

改进蚁群算法在合肥市包河区果蔬配送中的应用研究

邓灵斌, 邵 军^①

(南华大学 经济管理学院, 湖南 衡阳 421001)

[摘要] 路径优化是运输过程中必须考虑的问题。合理的路径既节省时间成本, 又提高工作效率。文章引入蚁群算法和改进蚁群算法, 利用算法的优越性, 解决合肥市包河区处于快速发展状态下居民对果蔬需求的现实性, 进而需要研究果蔬配送过程的路径问题, 基于找出最优路径解, 运用蚁群算法与改进蚁群算法解决果蔬配送, 并对两种算法求出的解进行对比分析, 找出最短路径, 结果说明解决包河区果蔬配送路径的必要性, 也说明算法在新环境中的适应性。

[关键词] 路径问题; 改进蚁群算法; 果蔬配送

[中图分类号] F252.3 **[文献标识码]** A **[文章编号]** 1673-0755(2015)06-0023-06

蚂蚁在觅食过程中选择的路径是规律性的, 彼此间信息转换, 使得觅食工作高效率。人们通过蚁群的这一活动, 结合现有技术模拟出蚁群行走路径, 并成功的应用到实际问题中。

一 蚁群算法的原理

1991年, 布鲁塞尔自由大学的 Marco Dorigo 等

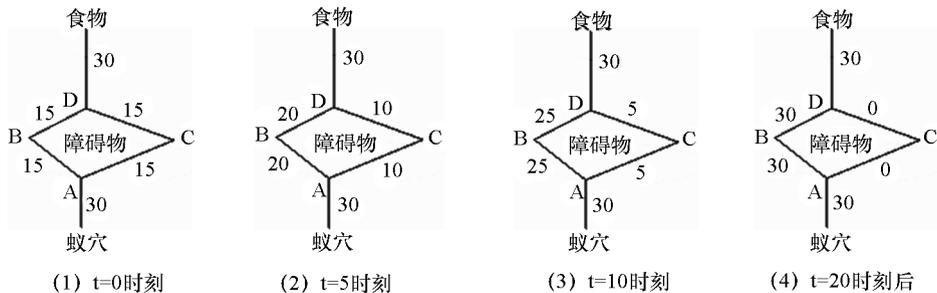


图1 蚁群觅食过程

图1表示有 $n = 30$ 只蚂蚁在洞穴处开始觅食, 蚁穴和食物之间存在障碍物, 30只蚂蚁有两条路径到达食物地点, 分别为“蚁穴—A—B—D—食物”和“蚁穴—A—C—D—食物”, 假设在完成工作的整个过程中两条路径上的蚂蚁总数是不变的。其中图1(1)表示 $t = 0$ 时刻蚂蚁在两条路径上的数量关系, 开始时刻两条路径上的蚂蚁数量都是 $n = 15$; 图1(2)表示 $t = 5$ 时刻路径“蚁穴—A—B—D—食物”上的蚂蚁数量开始由初始的 $n = 15$ 只蚂蚁变成该时

刻的 $n = 20$, 而路径“蚁穴—A—C—D—食物”由初始的 $n = 15$ 减少为 $n = 10$ 只; 图1(3)表示 $t = 10$ 时刻路径“蚁穴—A—C—D—食物”上的蚂蚁数量减少为 $n = 5$, 路径“蚁穴—A—B—D—食物”上的蚂蚁数量增加到 $n = 25$; 当30只蚂蚁工作一段时间后, 彼此间通过一种信息转换开始逐渐找到觅食的最短路径, 使得搬运时间缩短, 工作效率提高, 图1(4)表示在 $t = 20$ 后, $n = 30$ 只蚂蚁通过一段时间找到觅食搬运的最短路径。科学家们通过观察蚁群的觅食活

人^[1]根据蚂蚁搜索食物的这一特征, 结合著名的旅行商问题用软件模拟实施了一个以“蚂蚁”为基础的系统——蚁群算法。

下面通过蚂蚁寻食更好地诠释蚁群算法的基本原理, 见图1。

[收稿日期] 2015-09-15

[基金项目] 湖南省重点学科项目“管理科学与工程”资助

[作者简介] 邓灵斌(1973-), 男, 湖南祁东人, 南华大学经济管理学院副教授, 博士。

^①上海铁路局合肥货运中心助理工程师。

动,结合计算机技术研究出蚁群算法,并很快在其它领域成功应用,但在人类的日常生活中,可能某条路径是最短的,都集中选择在该条路径进行活动,同时,也要考虑复杂情况下最短路径是不是最佳的选择,这就要求必须结合实际问题来验证算法的精确性和有效性。

二 蚁群算法的模型

(一) 蚁群算法的步骤

蚁群算法的步骤^[2]如表1所示。

表1 蚁群算法的步骤

步骤		
Step1	初始时刻蚁群行走路径的产生	根据蚁群行走产生的多条从起点到终点的可行路径,每一条路径代表一只蚂蚁的爬行轨迹。
Step2	对信息素浓度进行调整	对已产生的每一条可行路径的长度进行计算,以及对每一条可行路径上相对应的信息素的增量进行计量,再根据公式对每一条可行路径上的信息素进行更新调整。
Step3	对形成的每一条可行路径进行一定的调整	将其中某条蚂蚁所走的弯曲路径按直线拉直为一条可行路径。如果本次可行路径比之前已经记录的最短路径的长度还小,则可以对本可行路径上的信息素浓度按照 Step2 进行调整更新。如果本次可行路径已经达到当前预先设定的目标值,则直接转至 Step5。
Step4	下一时刻蚂蚁可行路径的产生	该时刻产生的由起点到终点的可行路径,将按照公式进行信息素的转移概率计算,并转至 Step2。
Step5	结束	将当前可行路径作为最短路径输出。

(二) 蚁群算法的模型

将 n 只蚂蚁随机放在模拟仓库中的 m 个位置上,每只蚂蚁都具有记忆性,它们都能依据彼此之间留下的信息素浓度来确定下一步相应的转移概率来选择路径,并且蚂蚁能够将自己已经走过的路径存放于自己的禁忌表中,这些信息能够决定蚂蚁此次不会重复选择同一条路径,在完成所有路径后,蚂蚁可以根据每个所走路径的长短来找到最短路径,并重新释放新的信息素来转移搜索的行动路径。

首先,对蚁群的行为进行模拟所引用的符号进行描述。其中 N 表示蚁群中蚂蚁的个数; $d_{ij}(i=1, 2, \dots, n_1; j=1, 2, \dots, n_2; n_1$ 和 n_2 分别是平面化仓库的工作环境二维划分维数)为自动化仓库平面的工作环境位置点 i 与 j 的距离; $b_i(t)$ 为 t 时刻位于位置点 i 处的蚂蚁数目; $\tau_{ij}(t)$ 表示 t 时刻在路径 $\langle i, j \rangle$ 上残留的信息素轨迹的量,即信息素浓度。其中有

$$N = \sum_{i=1}^n b_i(t)。$$

其次,初始时刻对每一条自动化仓库选择的路径

上的信息素是相等的,则有 $\tau_{ij}(0) = C, C$ 为常数。

最后,在蚂蚁进行觅食过程中,蚁群中的个体会根据每条路径上的信息素来进行下一次路径的选择。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^[3]公式(1):

$$P_{ij}^k(t) = \begin{cases} \frac{(\tau_{ij}^\alpha(t)\eta_{ij}^\beta(t))}{\sum_{j \in \text{Allowed}_i^k; r \in \text{Allowed}_i^k} (\tau_{ir}^\alpha(t)\eta_{ij}^\beta(t))}, & \\ 0, & \text{否则} \end{cases} \quad (1)$$

在蚁群算法中,每一个位置点运动到另一个位置点,这个过程是逐步搜索的。在搜索过程中,每只蚂蚁 k 会在路径选择过程中生成一个 F 步的路径,循环次数为 t ,且 $1 \leq t \leq t_{\max}, t_{\max}$ 是预先设定的最大迭代次数。其中, t 表示蚂蚁算法的循环次数,每进行一次循环,路径上的信息素都会发生变化; $P_{ij}^k(t)$ 指蚂蚁 k 在第 t 次循环时选择路径位置点 i 到 j 的相应转移概率; $\tau_{ij}(t)$ 表示 t 时刻在路径 $\langle i, j \rangle$ 上残留的信息素浓度; $\eta_{ij}(t)$ 表示 t 时刻在路径之 $\langle i, j \rangle$ 间的可见局部启发式函数;参数 α 和 β 表示可以根据不同要求和目标,调整参数来控制信息素浓度和局部信息的影响权重。当 $\alpha = 0$ 时,信息素不起作用,只考虑能见度的局部启发式函数;当 $\beta = 0$ 时,只有信息素起作用,这时整个转移概率会快速收敛,会得到一个较差的转移概率。由于蚁群具有良好的记忆性,整个解集合会在整体路径选择过程中不断进行调整。

(三) 蚁群算法在车辆运输路径应用中的优点

路径问题是一种组合优化问题,它在整个运输过程的初始阶段和蚂蚁觅食具有相似性,具有蚁群算法的优点,具体优点如下^[6]。

- 1、蚁群算法具有系统性。
- 2、蚁群算法可以分布式计算。
- 3、蚁群算法具有自组织性。
- 4、蚁群算法具有正反馈性。
- 5、蚁群算法具有较强的鲁棒性。

三 改进的蚁群算法

现实生活中,容易忽略细节。在蚂蚁觅食过程中,没有充分考虑蚁群的搜索时间过长以及所有个体在搜索到一定程度上都倾向于某一局部最优解,很难进行下一步的搜索,使得更好的解难以发现,这时就要在已有模型中加入相关系数使得算法更好地适应新的环境,从而缩短觅食路程,即搜索过程的速度加快,也使得搜索的精度更加准确。

(一) 相关系数

在实际问题中存在各种不同程度的相关因素的

影响,这时就不能考虑单一的因素,蚁群算法具有全局性特征,通常会忽略局部环境中的相关因素的影响,这也就是在长时间路径选择中需要考虑的,随着路径选择的次数增多,路径上的信息素浓度会逐渐挥发掉,此时在蚁群算法中我们要考虑局部信息素浓度挥发程度的快慢,用挥发系数 ρ 来控制信息素浓度的挥发程度, $0 \leq \rho \leq 1$ 。

在经过 h 个时刻后,蚁群会完成路径的一个循环选择,这时残留在每条路径上的信息素浓度^[4]为

$$\tau_{ij}(t+h) = (1-\rho)\tau_{ij}(t) + \sum_{k=1}^N \Delta\tau_{ij}^k \quad \dots\dots(2)$$

其中,可以用 $\tau_{ij}(t+h)$ 对全局的信息素浓度进行调整。 ρ 表示某条路径上的挥发度, $(1-\rho)$ 表示某条路径上的剩余度; $\Delta\tau_{ij}^k$ 表示蚂蚁 k 在本次循环中

残留在路径 $\langle i, j \rangle$ 上的信息素浓度,它在局部调整准则中可定义^[5]为:

$$\Delta\tau_{ij}^k = \begin{cases} Q/L_k, & ; \\ 0, & \text{否则} \end{cases} \quad \dots\dots(3)$$

其中, Q 表示信息素浓度强度,它在一定程度上影响蚁群算法的收敛快慢,在运用中通常为常数; L_k 表示蚂蚁 k 在本次循环中走过的所有路径的总长度。

(二)改进的蚁群算法的收敛性分析

基于蚁群算法的模型加入局部信息挥发度 ρ 对蚁群算法进行收敛性分析,在示例中取蚂蚁的数量为 $N=20$,临界时刻 $m=20$,局部信息素挥发度 $\rho=0.3$ 。给定一个有大量不规则障碍物的环境位置图,大小为 800×400 像素,起点和终点分别靠近位置图的左上角和右下角^[6]。

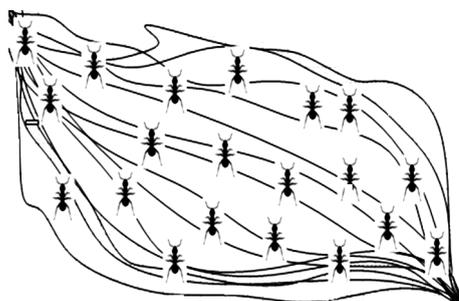


图2 在第20个时刻

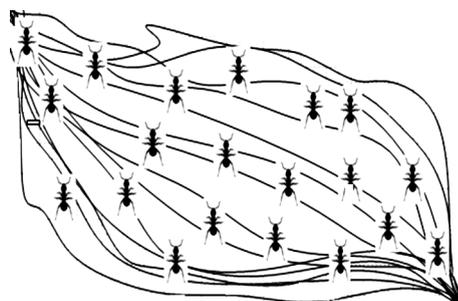


图3 在第60个时刻聚集的蚁群路径移动曲线
聚集的蚁群路径移动曲线

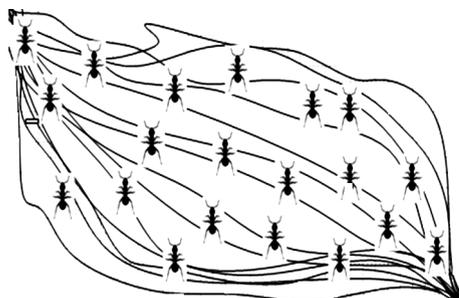


图4 在第120个时刻

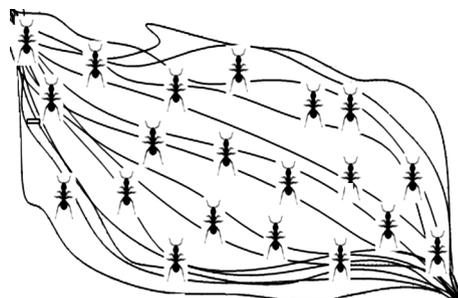


图5 最短路径移动长度蚁群最终的收敛曲线

图2中表示20只蚂蚁在第20个时刻聚集的路径移动曲线,根据图中可以明显发现大多数蚂蚁会选择靠近起点和终点连线的一个限定区域内,这使得该区域的信息素浓度高于周边其它区域。图3是蚁群在第60个时刻聚集的路径移动曲线,与图2相比,可以发现随着时间的推移越来越多的蚂蚁转移路径,逐渐选择靠近起点和终点连线的限定区域内,这会导致该区域的信息素浓度慢慢高于周边其它区域,从而会有图4在第120个时刻蚁群最终收敛的转移路径。图中灰色表示蚁群经过路径上所留下的信息素浓度,实体线表示一条已被修正过的最短路径移动曲线。图5表示最短路径移动长度的收敛曲

线。从图5可以看出,通过使用蚁群算法设计的模型,在第70个时刻蚁群所走的路径就能得到收敛的路径解,而第70~120个时刻就没有得到更好的路径解,这表明蚁群算法的收敛程度趋于平稳。

四 改进蚁群算法在包河区果蔬配送中的应用

在实际车辆运输问题中,通常会忽略局部性要素的影响,如轻装轻卸、防冻防热等因素。这时考虑在复杂情况下这些因素的存在性是必要的,本节对合肥市包河区果蔬配送过程中存在的问题进行相关分析,提出运用蚁群算法和改进蚁群算法解决车辆运输路径问题,仿真模拟并对输出结果进行分析,验

证改进蚁群算法在解决合肥市包河区果蔬配送车辆路径问题中的有效性和优势,同时也证明了算法的可行性。

(一)合肥市包河区简介

包河区是合肥市四大城区之一,位于安徽中部,长江淮河之间,巢湖之滨,素有面积和人口并称的“第一大区”,是安徽省委、省政府所在地。它集聚了商业、旅游、30 多所高校和科研院所。区域面积 340 平方公里,其中巢湖水面面积 70 平方公里。行政区域如表 2 所示。

表 2 包河区行政区划

区域划分	
7 个街道	常青街道、芜湖路街道、包公街道、骆岗街道、望湖街道、城街道、烟墩街道
2 个镇	淝河镇、大圩镇
1 个省级工业园区	包河工业区
1 个街道级社区	滨湖世纪社区,未来滨湖新区将建 8 个大社区

(二)包河区果蔬配送车辆运输过程中存在的问题

随着包河区经济的发展,城区建筑耸起,道路错综复杂,截止 2014 年底,包河区常住人口有 81.77 多万人。其中居民对果蔬的需求量逐年增加,而且对果蔬的质量要求也越来越高,这不仅仅要求果蔬本身的质量,也对果蔬的运输过程提出了更高要求,其中车辆的运输路径问题是极其重要的,在保证居民能够快速吃到新鲜的果蔬时,也出现了不少的干扰因子,如何能够更快使得居民吃到新鲜的果蔬就要考虑在运输路径上进行优化,合理的分配,快速的响应是提供效率的最好方法。

由于果蔬之间存在着差别,在保持果蔬新鲜的同时,也要针对不同果蔬自身的不同考虑不同的运输方式,包河区果蔬运输存在以下问题:

- 1、运输工具的选择,不同果蔬对车辆的要求也不一样,由于城区一些路段会禁止大型车辆的通行,大型车辆就不得不绕道而行,这就加大了运输路径的长度,增加运输成本。
- 2、客户预约配送的时间因子,即优先级的问题,对于客户事先定好时间段配送时,要充分考虑路径规划的问题,提高配送效率。
- 3、顾客需求量的变化,顾客需求量的增减会直接影响车辆选择配送路径。
- 4、容量限制,对于限制量的问题,要事先对几个配送目标进行路径规划。
- 5、总长度限制,即果蔬的保鲜时间的约束,有些果蔬要求在有限的时间配送到需求点,才能保证果蔬的新鲜度,从而可以保证销售量。
- 6、果蔬运输技术要求,为了保持果蔬的新鲜和色泽度,在运输途中必须加强有效管理。由于果蔬的这一特点,对果蔬运输过程的要求为快装快运、轻装轻卸、防热防冻。

(三)改进蚁群算法在包河区果蔬配送中的优化应用

1、基于蚁群算法的车辆最短路径收敛性分析

图 6-8 表示车辆 4、5、6 到分别到 A、B、C 在迭代次数 = 50 次情况下基于蚁群算法车辆路径的最短路径长度和平均路径长度,通过两条路径的收敛曲线得到最优路径。

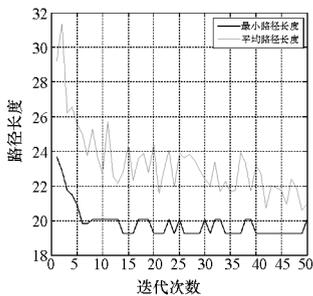


图 6 4 号车辆最短路径的蚂蚁爬行图

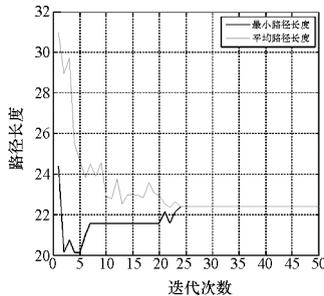


图 7 5 号车辆最短路径的蚂蚁爬行图

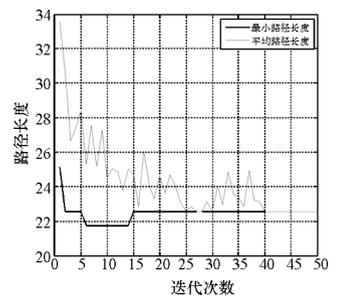


图 8 6 号车辆最短路径的蚂蚁爬行图

2、基于改进蚁群算法的车辆最短路径收敛性分析

图 9-11 表示 7、8、9 号车辆分别到 A、B、C 在迭代次数 = 50 次情况下车辆路径的最短路径长度和平均路径长度,通过两条路径的收敛曲线得到最优路径。

3、改进蚁群算法在包河区果蔬配送车辆路径研究

合肥市包河区某果蔬配送中心向城市的三个派送点配送果蔬,A 为孙岗,B 为周冲,C 为吴岗,图中黑色为建筑物,图中左上角表示合肥市包河区某果蔬配送点。已知:车辆总数为 9,每辆的载重量和功能相同,每

个派送点都有 3 辆车配送果蔬,每个派送点的果蔬需求量为 D 为 10 吨,每辆车 $k(k=1,2,3)$ 的载重量 t 为 3

吨,每派送点有 3 辆车都从出发点向 3 个派送点(A、B、C)分发果蔬,任务结束后要返回出发点。

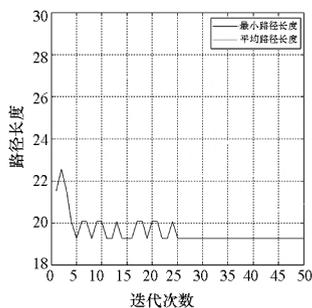


图 9 7 号车辆最短路径的蚂蚁爬行图

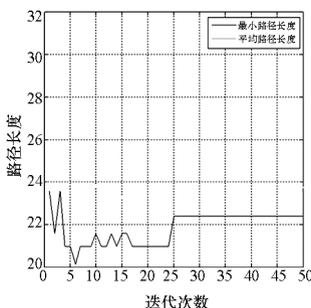


图 10 8 号车辆最短路径的蚂蚁爬行图

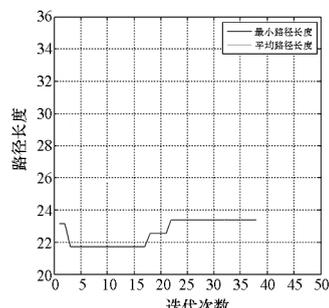


图 11 9 号车辆最短路径的蚂蚁爬行图

前提:每个派送点客户需求的总量不能超过每辆车的载重量,某些约束条件被满足。

约束条件:(1)容量限制,它是指每个配送点 n_i 都有一定的需求量 $d_i \leq D$ 。(2)总长限制,任何路径或者时间都不能超过某一常数。

分析:通过分析各种因素的影响,并问题的分析,建立数学模型,求解。

结果和目的:得到最满意解。 m 个客户对其需求都是满意的,并且使得在整个过程中总运输成本

最小。

通过 MATLAB 进行 $20 * 20$ 的场景模拟研究,迭代次数 = 50。其中算法为蚁群算法,车辆为蚂蚁。图 12 分别表示 1、2、3 号车辆全部爬向目标 A、目标 B 和目标 C 的全部蚂蚁爬过的轨迹。图 13 表示通过迭代后最终找出的路径图,分别为 1、2、3 号车辆全部爬向目标 A、目标 B 和目标 C 的最短路径的蚂蚁爬行轨迹。具体路径迭代见图 12 和图 13。

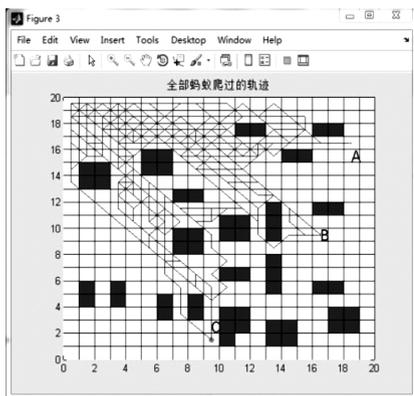


图 12 全部蚂蚁爬过的轨迹

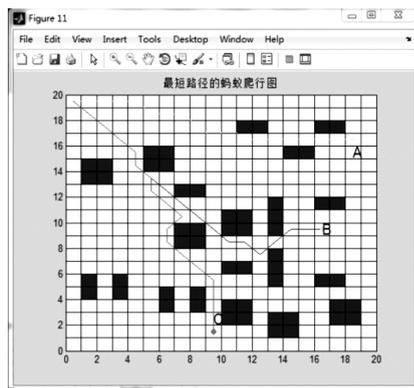


图 13 改进蚁群算法的最短路径的蚂蚁爬行图

五 优化前后的对比

通过 MATLAB 仿真最终求得总路径的最短路径。表 3 是车辆 1—9 分别到目标点 A、B、C 的路径长度。

表 3 仿真路径计算结果

路径配对	1,2,3	4,5,6	7,8,9
A 距离	23.382	22.382	19.242
B 距离	23.382	22.382	20.070
C 距离	22.968	22.554	20.898

具体如表 4 所示。

表 4 优化前与优化后的结果对比

	最短路径总长度
原始长度	69.732
蚁群算法	67.318
改进蚁群算法	60.210

六 总结与展望

对于蚁群算法来说,它的基本模型、基本框架和基本流程都已经存在,重要的是基于其基本原理与

相关问题中的限制条件进行融合,建立新的模型,使得在解决问题中更能发挥算法的有效性。在实际生活中,某些环境问题是已经有的,但还是存在不可预测的复杂问题,这就要求我们对算法模型的收敛性和算法的复杂程度做更深入的研究,从而使算法更好应用到实际问题中。通过仿真研究可以发现,蚁群算法是一种全局搜索算法,它通过不断的迭代能够避免找到局部极优问题,仿真研究是基于特定的环境的研究,在对更大空间的搜索可能需要更多的时间,所以在求解过程中可以加入针对具体问题的算法,使得可以快速找到全局最优解,避免出现局部极优解,使得二者能够有效地结合。从复杂性来看,蚁群是一个复杂的群体,它们在寻找食物过程中,从发现觅食源到最终找到最短路径的过程是很复杂的。从仿真的角度看是一个不断叠加的过程,它们通过彼此的相互作用、相互影响、相互联系、相互协同,最终找到最短路径的过程。蚁群算法是具有概率性的特征,可以将其应用到生活中的更多问题中,结合科学技术分析其收敛性,以及在不同约束条件下,分析约束条件与算法的关系,使得算法能够解决更复杂的优化问题,更好地与实践相结合,使它随着科学技术的发展而更加光明。

[参考文献]

- [1] M Dorigo, L Gambardella. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem [J]. IEEE Trans. On Evolutionary Computation, 1997, 1(1):53-66.
- [2] Laheeb M, Alzubaidy Bara S, Alhafid. Proposed Software Testing Using Intelligent Water Drop and Ant Colony Optimization Algorithm [J]. IJCSI International Journal of Computer Science Issues, 2013, 10(5):78-82.
- [3] Zainudin Zuhri, Irving Vitra Papatungan. A Hybrid Optimization Algorithm Based on Genetic Algorithm and Ant Colony Optimization [J]. International Journal of Artificial Intelligence & Applications (IJAIA), 2013, 4 (5): 63-75.
- [4] Edson Flórez1, Wilfredo Gómez2 and MSc. Lola Bautista. An Ant Colony Optimization Algorithm for Job Shop Scheduling Problem [J]. International Journal of Artificial Intelligence & Applications (IJAIA), 2013, 14 (4): 53-66.
- [5] Han-chen Huang. The Applications of Ant Colony Optimization Algorithm in Tour Route Planning [J]. Journal of Theoretical and Applied Information Technology 30th June 2013, 52(3):343-347.
- [6] 邓灵斌, 邵军. 蚁群算法在自动化仓库路径规划中的应用 [J]. 情报探索, 2014(12):70-72, 132.

Application of Improved and Colony Algorithm in the Baohe District of Hefei city Distribution of Fruits and Vegetables

DENG Ling-bin, SHAO Jun

(University of South China, Hengyang 421001, China)

Abstract: Routing optimization must be considered in the process of transportation. Reasonable paths can save time and improve the working efficiency as well. So ant colony algorithm and its improved algorithm are introduced with their superiority to solve the problem in distribution routings with the reality that residents' demand for fruits and vegetables in Baohe area are under a state of rapid development. Based on finding the optimal path, the solutions of two algorithms are compared to find out the shortest path. The results illustrate the necessity of solving the distribution routings of fruits and vegetables in Baohe area, and also show that the algorithms are adaptive in a new environment.

Key words: routing problem; improved ant colony algorithm; fruit and vegetable distribution