

基于最短路径的外卖配送时间优化问题研究

闫碧玉, 侣友友*, 马靖宇*

内蒙古大学数学科学学院, 内蒙古 呼和浩特
Email: 1575290024@qq.com, *1780578062@qq.com, *1131934347@qq.com

收稿日期: 2021年3月11日; 录用日期: 2021年4月18日; 发布日期: 2021年4月25日

摘要

针对送餐行业顾客的平均等待时间问题, 以最短路径原则为基础, 建立相应的数学模型。以某地区某天顾客的下单情况为研究对象, 运用图论理论将该地区地理简图抽象为无向赋权图, 建立顾客等待时间模型, 并利用Dijkstra算法得出两区域之间的最短路径, 采用随机模拟算法生成动态订单, 估计出顾客的平均等待时间。在此基础上, 通过改变配送中心选址、增加配送人员数以缩短顾客平均等待时间, 达到模型优化的效果。

关键词

外卖配送问题, 最短路径原则, Dijkstra算法, 图论, 时间优化

Research on Optimization of Takeaway Delivery Time Based on Shortest Path

Biyu Yan, Youyou Si*, Jingyu Ma*

School of Mathematical Sciences, Inner Mongolia University, Hohhot Inner Mongolia
Email: 1575290024@qq.com, *1780578062@qq.com, *1131934347@qq.com

Received: Mar. 11th, 2021; accepted: Apr. 18th, 2021; published: Apr. 25th, 2021

Abstract

Aiming at the average waiting time of customers in the food delivery industry, a corresponding mathematical model is established based on the shortest path principle. Taking the order situation of customers in a certain area as the research object, using graph theory to abstract the geographic diagram of the area as an undirected weighted graph, building a customer waiting time model, and using Dijkstra algorithm to find the shortest distance between the two areas, using a

*通讯作者。

random simulation algorithm to generate dynamic orders, and estimate the average waiting time of customers. On this basis, by changing the location of the distribution center and increasing the number of distribution personnel to shorten the average waiting time of customers, the effect of model optimization is achieved.

Keywords

Takeaway Delivery Issues, Shortest Path Principle, Dijkstra Algorithm, Graph Theory, Time 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

快节奏的城市生活带来送餐行业的繁荣,很多靠近城市中心的餐馆通宵营业,在接到外卖订单后,将美味的食物交给送餐员——“外卖小哥”,送到顾客手中。由于顾客耐心有限,所以“外卖小哥”需要尽可能快速地到达餐馆取餐并送到顾客所在的地址,否则会被顾客投诉导致被扣工资甚至被解雇的危险。所以如何缩短顾客等待时间成为了外卖配送的中心问题。并且在消费升级趋势下,人们对外卖配送服务的要求也日益增高,这也促使网络外卖平台不断加码以提供更高效的服务体系。要发展现代化、系统化的外卖配送系统,不容忽视的问题就是配送节点和配送线路,然而配送中心选址和配送人员数目又是对配送中心建设规划的重要问题,不仅直接影响到外卖配送的时间,而且还会影响到外卖行业的运营绩效以及未来的发展[1]。因此通过合理选取配送中心的地址,增加配送人员数,缩短顾客平均等待时间,有利于餐馆快速响应顾客需求,提高外卖行业的服务水平以及增强顾客对配送环节的满意度具有重要的现实意义。

由于配送问题在实际生活中广泛存在,目前各国学者对该问题进行了部分研究,但还有许多问题值得商榷。杨笠涵等[2]以合肥市某知名快递企业配送路径优化问题为研究对象,分析该企业配送问题,建立以配送网络成本最小为优化目标的数学模型。范立南等[3]利用遗传算法和 TSP 问题的相关理论,采用改进的自适应遗传蚁群混合算法,对沈阳大学校园内外卖配送路线进行了合理的规划,有效缩短校园外卖配送路径长度,有效提升外卖员的配送效率。黄驰等[4]应用旅行商问题的理论,对外卖配送路径问题进行建模,建立了客户房间之间的距离矩阵,运用遗传算法、MATLAB 编程得出每两个客户房间之间的最短距离、以及送完外卖后回到餐馆的最短路径。翟劲松[5]等在满足顾客需求量和时间窗约束条件下,以配送时间最短为目标,建立了外卖配送路径优化模型,运用遗传算法得出了最优的配送路径。这样的研究不胜枚举,但是以配送中心选址和配送人员数目为优化对象的研究还是寥寥无几。

本文将优化顾客平均等待时间为目标,对外卖配送中心选址问题进行研究。以最短路径原则为基础,建立了相应的数学模型,并运用图论理论将该城区地理简图抽象为无向赋权图,利用 Dijkstra 算法得出两城区之间的最短路径,以求解顾客等待时间模型,估计出顾客的平均等待时间,以及给出配送中心选址的最佳方案。与此同时,对该配送中心的配送人员数进行优化,实现两者的共同优化。

2. 顾客平均等待时间模型建立

2.1. 问题描述

以内蒙古大学周边(如下图 1 所示)某天的送餐行业顾客下单情况为对象研究其顾客下单规律,并且基

于配送中心选址和配送人员调度来优化顾客平均等待时间。由于利用高德地图 API 获取的用户路径信息包含了大量的坐标数据，实际处理难度较大，因此我们可以将该地区实际地图抽象为地理简图，其中呼和浩特乌兰夫公园和青城公园是天然屏障。在这一问题中，本文本着区内距离尽可能小，区间距离尽可能大的分类原则，使外卖各商家客户的配送都属于同城内小区域的配送。因此借助高德地图 API 接口获得内蒙古大学北校区周边地区的经纬度信息和各节点间距离，抽象出地理简图(如下图 2 所示)。



Figure 1. Map around Inner Mongolia University
图 1. 内蒙古大学周边地图

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 11 | 46 | 46 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 21 | 22 | 46 | 24 | 25 | 26 | 27 | 47 | 47 | 30 |
| 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
| 41 | 42 | 43 | 44 | 45 | 12 | 13 | 23 | 28 | 29 |

Figure 2. Geographical sketch around Inner Mongolia University
图 2. 内蒙古大学周边地理简图

上图 2 中直线表示可行路径——公路，每个方格表示一个区域(边长都为 1 公里)，方格内数字为该区域编号，其中 46、47 区代表天然屏障，无法直接穿越，即公司总部，顾客位置，餐馆位置不能位于 46 区和 47 区。该地区仅有一个送餐公司，总部现设在 25 区，有送餐员 6 名，送餐员每次送餐结束后，先返回公司总部，(等待)领取下一个订单(先到先得，按下单时间顺序领取)，然后再前往订单中指定的餐馆取餐，最后到下单顾客的地址送餐。根据该地区某日顾客下单情况，总结出下表 1。

Table 1. Customer order status table
表 1. 顾客下单情况示意表

| 序号 | 时间 | 顾客位置 | 餐馆位置 | 序号 | 时间 | 顾客位置 | 餐馆位置 |
|-----|-------|------|------|-----|--------|------|------|
| 1 | 0:00 | 16 | 2 | 2 | 0: 12 | 11 | 9 |
| 3 | 0:18 | 10 | 10 | 4 | 0: 18 | 31 | 6 |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 254 | 23:12 | 41 | 4 | 255 | 23: 18 | 31 | 25 |
| 256 | 23:36 | 31 | 15 | 257 | 23: 48 | 38 | 17 |

2.2. 模型的准备工作

2.2.1. Dijkstra 算法

Dijkstra 算法是从一个顶点 u_0 到 G 的其余各顶点的最短路径算法, 用来解决有权图中最短路径问题。特点是从起点开始, 每次遍历到距始点最近且未访问过的顶点的邻点, 直到扩展到终点 v_0 为止。具体步骤如下:

Step1: 令 $I(u_0) = 0$, 对 $v \neq u_0$, 令 $I(v) = \infty$, $S_0 = \{u_0\}$, $i = 0$ 。

Step2: 对每一个 $v \in \overline{S_i} (\overline{S_i} = V \setminus S_i)$, 用 $\min_{u \in S_i} \{I(v), I(u) + w(uv)\}$ 代替 $I(v)$, 这里 $w(uv)$ 表示顶点 u 和 v 之间边的权值。计算 $\min_{u \in S_i} \{I(v)\}$, 把达到这个最小的一个顶点记为 u_{i+1} , 令 $S_{i+1} = S_i \cup \{u_{i+1}\}$ 。

Step3: 若 $i = |V| - 1$, 则停止; 若 $i < |V| - 1$, 则用 $i + 1$ 代替 i , 转 step2。

2.2.2. 最短送餐路径模型

最短送餐路径模型的建立以送餐员从公司总部到餐馆再从餐馆到顾客, 最后从顾客返回公司总部经过距离最短为目标, 就可以得到每次送餐的最短路径。为了计算方便将 46 区和 47 区重新编号, 将城市地理简图抽象为无向赋权图 $G(V, E)$ (如下图 3 所示), 起到加速运算的作用。取两相邻路口之间的道路长度为其对应的权值, 将其抽象成一个加权邻接矩阵 $A_{50 \times 50}$, 建立初始加权邻接矩阵 $A_{50 \times 50}$ 。

$$A_{50 \times 50} = \begin{bmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,49} & s_{1,50} \\ s_{2,1} & s_{2,2} & \cdots & s_{2,49} & s_{2,50} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ s_{49,1} & s_{49,2} & \cdots & s_{49,49} & s_{49,50} \\ s_{50,1} & s_{50,2} & \cdots & s_{50,49} & s_{50,50} \end{bmatrix}$$

其中

$$s_{i,j} = \begin{cases} 1, & \text{当 } v_i \text{ 和 } v_j \text{ 之间有公路连通, } i \neq j \\ \infty, & \text{当 } v_i \text{ 和 } v_j \text{ 不相邻或由于 46, 47 区阻隔时, } i \neq j \\ 0, & i = j \end{cases}$$

由于为 $G(V, E)$ 无向赋权图, $A_{50 \times 50}$ 为对称矩阵, 有 $s_{i,j} = s_{j,i}$ 。

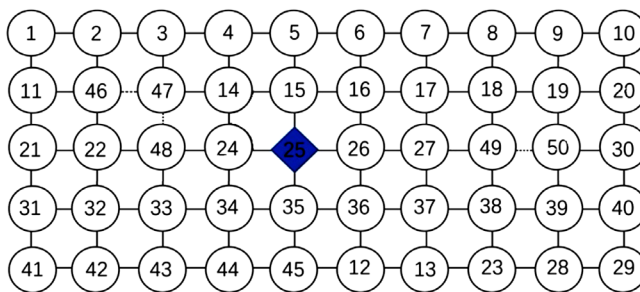


Figure 3. Undirected weighted graph
图 3. 无向赋权图

通过 Dijkstra 算法, 以公司总部位置为起始点, 可以得到一个起始点与各个点之间的最短距离矩阵 $leastpath_{45 \times 1}$, 并将其总结表格(如下表 2 所示), 提取出餐馆与公司之间的最短距离, 再将餐馆位置替换成起始点, 同理可得餐馆到顾客之间的最短距离。最后在以顾客位置为起始点得到从顾客位置回到公司总部的最短距离。

Table 2. Shortest distance
表 2. 最短距离

| 城区 | 距离 | 城区 | 距离 | 城区 | 距离 | 城区 | 距离 | 城区 | 距离 |
|----|----|----|----|----|----|----|----|----|----|
| 1 | 6 | 10 | 7 | 19 | 5 | 28 | 5 | 37 | 3 |
| 2 | 5 | 11 | 5 | 20 | 6 | 19 | 6 | 38 | 4 |
| 3 | 4 | 12 | 3 | 21 | 4 | 30 | 5 | 39 | 5 |
| 4 | 3 | 13 | 4 | 22 | 3 | 31 | 5 | 40 | 6 |
| 5 | 2 | 14 | 2 | 23 | 5 | 32 | 4 | 41 | 6 |
| 6 | 3 | 15 | 1 | 24 | 1 | 33 | 3 | 42 | 5 |
| 7 | 4 | 16 | 2 | 25 | 0 | 34 | 2 | 43 | 4 |
| 8 | 5 | 17 | 3 | 26 | 1 | 35 | 1 | 44 | 3 |
| 9 | 6 | 18 | 4 | 27 | 2 | 36 | 2 | 45 | 2 |

2.3. 模型构建

为了保证模型准确、运算简便，做出以下几点假设：

- 1) 假设送餐员在公路上的平均行驶速度为 30 公里每小时。
- 2) 假设每块区域的所有顾客位置、餐馆位置和公司总部都位于该地区区域的左下角。
- 3) 假设商家和顾客服务时间之和服从正态分布 $N(\mu, \sigma^2)$ 。其平均值是 3 分钟，方差为 1 分钟，即 $\mu = 3, \sigma^2 = 1$ 。
- 4) 忽略送餐员拐弯的时间，仅考虑沿道路行驶的时间。

一次完整的送餐包括送餐员从公司总部出发前往餐馆、从餐馆前往顾客位置、将外卖移交到顾客手中及从顾客位置返回公司总部四部分组成。故可将顾客的等待时间划分为送餐员回程时间，从公司到餐馆时间，从餐馆到顾客位置所需时间及服务时间四部分，分别用 $t_{hc}^{(j)}, t_{cr}^{(j)}, t_{rb}^{(j)}, t_{dt}^{(j)}$ 表示上述四个时间(j 指第 j 个顾客)，则顾客的等待时间模型为： $t_j = t_{hc}^{(j)} + t_{cr}^{(j)} + t_{rb}^{(j)} + t_{dt}^{(j)}$ ，上述所有的时间都应为最短时间，根据最短送餐路径模型可确定最短路径。

基于上述分析，可将参数设置如下：设 T 为下一次顾客下单的时间点， T' 为本次下单时间点， $t_d^{(j,j+1)}$ 为上一次顾客下单和本次顾客下单之间的时间间隔，则每一次顾客下单时间点可由递推公式表示为： $T = T' + t_d^{(j,j+1)}$ 。

t_j 为顾客的等待时间， $t_{hc}^{(j)} = T_{\min}^{(j)} - T^{(j)}$ 为回程等待时间，即最快回到公司总部的送餐人到达公司的时刻 $T_{\min}^{(j)}$ 与顾客下单时刻 $T^{(j)}$ 之差；其中 $T^{(j)}$ 为顾客下单的时间点， $T_{\min}^{(j)} = \min\{T_{s1}^j, T_{s2}^j, T_{s3}^j, T_{s4}^j, T_{s5}^j, T_{s6}^j\}$ 为最先返回公司的送餐员到达公司的时间点。

若在下一次顾客下单时，有送餐员位于公司，则 $t_{hc}^{(j)} = 0$ ，所以有 $t_j = t_{cr}^{(j)} + t_{rb}^{(j)} + t_{dt}^{(j)}$ 。我们还有如下关系： $t_{cr}^{(j)} = \frac{L_{cr}}{v}, t_{rb}^{(j)} = \frac{L_{rb}}{v}$ ，其中 L_{cr} 为公司到餐馆的最短距离， L_{rb} 为餐馆到顾客的最短距离， v 代表送餐员行驶的平均速度。

3. 最短送餐路径模型和顾客等待时间模型求解

3.1. 算法概述

首先对表 1 中的下单时间，顾客位置，餐馆位置进行数据处理。统计顾客下单时间间隔、下单位置以及顾客位置的概率分布列，运用极大似然估计法和 χ^2 拟合检验法，通过样本频率估计总体概率，可以

得出顾客下单间隔、顾客位置以及餐馆位置的总体概率分布函数分别为 $F_1(x)$, $F_2(x)$, $F_3(x)$ 。同时由于服务时间遵循正态分布, 则其概率分布函数为 $F_4(x) = \int_0^x \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-t)^2}{2}} dt (0 < x < +\infty)$ 。设定顾客的下单时间点为观测量, 下单间隔、餐馆位置、顾客位置及服务时间均设定为模拟量, 运用 MATLAB 对 0 到 1 之间进行随机数产生模拟值, 并构建随机数和随机变量的函数关系, 即根据其概率分布列, 产生相应的模拟量。遵循最短路径原则, 以顾客等待时间最短为目标, 建立顾客等待时间模型, 得到顾客平均等待时间。

3.2. 算法步骤

下单时间间隔, 餐馆位置, 顾客位置都为离散型分布, 三者可用相同的模拟算法, 步骤如下:

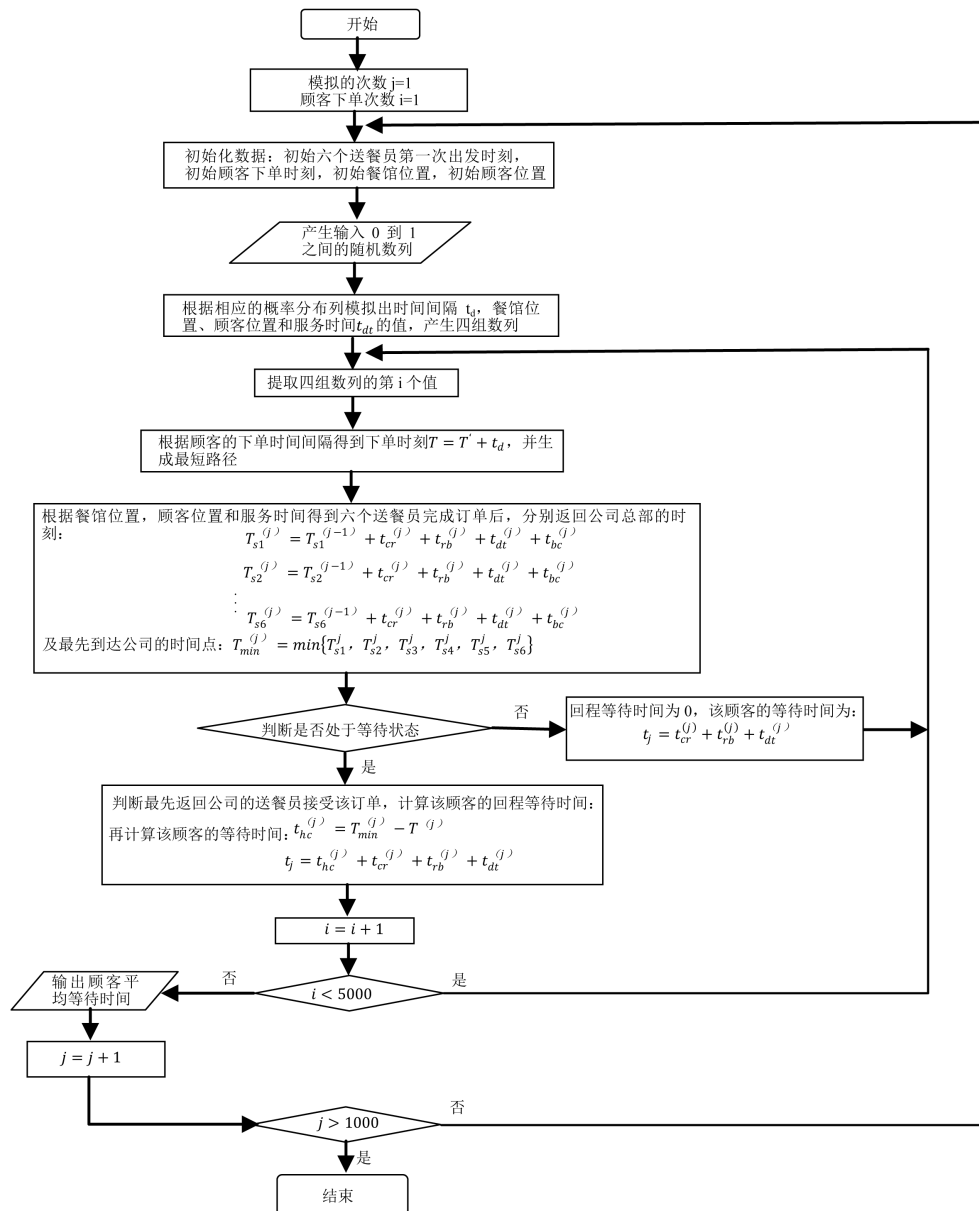


Figure 4. Overall flow chart of the algorithm

图 4. 算法整体流程图

Step1: 用计算机产生 n 个 $[0,1]$ 之间的随机数 $\xi_1, \xi_2, \dots, \xi_n$;
 Step2: 令 $r = 1$, 判断 ξ_r , 若 $F_1(k-1) \leq \xi_r < F_1(k)$, ($k = 1, 2, \dots, 36$), 则使 $\xi_r = k$ 。
 Step3: 判断 r 是否小于 n , 若是, 令 $r = r + 1$, 返回 Step2; 否则, 结束。

服务时间为连续性分布, 方法如下:

Step1: 用计算机产生 n 个 $[0,1]$ 之间的随机数 $\xi_1, \xi_2, \dots, \xi_n$;
 Step2: 令 $r = 1$, 判断 ξ_r , 若 $\xi_r = F_3(k), (k \geq 0)$, 则使 $\xi_r = k$;
 Step3: 判断 r 是否小于 n , 若是, 令 $r = r + 1$, 返回 Step2; 否则, 结束。
 算法流程图如图 4 所示。

3.3. 结果分析

经 3.2 节所述方法, 求解得到了 1000 次每次 5000 个顾客下单的情况, 得到了共 1000 组数据结果。用 SPSS21.0 对其进行描述性分析可得下表 3 及下图 5。

Table 3. Simulation solution results table

表 3. 模拟求解结果数据表

| 最大值 | 最小值 | 平均值 | 方差 | 标准差 | 偏度 | 峰度 |
|-------------|-------------|-------------|-------|--------|-------|------|
| 38.1211 min | 23.7571 min | 28.4644 min | 4.241 | 2.0593 | 0.859 | 0.07 |

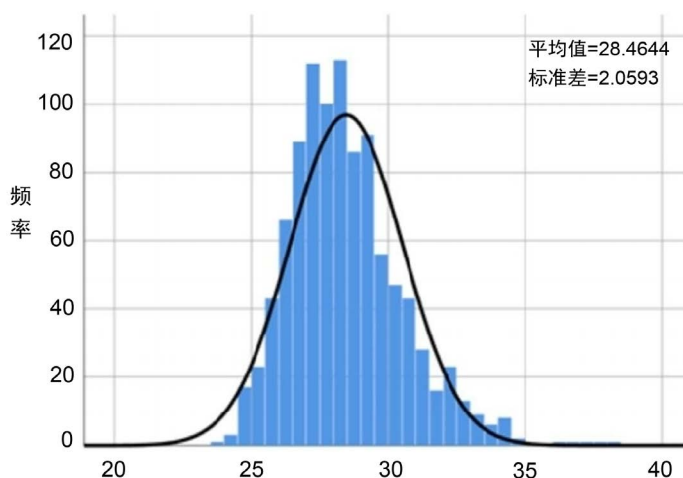


Figure 5. Data distribution of simulation solution results
 图 5. 模拟求解结果数据分布图

由图表数据可以看出数据较为集中的分布在中部, 集中于 27 至 29.5 之间。由表中数据可得顾客平均等待时间为 28.4644 分钟, 且方差较小, 证明数据幅度不大, 稳定性高。所以顾客的等待时间应为 28.4644 分钟。

4. 基于配送中心选址和增加配送人员子方案的顾客平均等待时间优化

4.1. 基于配送中心选址子方案的顾客平均等待时间优化

当起始位置发生改变, 基于顾客下单规律与最短送餐路径模型, 进行顾客等待时间模型的分析, 顾客的等待时间由回程等待时间, 从公司到餐馆时间, 从餐馆到顾客位置所需时间, 服务时间四部分组成, 回程等待时间和公司到餐馆所需时间均发生改变, 分别用 $t_{hc,k}^{(j)}, t_{cr,k}^{(j)}, t_{rb}^{(j)}, t_{dt}^{(j)}$ 表示上述四个时间, k 代表总部位

置是在 k 区域, j 指第 j 个顾客, 则顾客等待时间的函数模型 $t_{j,k} = t_{hc,k}^{(j)} + t_{cr,k}^{(j)} + t_{rb}^{(j)} + t_{dt}^{(j)}$, $t_{j,k}$ 为公司总部位于 k 区域时顾客的等待时间, 上述所有的时间都应为最短时间, 根据最短送餐路径模型可确定最短路径。

参数如下: $t_{hc,k}^{(j)} = T_{\min,k}^{(j)} - T_k^{(j)}$ 为公司总部位于 k 区域的回程等待时间,

$T_{\min,k}^{(j)} = \min\{T_{s1,k}^j, T_{s2,k}^j, T_{s3,k}^j, T_{s4,k}^j, T_{s5,k}^j, T_{s6,k}^j\}$ 为公司总部位于 k 区域时最先返回公司的送餐员到达公司的时间点, 其中 $T_{s1,k}^j$ 表示对 j 次顾客下单时, 为公司总部位于 k 区域时, 第一位送餐员处理完手上的订单返回公司的时间点。

此外, 还有如下关系式: $t_{cr,k}^{(j)} = \frac{L_{cr,k}}{v}, t_{rb}^{(j)} = \frac{L_{rb}}{v}$, $t_{cr,k}^{(j)}$ 为公司总部位于 k 区域公司到餐馆时间, $L_{cr,k}$ 为公司总部位于 k 区域时公司到餐馆的最短距离, v 代表送餐员行驶的平均速度。

参照流程图 4, 设置模拟 3 次每次 5000 次下单, 可输出最终位于 26 区左下角时, 平均响应时间最短, 该时间为 27.6216 分钟。因此, 公司总部应该设置在 26 区。

4.2. 基于增加配送人员子方案的顾客平均等待时间优化

当改变送餐员数量, 即改变了回程等待时间。顾客等待时间的函数模型仍为 $t_j = t_{hc}^{(j)} + t_{cr}^{(j)} + t_{rb}^{(j)} + t_{dt}^{(j)}$, 但 $t_{hc}^{(j)}$ 会随送餐员数量增加而减小, 上述所有的时间都应为最短时间, 根据最短送餐路径模型可确定最短路径。则 t_j 为顾客等待时间, $t_{hc}^{(j)} = T_{\min}^{(j)} - T^{(j)}$ 为回程等待时间, 即最快回到公司总部的送餐人到达公司的时刻 $T_{\min}^{(j)}$ 与顾客下单时刻 $T^{(j)}$ 之差; 其中 $T^{(j)}$ 为顾客下单的时间点, $T_{\min}^{(j)} = \min\{T_{s1}^j, T_{s2}^j, T_{s3}^j, T_{s4}^j, T_{s5}^j, T_{s6}^j\}$ 为最先返回公司的送餐员到达公司的时间点。

若在下一顾客下单时, 有送餐员位于公司, 则 $t_{hc}^{(j)} = 0$, 所以有 $t_j = t_{cr}^{(j)} + t_{rb}^{(j)} + t_{dt}^{(j)}$, 假设送餐员数量由原来的 t 个增加到了 $t+k$ 个, 则最起码可以保证前 $t+k$ 个订单无需等待, 即回程等待时间为 0。我们还有如下关系: $t_{cr}^{(j)} = \frac{L_{cr}}{v}, t_{rb}^{(j)} = \frac{L_{rb}}{v}$, 其中 L_{cr} 为公司到餐馆的最短距离, L_{rb} 为餐馆到顾客的最短距离, v 代表送餐员行驶的平均速度。

同理参照流程图 4, 设置模拟 3 次每次 5000 次下单的情况求解, 我们得出当增加人数趋于无穷时, 顾客的平均等待时间于 17.113 min 浮动。本持着送餐成本尽可能低以及顾客等待时间尽可能短的原则, 本文取输出数据中第一次达到此值的数据, 其对应的送餐员数为最小应该增加的送餐员数, 即 9 人, 可达到顾客平均等待时间所能达到的最小值。

5. 结论

由于当前缺少有关外卖配送优化问题的研究, 故本文在多车型配送, 单配送仓库条件下, 通过分析内蒙古大学周边某天顾客的下单情况, 建立顾客等待时间模型, 并利用 Dijkstra 算法以及随机模拟算法进行求解, 在此基础上, 通过改变公司总部选址和增加送餐员数量来优化顾客平均等待时间。实验结果验证了模型的合理性以及算法的有效性, 并得出以下结论:

- 1) 该模型基于最短路径原则, 大大缩短了配送距离和顾客平均等待时间, 从而提高了配送效率。所以送餐人员可以在短时间内完成配送, 提高顾客满意度, 有助于外卖行业的长久发展。
- 2) 本文建立的模型可用于确定配送中心的位置以及最优配送人员数, 且可为外卖配送优化问题提供较为新颖的求解思路, 使外卖配送规划更具科学性, 对外卖配送领域的研究有一定的贡献。

但是该模型所用到的数据只是一天的数据, 不同日期对应的下单时间间隔, 顾客位置, 餐馆位置会发生相应的改变, 在下一步的研究中, 还需结合大数据方法, 用海量数据研究其动态变化规律, 使模型更加优化。

参考文献

- [1] 吴丽敏. 基于顾客满意度的多目标配送中心选址方法研究[J]. 物流科技, 2014(6): 95-97.
- [2] 杨粟涵, 于蕾. 基于遗传算法的快递配送路径优化问题研究[J]. 现代信息科技, 2020, 4(9): 99-103.
- [3] 范立南, 吕鹏. 基于改进遗传算法的校园外卖配送路径规划[J]. 物流科技, 2021(1): 14-19.
- [4] 黄驰, 黄耿石, 朱小玲. 基于遗传算法的送外卖最短路径研究[J]. 科技传播, 2016, 12(6): 94-95.
- [5] 翟劲松, 台玉红. 基于时间窗约束下的外卖配送路径优化[J]. 物流科技, 2018(3): 15-18.