

1 引言.....	3
1.1 编写目的.....	3
1.2 背景.....	3
1.3 定义.....	3
1.4 参考资料.....	4
2 任务概述.....	5
2.1 目标.....	5
2.2 用户的特点.....	6
3 需求规定.....	7
3.1 对功能的规定.....	7
3.1.1 位置获取功能.....	7
3.1.2 地图渲染功能.....	7
3.1.3 路径规划功能.....	8
3.1.4 交互式引导功能.....	8
3.1.5 常见问题快速查询功能.....	10
3.2 对性能的规定.....	10
3.2.1 精度.....	10
3.2.2 时间特性要求.....	11
3.2.3 灵活性.....	11
3.3 输入输出要求.....	11
4 运行环境规定.....	11

4.1 设备 .....	11
4.2 支持软件 .....	12
4.3 接口 .....	12
4.3.1 模拟定位服务接口.....	12
4.3.2 地图数据接口.....	12
4.3.3 路径规划接口.....	12
4.3.4 用户接口 .....	13
5 算法论述.....	13
5.1 解题思路 .....	13
5.1.1 数字地图的制作.....	13
5.1.2 数字地图的发布.....	14
5.1.3 数字地图的展示.....	14
5.1.4 路径规划的实现.....	14
5.1.5 纠偏功能的实现.....	14
5.2 算法 .....	15
5.2.1 Delaunay 三角网剖分.....	15
5.2.2 Dijkstra 算法.....	16
5.2.3 A*算法 .....	17
5.3 改进 .....	18

# 需求分析与算法论述

## 1 引言

### 1.1 编写目的

本需求说明书目的在于：将用户提供的需求描述系统化、精确化、全面化。从而实现：

1. 便于用户、分析人员和设计人员进行理解和交流。
2. 支持目标软件系统的确认。
3. 控制系统进化过程。

预期读者：软件设计者和测试者。

### 1.2 背景

说明：

- a. 待开发软件系统的名称：室内交互式引导 APP；
- b. 本项目的任务提出者：中国民航信息网络股份有限公司；
- c. 本项目的任务开发者：Dream Travellers 团队；
- d. 此软件系统任务用户：机场旅客；

### 1.3 定义

定位接口：用于进行用户定位的模拟定位服务接口。

起点：导航功能的起始点。

中转点：导航功能的中转点，即从起点出发，在到达终点前需要

经过的点。

终点：导航功能的终点。

FAQ: Frequently Asked Questions, 即常见问题解答。

#### 1.4 参考资料

- [1] 朱建佳, 邹仕洪. 基于 Android 平台的室内定位系统的设计与实现[J]. 2013.
- [2] 徐静. 室内移动导航系统的路径规划方法研究[D]. 长春理工大学, 2009.
- [3] Miu A K L. Design and implementation of an indoor mobile navigation system[D]. Massachusetts Institute of Technology, 2002.
- [4] 陶嘉明. 基于 ArcGIS 的室内地图服务系统的设计与实现[D]. 北京邮电大学, 2011.
- [5] 杨德君. 室内精确定位导航系统的设计与实现[D]. 北京邮电大学, 2011.
- [6] 刘兆宏, 王科, 丰江帆, 等. 矢量室内地图建模与制作方法[J].
- [7] 卢伟, 魏峰远, 张硕, 等. 室内路网模型的构建方法研究与实现[J]. 导航定位学报, 2014 (4): 63-67.

## 2 任务概述

### 2.1 目标

近年来，随着无线网络的高速发展，以及智能手机、平板电脑等通讯设备的普及，大量的移动应用软件应运而生，导航与智能手机也在紧密的结合，GPS 卫星定位与导航功能已经成为了现在智能手机的基本配置和功能，而且借助于 3G 和 4G 等通信网络，可以方便用户进行户外的定位，让用户便捷出行。然而，其随着城市变化突飞猛进，城市的建筑物不断地增加，大型建筑物内部的结构也更加紧凑和密集，有效的导航技术能帮助人们用最短的时间，走最短的路，尽快地完成事情，提高效率。一直以来，人们对于室内导航定位都有强烈需求，尤其在一些复杂室内环境中，如大超市、候机楼、展会、仓库、机场大厅等。机场等大型公共建筑内部人流复杂，寻找一条通向目的位置的路径很不直观。

当用户进入陌生室内建筑时，由于建筑物内布局复杂，无法得知自身所处的环境位置，通过室内定位系统，找出自身所在位置和显示周围的地理信息，得到目的地信息并和目的地形成导航，从而快速准确地指引用户。随着移动互联网蓬勃发展，各种位置服务(Location Based Service)应用层出不穷，为解决室内导航的问题提供了一条路径。据民航局统计，2013 年机场旅客吞吐量 7.54 亿人次，比上年增长 11.0%，旅客吞吐量达百万以上的机场有 61 个（数据来源：民航局《2013 年民航行业发展统计公报》）。2014 全年的统计数据还未出

炉，但预计旅客量仍将增长 10%。一般旅客提前 1 至 2 小时抵达机场，若因流量管制、天气等原因导致航班延误，旅客逗留机场时间还会增加。这给机场旅客疏导带来了难题。

为解决机场旅客疏导的问题，打算设计一款用于机场的室内交互式引导 APP。该 APP 提供实时定位功能，进行可视化引导。当用户进入机场中，可通过该 APP 获取他当前的位置，并且通过导航功能引导用户完成领取登机牌、托运行李、前往安检通道、到达登机口等功能，还提供了多点导航功能，通过语音、文字或地图选点等选取目的地，方便旅客自主地选择目的地，如商店，餐厅等。同时提供了交互式查询，如方便用户查询航班动态、行李状态、交通指南、天气状况、乘机指南等自助服务，提升用户的出行体验质量，让旅客出行更方便，同时也有效地疏导机场旅客。

## 2.2 用户的特点

本软件的用户为机场的旅客，用户一般拥有一台以上的移动设备，其最大的需求是解决“找路”的问题。

由于用户在到达机场时会使用此 APP，根据机场的客流量，预计本软件的使用频度为 40000 人次/天。

### 3 需求规定

#### 3.1 对功能的规定

##### 3.1.1 位置获取功能

获取位置信息是导航软件的必备功能，包括获取起点、终点以及中转点的位置信息，具体实现方式应有以下几种：

1. 通过定位服务接口获取用户当前所在位置
2. 在列表中选择位置，要求具有搜索功能
3. 利用摄像头拍摄商户照片或扫描二维码确定位置
4. 利用麦克风使用语音输入位置

##### 3.1.2 地图渲染功能

1. 导航功能在机场平面图上实现，该功能应以清晰直观、简洁易懂的方式在地图上绘制出起点、中转点、终点以及路线等标记。
2. 用户可以通过手势或滑块对地图进行缩放，缩放的规模应受到限制，避免显示的地图过小或过大。
3. 在缩放或移动地图时，地图上的标记也应做相应调整，保证在地图上能够正常显示。
4. 当用户点击地图上某区域时，应弹出标签显示该区域的详细信息，并提供直接导航至此处的按钮。

### 3.1.3 路径规划功能

1. 路径规划功能应为用户规划一条从起点到终点并经过中转点（如果存在）的最优路径，其中中转点应按是否有先后顺序分为两类。对于规定了先后顺序的中转点，应按照顺序依次经过；否则应按照距离最短的原则将中转点插入路径当中。
2. 若用户在行进过程中未按照规划好的路径前行，偏离了路线，这时应该重新进行路径规划。以用户当前所处的点作为起点，未到达的中转点和终点不变，重新规划路径并进行导航，新的路径仍应是当前点经过中转点到达终点的最优路径。
3. 机场平面分为公共区域和安全区域。这两个区域是有向的，例如电梯 ES5 和 ES6 可从 1 楼到达 2 楼，但不可从 2 楼到达 1 楼，规划路径时应当考虑到这一点。
4. 规划路径时不应只考虑路径长度，根据机场的实际情况，应尽量规划出一条通畅且不逆行的道路，从而避免机场的拥堵，同时优化用户的体验。

### 3.1.4 交互式引导功能

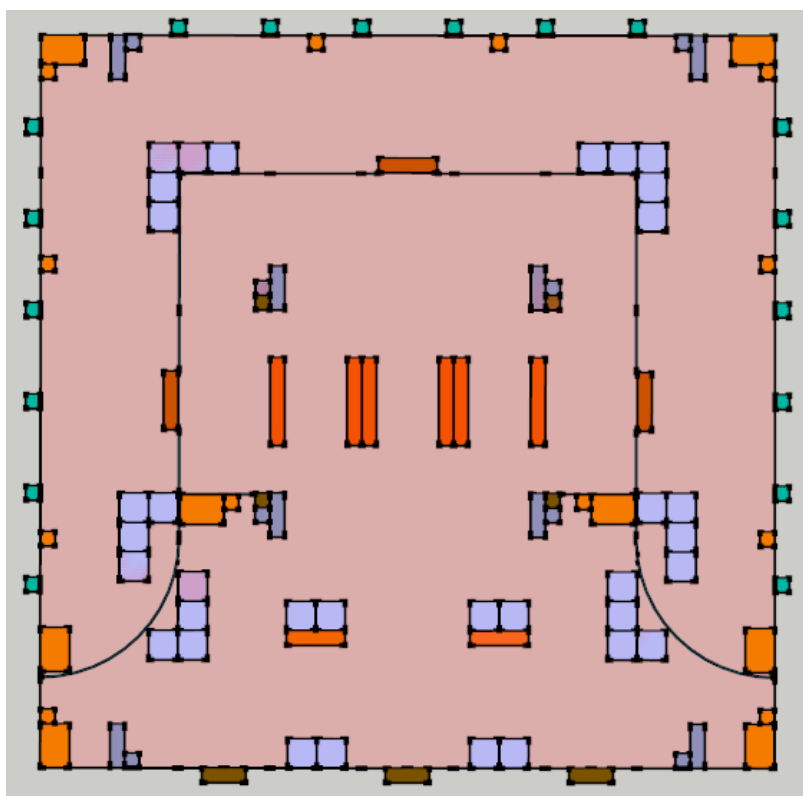
1. 在引导用户前往目的地的过程中，地图上要实时显示用户当前所在位置，默认该点为地图中心点，并将地图放大到合适的倍数。
2. 在地图的一角显示缩略图，为用户提供全局的路径信息。
3. 根据路径规划结果进行提示，如前行、左右转向等。
4. 当用户向前直行时，提示用户直行米数，并且随着用户的前行该



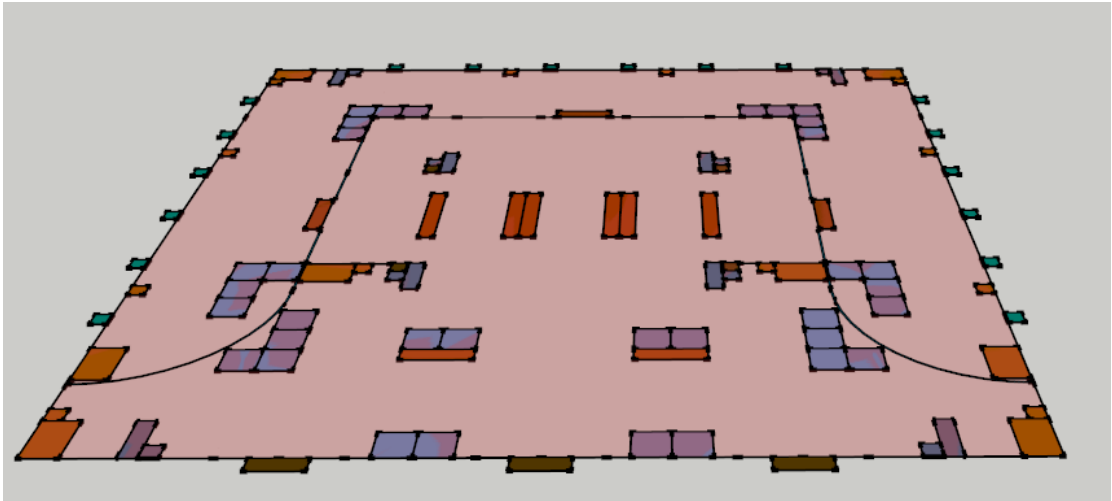
距离不断减少。同样，遇到路径中的转弯时，也要提示用户前方多少米左转或前方多少米右转，方便用户提前做出判断。

5. 在直行和左右转弯进行状态切换时，应给出语音提示，同时以震动的方式将方向反馈给用户，如直行进行一次短震、左转进行一次长震、右转进行两次短震。
6. 地图的显示应能在俯视图与透视图间切换。

俯视图：



透视图：



### 3.1.5 常见问题快速查询功能

APP 中应将用户常见的问题总结为列表(FAQ)，在用户选择后给出相应的解决方案。例如，当用户选择“遗失了身份证怎么办？”后，首先应给出办理临时登机证明的方法，并提供按钮将用户直接导航至民航公安机关的值班场所或相应的问讯处，帮助用户快速解决问题；当用户需要直接登机时，我们可以提供登机的一条龙服务，即直接引导用户前往最近的值机柜台、安检口以及登机口；当发生危险情况，需要紧急疏散时，我们也可以根据机场的安排将用户引导至安全出口处。

## 3.2 对性能的规定

### 3.2.1 精度

定位及规划应精确到 1 米一下。

### 3.2.2 时间特性要求

所有操作应在数秒内完成。

### 3.2.3 灵活性

- a. 运行环境的变化：在不同分辨率的设备上都能够正常运行；
- b. 精度的变化：精度的变化将影响定位的精确度，但软件仍能正常运行；
- c. 计划的变化或改进：较易改进。

## 3.3 输入输出要求

输入数据通过定位接口、点击地图、用户在列表中选择以及摄像头和麦克风得到，分为起点、中转点和终点三类；输出为用户界面、规划好的路径以及交互引导信息。

## 4 运行环境规定

### 4.1 设备

列出运行该软件所需要的硬设备。说明其中的新型设备及其专门功能，包括：

- a. 处理器型号及内存容量：CPU 主频 1GHz 以上，RAM 容量 1GB 以上；
- b. 输入及输出设备的型号和数量，联机或脱机：Android 4.0 及以上系统的移动终端 1 台，需联机；

- c. 数据通信设备的型号和数量：服务器 1 台、Android 4.0 及以上系统的移动终端 1 台；

## 4.2 支持软件

支持 Android 4.0 及以上系统。

## 4.3 接口

### 4.3.1 模拟定位服务接口

1. 请求方式：HTTP GET 请求；
2. 输入参数：无；
3. 返回值：json 格式 $[x,y,z]$ ，其中  $x,y$  取值范围为 $(0,1)$ ，以地图左下角为坐标原点， $z$  的取值为 $[0,2]$ ，对应 B1~F2 层。

### 4.3.2 地图数据接口

1. 请求方式：HTTP GET 请求；
2. 输入参数：请求的图层、图片格式、版本号、地图种类等；
3. 返回值：对应格式的地图图片。

### 4.3.3 路径规划接口

1. 请求方式：HTTP POST 请求；
2. 输入参数：JSON 格式的目的地对象数组字符串，对象包含  $x$ 、 $y$ 、 $z$  坐标信息。

3. 返回值：JSON 格式的字符串，包含了一个 GeoJSON 格式的对象数组，每个对象保存了一段路径以及这段路径的几何和长度信息。

#### 4.3.4 用户接口

即用户界面，根据屏幕提示，用户可自行选择起点、中转点和终点。

### 5 算法论述

#### 5.1 解题思路

##### 5.1.1 数字地图的制作

将题目所给的 skp 文件导出为 CAD 的 dwg 格式，再导入地图制作软件中，从中提取出几何信息，制作以下 Shapefile 文件：

外框：对地图外墙的图形化表示，线型；

组件：对地图上组件的图形化表示，多边形，包含 Type（组件类型）、Name（组件名称）、Id（组件编号）等字段；

墙：对地图内墙的图形化表示，多边形；

兴趣点(POI, Point of Interest)：对功能组件的抽象表示，点型，包含 Type(组件类型)、Name(组件名称)、Id(组件编号)、Point\_X(X 坐标)、Point\_Y(Y 坐标) 等字段。

路网：对地图上道路的抽象表示，线型，包含 Length（长度）等

字段。路网的生成使用了 Delaunay 三角剖分法，在编写算法生成并筛选路径后，人工再对其进行优化。

### 5.1.2 数字地图的发布

制作好 Shapefile 文件后，将用于显示的图层（外框、组件和墙）放在开源地理信息系统服务器 GeoServer 中发布。

### 5.1.3 数字地图的展示

利用开源框架 OpenLayers，编写 JavaScript 代码，从 GeoServer 获取地图数据并展示在浏览器中（Android App 中使用 WebView）。

### 5.1.4 路径规划的实现

将制作好的路网的 Shapefile 导入 PostgreSQL 数据库中，安装 PostGIS 和 pgRouting 开源插件，通过执行 SQL 语句实现路径规划。

服务器脚本使用 PHP 编写，该脚本从 POST 请求中接收 JSON 格式的目的地对象数组字符串，对其进行解析后，分段使用 SQL 语句对数据库进行查询，并将查询得到的 GeoJSON 格式的结果存放于数组中，以 JSON 字符串的格式返回路径数据。浏览器收到数据后，对数据进行解析并利用 OpenLayers 进行显示。

### 5.1.5 纠偏功能的实现

在得到路径规划的结果后，可以生成一定宽度的路径缓冲区，若当前位置与路径缓冲区不相交，则说明当前位置偏离了规划的路径。

此时应从当前位置开始，对尚未到达的目的地重新进行路径规划。

各目的地点同样需要生成缓冲区。当前位置与目的地点的缓冲区相交时，认为已经到达该目的地，该目的地会被移出目的地列表。

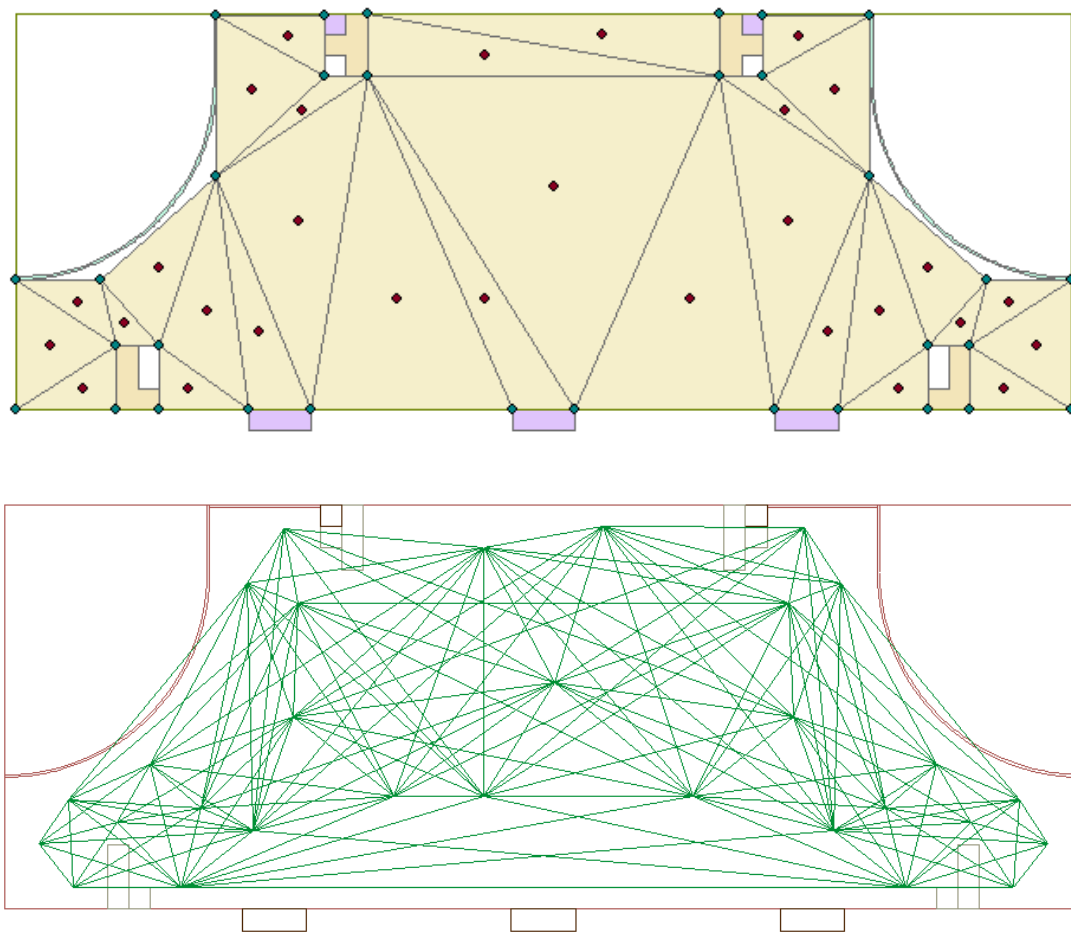
缓冲区的生成以及相交的判断都通过开源的 JavaScript 库 JSTS 来实现。

## 5.2 算法

### 5.2.1 Delaunay 三角网剖分

Delaunay 三角网剖分方法属于生成非结构化网格的方法，这类方法逐步成为目前最流行的全自动网格生成方法之一，其对于区域边界线和内部媒质分界线形状不规则的情况以及场的分布变化较大的情况都能较好的适应。

Delaunay 三角网剖分是前苏联数学家 Delaunay 在 1934 年提出的：对于任意给定的平面点集，只存在着唯一的一种三角网剖分方法，满足所谓的“最大化最小角”优化准则，即所有最小内角之和最大。这种剖分方法遵循“空外接圆”和“最小角最大”准则，因此，在各种二维三角网剖分中，只有 Delaunay 三角网剖分才同时满足全局和局部最优，特别适用于有限元分析应用中的网格生成，获得性能优良、形状最佳的三角形单元。



### 5.2.2 Dijkstra 算法

戴克斯特拉算法（英语：Dijkstra's algorithm）是由荷兰计算机科学家艾兹赫尔·戴克斯特拉提出。迪科斯彻算法使用了广度优先搜索解决非负权有向图的单源最短路径问题，算法最终得到一个最短路径树。该算法常用于路由算法或者作为其他图算法的一个子模块。

该算法的输入包含了一个有权重的有向图  $G$ ，以及  $G$  中的一个来源顶点  $S$ 。我们以  $V$  表示  $G$  中所有顶点的集合。每一个图中的边，都是两个顶点所形成的有序元素对。 $(u, v)$  表示从顶点  $u$  到  $v$  有路径相连。我们以  $E$  表示  $G$  中所有边的集合，而边的权重则由权重



函数  $w: E \rightarrow [0, \infty]$  定义。因此,  $w(u, v)$  就是从顶点  $u$  到顶点  $v$  的非负权重 (weight)。边的权重可以想像成两个顶点之间的距离。任两点间路径的权重, 就是该路径上所有边的权重总和。已知有  $V$  中有顶点  $s$  及  $t$ , Dijkstra 算法可以找到  $s$  到  $t$  的最低权重路径 (例如, 最短路径)。这个算法也可以在一个图中, 找到从一个顶点  $s$  到任何其他顶点的最短路径。

### 5.2.3 A\*算法

A\*搜索算法, 俗称 A 星算法, 是一种在图形平面上, 有多个节点的路径, 求出最低通过成本的算法。该算法像 Dijkstra 算法一样, 可以找到一条最短路径; 也像 BFS 一样, 进行启发式的搜索。

在此算法中, 如果以  $g(n)$  表示从起点到任意顶点  $n$  的实际距离,  $h(n)$  表示任意顶点  $n$  到目标顶点的估算距离, 那么 A\*算法的公式为:  $f(n)=g(n)+h(n)$ 。这个公式遵循以下特性:

1. 如果  $h(n)$  为 0, 只需求出  $g(n)$ , 即求出起点到任意顶点  $n$  的最短路径, 则转化为单源最短路径问题, 即 Dijkstra 算法;
2. 如果  $h(n) \leq$  “ $n$  到目标的实际距离”, 则一定可以求出最优解。

而且  $h(n)$  越小, 需要计算的节点越多, 算法效率越低。

一种具有  $f(n)=g(n)+h(n)$  策略的启发式算法能成为 A\*算法的充分条件是:

1. 搜索树上存在着从起始点到终了点的最优路径;
2. 问题域是有限的;

3. 所有结点的子结点的搜索代价值 $>0$ ;
4.  $h(n) \leq h^*(n)$  ( $h^*(n)$  为实际问题的代价值);

当此四个条件都满足时, 一个具有  $f(n)=g(n)+h(n)$  策略的启发式算法能成为 A\* 算法, 并一定能找到最优解。

### 5.3 改进

目前使用 Delaunay 三角剖分算法和自编路径生成算法所产生的路网还不能满足令路径足够短的需求, 仍需要对算法进行改进并对生成的路网进行人工优化。

考虑到机场的实际客流情况, 我们计划为路网添加一个根据该路径上的人流密度等因素决定的权值, 机场管理者可以根据需要对齐进行设置。这样做的好处在于进行路径规划时可以优先保证用户通行的畅通, 避免让用户感到软件规划出的路径难走, 同时也减少了机场拥堵的发生。

此外, 我们还计划在生成的路径之外人工规划出一些主要的路径, 例如从出入口到值机柜台再到登机口的路径。人工规划出的路径具有以下特点:

1. 用户沿该路径行走时不经过客流量大的区域 (如出入口、商店附近等), 保证通行的畅通;
2. 用户沿该路径行走时不会逆行, 避免与其他旅客冲撞;
3. 用户沿该路径行走能尽可能的节约时间。