-08.
- [12] 王永琦, 江潇潇. 基于混合灰狼算法的机器人路径规划[J]. 计算机工程与科学, 2020, 42(7): 1294-1301.
- [13] Gai, J., Shen, J., Hu, Y., *et al.* (2020) An Integrated Method Based on Hybrid Grey Wolf Optimizer Improved Variational Mode Decomposition and Deep Neural Network for Fault Diagnosis of Rolling Bearing. *Measurement*, **162**, Article ID: 107901. <https://doi.org/10.1016/j.measurement.2020.107901>
- [14] 石春花, 刘环. 基于正余双弦自适应灰狼优化算法的医药物流配送路径规划[J]. 数学的实践与认识, 2020, 50(14): 114-127.
- [15] 袁光辉. 基于改进灰狼优化算法的物流配送路径规划[J]. 荆楚理工学院学报, 2019(3): 12-19.
- [16] Meng, X., Jiang, J. and Wang, H. (2021) AGWO: Advanced GWO in Multi-Layer Perception Optimization. *Expert Systems with Applications*, Article ID: 114676. <https://doi.org/10.1016/j.eswa.2021.114676>
- [17] Prakash, B. and Viswanathan, V. (2021) ARP-GWO: An Efficient Approach for Prioritization of Risks in Agile Software Development. *Soft Computing*, **25**, 5587-5605. <https://doi.org/10.1007/s00500-020-05555-7>