

程序设计与算法综合训练

实验报告

实验名称：旅游景点咨询系统的设计与实现

专业班级：人工智能二班

学号：WA2214014

姓名：杨跃浙

目录

一、实验内容及要求	3
1.1 实验目的	3
1.2 实验内容	3
1.3 实验要求	3
1.4 实验任务	4
二、任务 1：基于所给的旅游景点图、路径及权重——底层开发	4
2.1 问题描述	4
2.2 基础内容	5
2.2.1 图	5
2.2.2 最短路径算法	5
2.3 景点图最短路径的程序	6
2.4 程序运行结果	8
三、任务 2：ArcGIS 最短路径分析——应用操作	8
3.1 问题描述	8
3.2 ArcGIS 软件的了解	8
3.2 地图获取	9
3.3 最短路径分析部分操作截图	12
3.4 分析结果	19
四、任务 3 基于 ArcGIS API 实现 web 端地图最短路径分析——二次开发	19
4.1 任务 1	19
4.1.1 Arcgis server 环境安装	19
4.1.2 Arcgis server 地图发布；	24
4.1.3 基于 ArcGIS API 实现 web 端地图最短路径分析	24
4.2 任务 3：参考 Arcgis API 帮助文档，实现百度地图的附近搜索美食	29
4.2.1 准备工作	29
4.2.2 关于 ArcGIS API for JavaScript 的相关代码实现	31
4.2.3 尝试修改目标	32
五、实验总结	35

一、实验内容及要求

1.1 实验目的

加深对图状结构的理解，掌握最短路径的计算方法，强化学生的逻辑思维能力和动手能力。创建有向网表示的旅游景点导游图

1.2 实验内容



原理理解

- 1、创建一个有向图表示的黄山旅游景点的导游图。
- 2、输入两个景点名，就可以得到其最短路径，如果两者无路径可通，就得出“两景点不可达的信息”。

1.3 实验要求

完成任务 1：基于所给的旅游景点图、路径及权重--底层开发—4 学时+2 学时

完成任务 2：ArcGIS 最短路径分析—应用操作—2 学时+2 学时（课外）

完成任务 3：基于 ArcGIS API 实现 web 端的地图最短路径分析—二次开发-4 学时

1.4 实验任务

任务 1：基于所给的旅游景点图、路径及权重--底层开发

任务 2：ArcGIS 最短路径分析—应用操作

任务 3：基于 ArcGIS API 实现 web 端的地图最短路径分析—二次开发

二、任务 1：基于所给的旅游景点图、路径及权重—底层开发

2.1 问题描述

任务 1：基于所给的旅游景点图、路径及权重。--底层开发

编程实现：输入两个景点名，计算最短路径、最短路径长度。

旅游景点图



路径及权重

```
玉屏索道入口 -> 玉屏索道出口 6
玉屏索道出口 -> 天都峰 6
玉屏索道出口 -> 迎客松 1
天都峰 -> 迎客松 2
迎客松 -> 排云溪站 12
排云溪站 -> 探海亭 1
探海亭 -> 排云亭 2
排云溪站 -> 排云亭 2
排云亭 -> 光明顶 2
光明顶 -> 白鹤岭 1
排云亭 -> 白鹤岭 2
排云亭 -> 始信峰 2
排云亭 -> 松谷庵站 9
松谷庵站 -> 始信峰 9
白鹤岭 -> 云谷索道入口 1
云谷索道入口 -> 始信峰 2
始信峰-> 云谷索道入口 2
云谷索道入口 -> 云谷索道出口 5
```

2.2 基础内容

2.2.1 图

```
//图的结构体
typedef struct {
    char Vertex[MAXSIZE];      //顶点名称
    int Matrix[MAXSIZE][MAXSIZE]; //邻接矩阵
    int EdgeNum, VexNum; //边的个数和顶点的个数
}MGraph;
```

2.2.2 最短路径算法

2.2.2.1 迪杰特斯拉算法

Dijkstra 算法是一种单源最短路径算法，用于计算从一个源点到其他所有点的最短路径。

初始化

visit 数组全为 0, dist 数组全为 INF, path 数组全为-1

第 1 步:

- 1) 出发点是第一个已确定最短路径的顶点 v_i
- 2) 设置 $visit[i] = 1$, 代表已访问
- 3) 更新与出发点有路径的顶点所对应的最短路径和前驱顶点。

第 2 步-第 n 步开始循环:

- 1) 从 $dist$ 数组中选取距离最短的并且在 $visit$ 数组中数值为 0 (未被访问过) 的一个顶点作为中间节点 v_j 。设置 $visit[j] = 1$ 。判断: 通过中间节点 v_j 能否实现到未访问的顶点距离更短。
- 2) 若更短, 则更新 $dist$ 数组和 $path$ 数组。

2.2.2.2 弗洛伊德算法

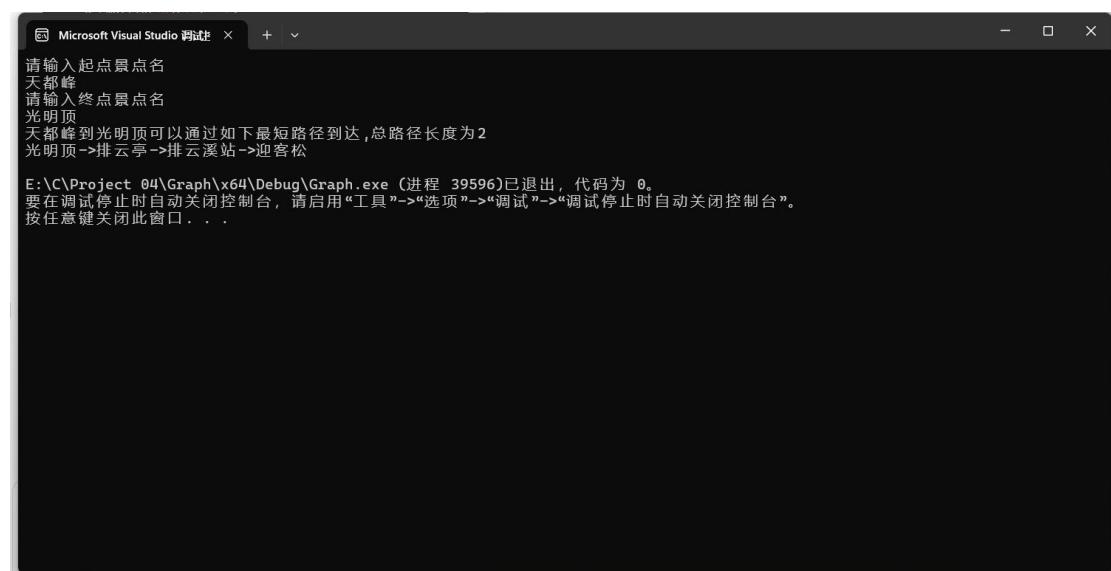
Floyd 算法是一种多源最短路径算法, 用于计算任意两点之间的最短路径。

Floyd 算法采用动态规划的思想, 分多个阶段解决问题。若图 G 有 n 个顶点 ($V_1 \sim V_n$), 求图中每一对顶点之间的最短路径分 n 个阶段:

- 0) 初始化, 在没有其它顶点中转的情况下, 求得各顶点间的最短路径。
 - 1) 如果在各顶点间增加 V_1 作为中转点, 求得各顶点间新的最短路径。
 - 2) 再增加 V_2 作为中转点, 求得各顶点间新的最短路径。
-
- n) 最后增加 V_n 作为中转点, 求得各顶点间最终的最短路径。

2.3 景点图最短路径的程序

程序调试过程中, 发现一开始不能正确输出路径和长度



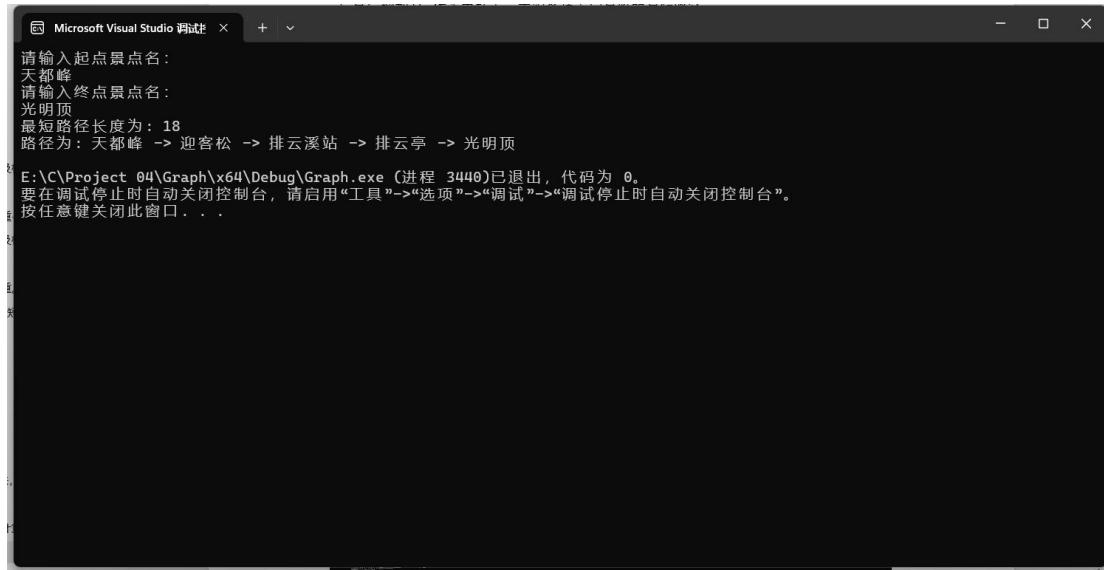
修改最短路径求解代码

```

void dijkstra_method(MGraph* g, int dist[MAXSIZE], int path[MAXSIZE], int v0) {
    bool visited[MAXSIZE] = { false };
    for (int i = 0; i < g->VexNum; ++i) {
        dist[i] = g->Matrix[v0][i];
        if (dist[i] < INF) {
            path[i] = v0;
        }
        else {
            path[i] = -1;
        }
    }
    visited[v0] = true;
    dist[v0] = 0;
    for (int i = 1; i < g->VexNum; ++i) {
        int min_dist = INF;
        int u = -1;
        for (int j = 0; j < g->VexNum; ++j) {
            if (!visited[j] && dist[j] < min_dist) {
                u = j;
                min_dist = dist[j];
            }
        }
        if (u == -1) break;
        visited[u] = true;
        for (int j = 0; j < g->VexNum; ++j) {
            if (!visited[j] && g->Matrix[u][j] < INF) {
                int new_dist = dist[u] + g->Matrix[u][j];
                if (new_dist < dist[j]) {
                    dist[j] = new_dist;
                    path[j] = u;
                }
            }
        }
    }
}

```

2.4 程序运行结果



The screenshot shows a Microsoft Visual Studio Debug window. The title bar says "Microsoft Visual Studio 调试". The main area contains the following text:
请输入起点景点名：
天都峰
请输入终点景点名：
光明顶
最短路径长度为：18
路径为：天都峰 -> 迎客松 -> 排云溪站 -> 排云亭 -> 光明顶
E:\C\Project_04\Graph\x64\Debug\Graph.exe (进程 3440)已退出，代码为 0。
要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口...

此时输出是正确的

三、任务 2：ArcGIS 最短路径分析—应用操作

3.1 问题描述

任务 2：ArcGIS 最短路径分析—应用操作

- 1) 查阅资料、了解 ARCGIS;
- 2) 查阅资料，练习 ARCGIS 的最短路径分析

3.2 ArcGIS 软件的了解

ArcGIS 是由美国 Esri 公司开发的地理信息系统 (GIS) 软件套件。它提供了一整套用于创建、编辑、分析和展示地理空间数据的工具和功能。ArcGIS 的核心组成部分包括：

ArcGIS Pro：是 ArcGIS 的最新桌面应用程序，提供了先进的三维地图制作、数据编辑、空间分析和地图制图等功能。

ArcMap: 是 ArcGIS 的传统桌面应用程序，提供了广泛的地图制作和空间分析工具，被许多 GIS 专业人员广泛使用。

ArcGIS Online: 是一种基于云的 GIS 平台，允许用户在 Web 浏览器中访问、创建和共享地图和 GIS 数据，并与其他用户进行协作。

ArcGIS Enterprise: 是 Esri 公司提供的企业级 GIS 解决方案，允许组织在自己的基础设施上构建、部署和管理 GIS 应用程序和服务。

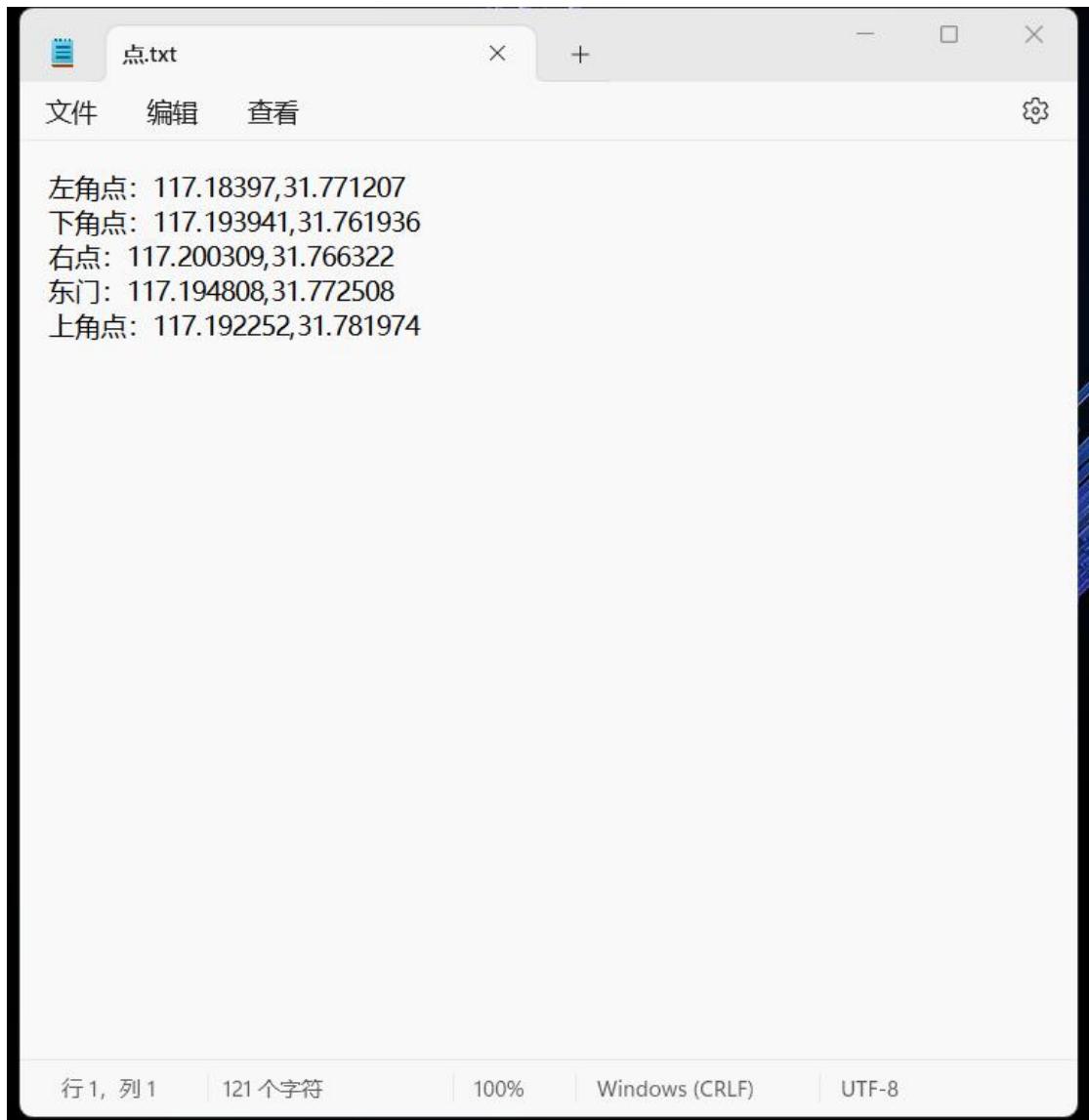
ArcGIS 软件可以用于各种领域，包括自然资源管理、城市规划、环境保护、应急响应、商业分析等。它被广泛应用于政府部门、科研机构、企业以及教育机构等领域，成为了全球最流行的地理信息系统软件之一。

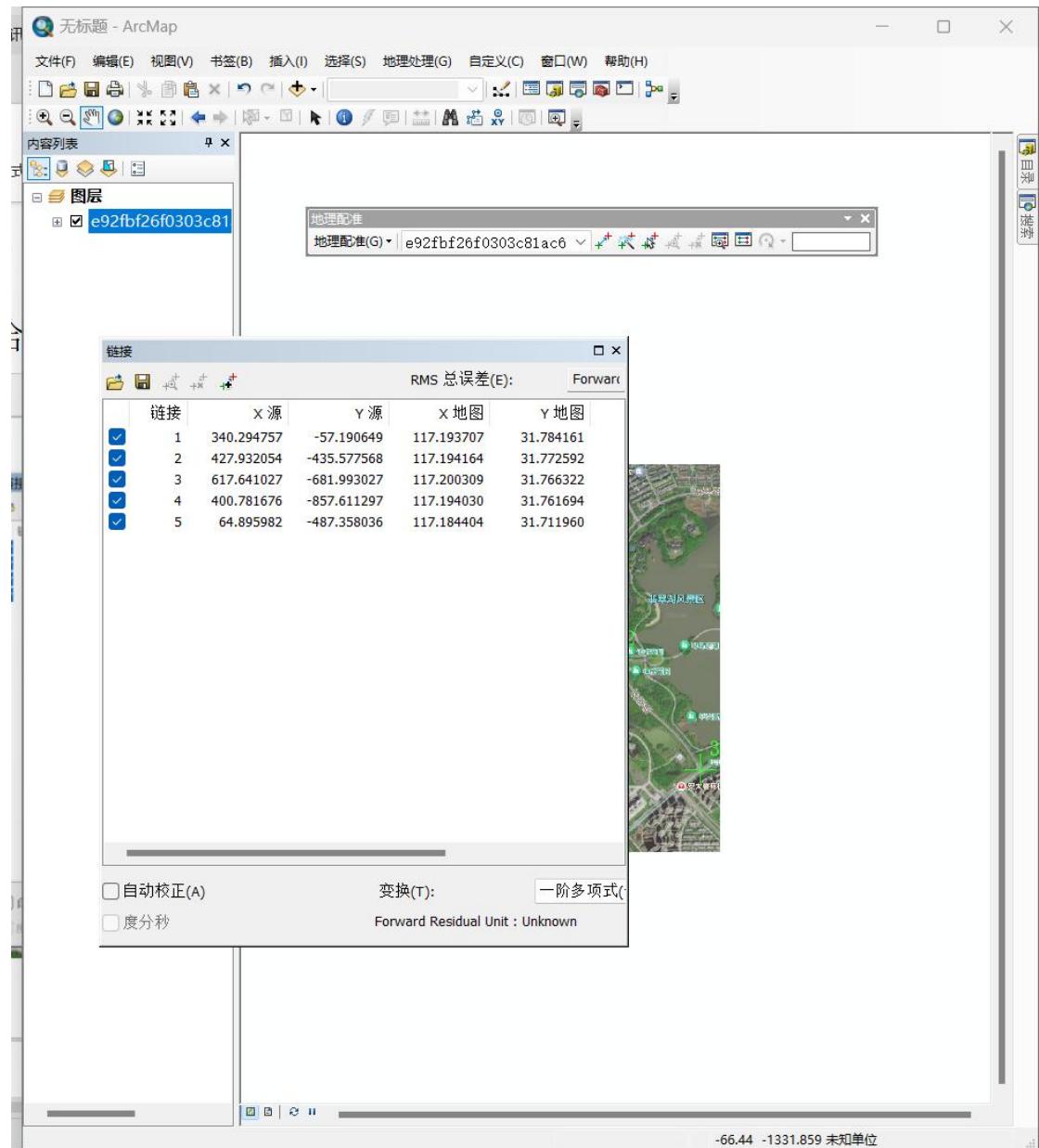
3.2 地图获取

3.2.1: 百度底图上找到安大新区位置，切换至遥感影像模式截图保存



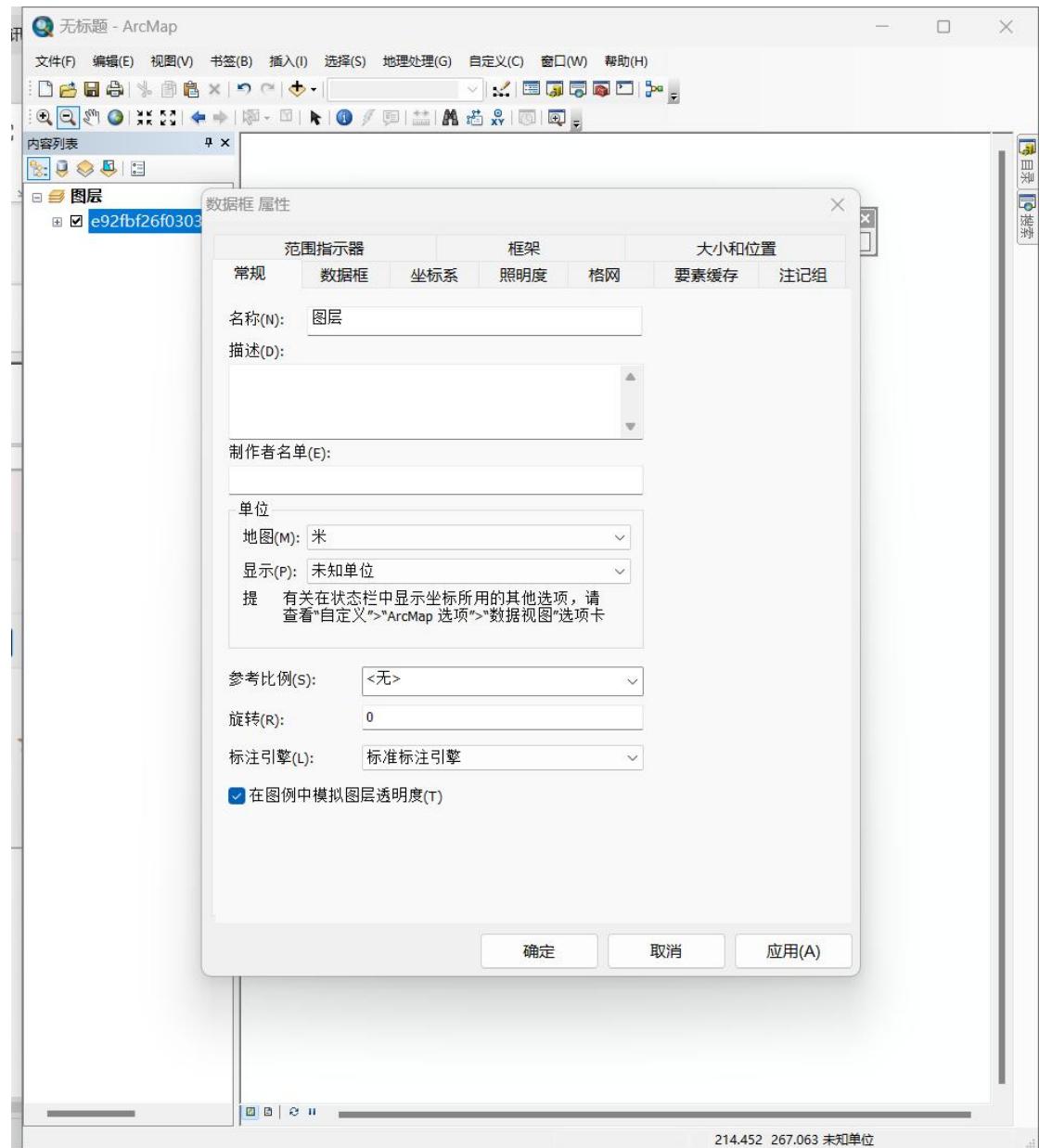
2.2 配置地理坐标

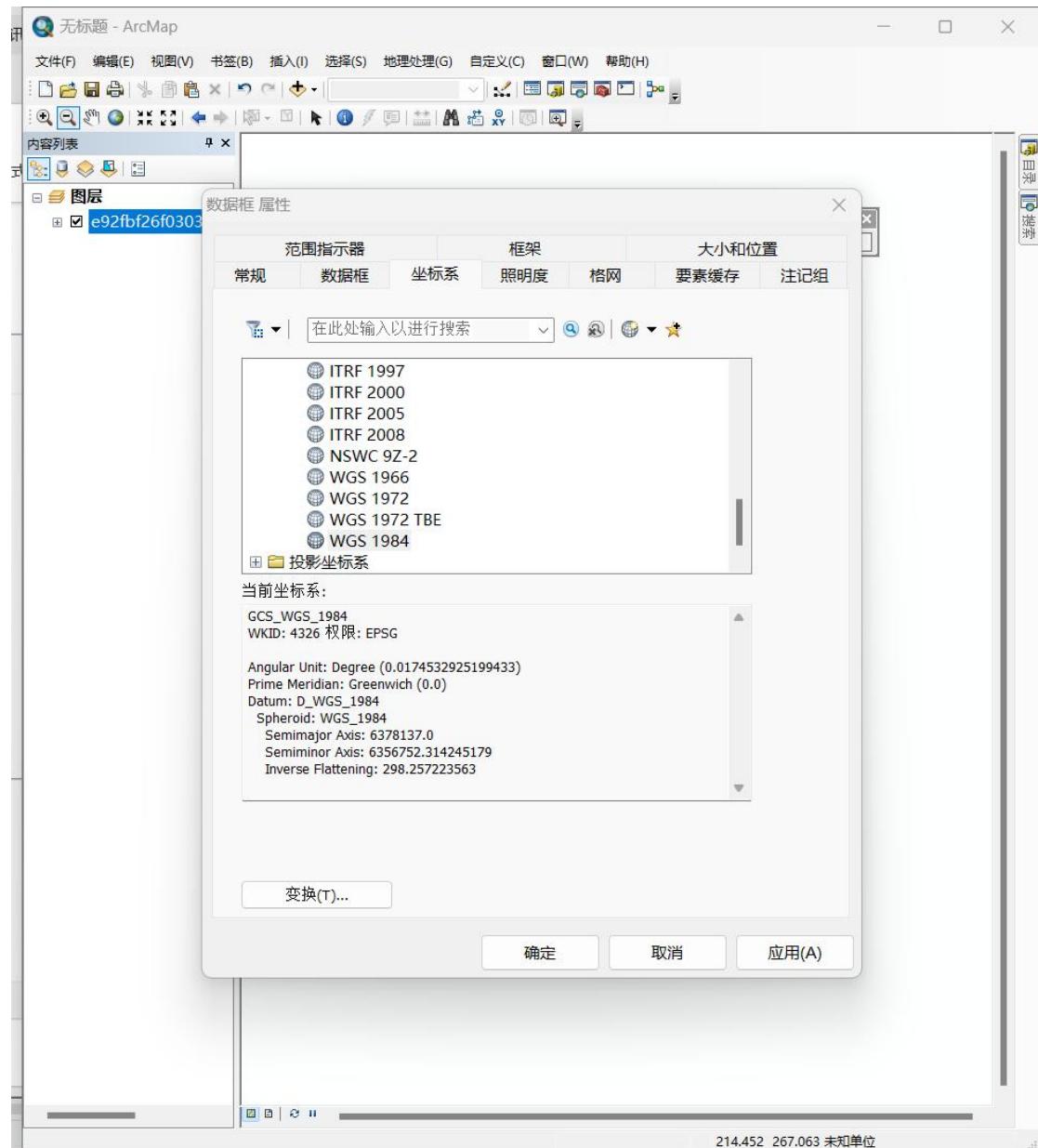


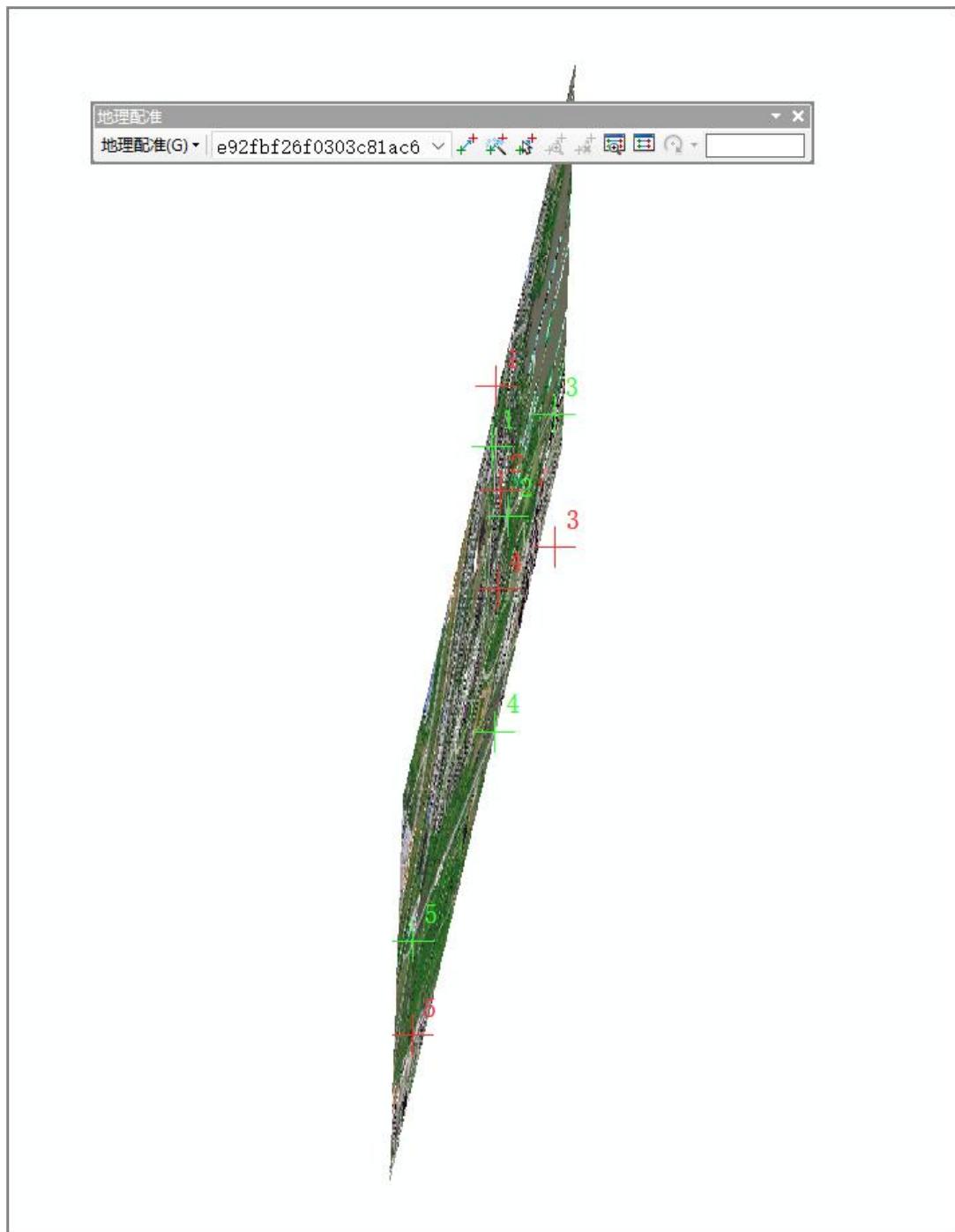


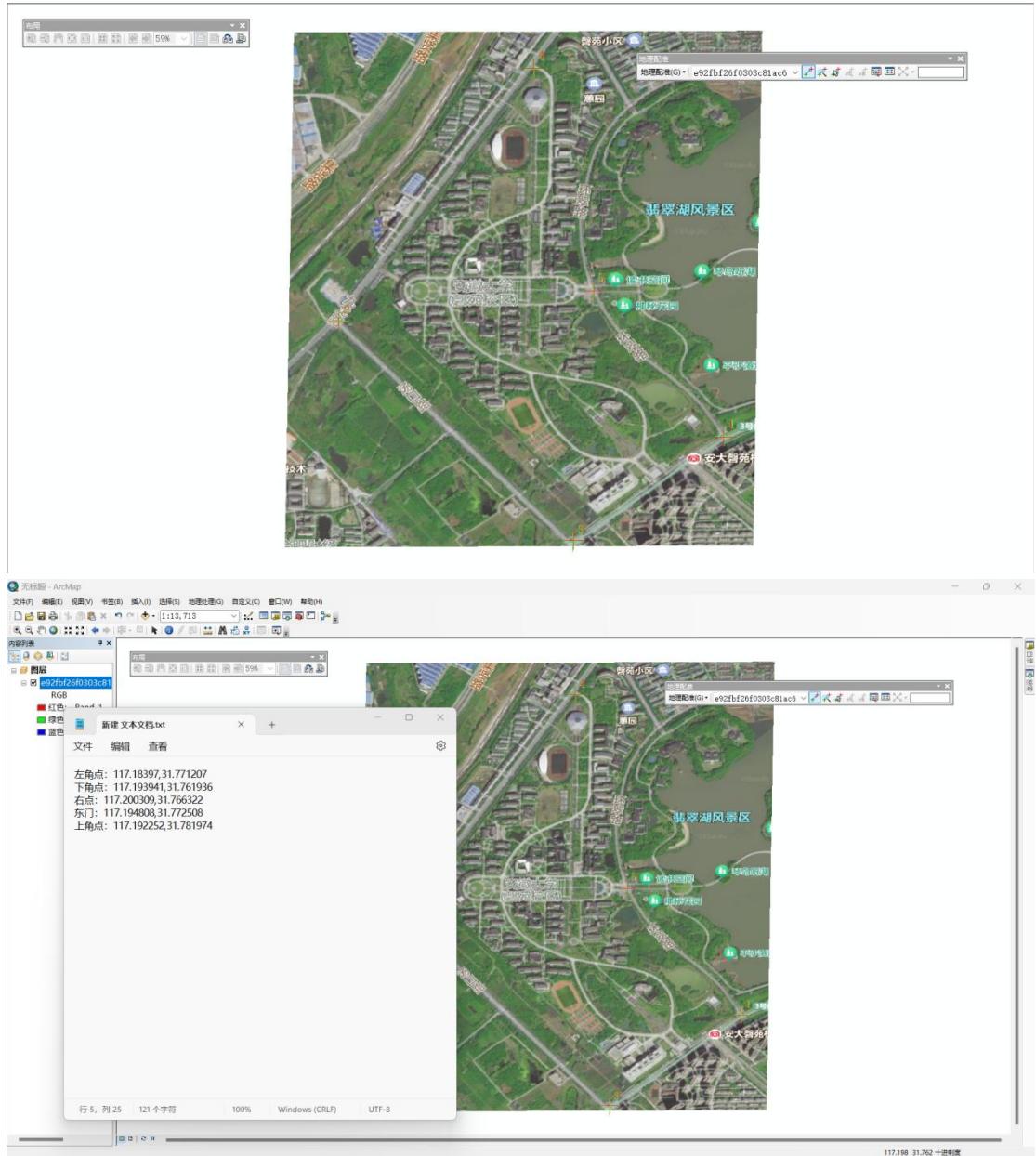
3.3 最短路径分析部分操作截图

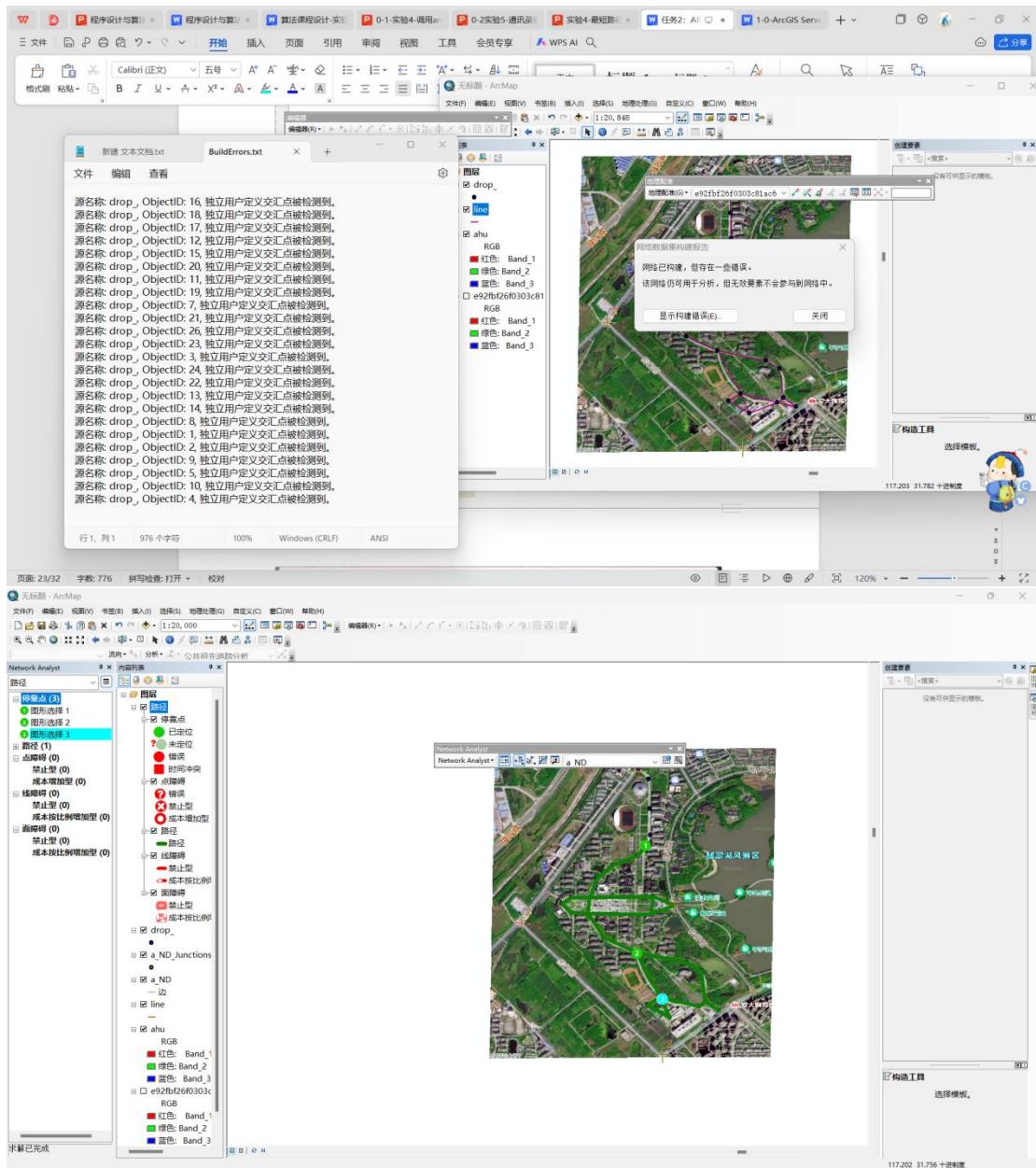
操作截图：

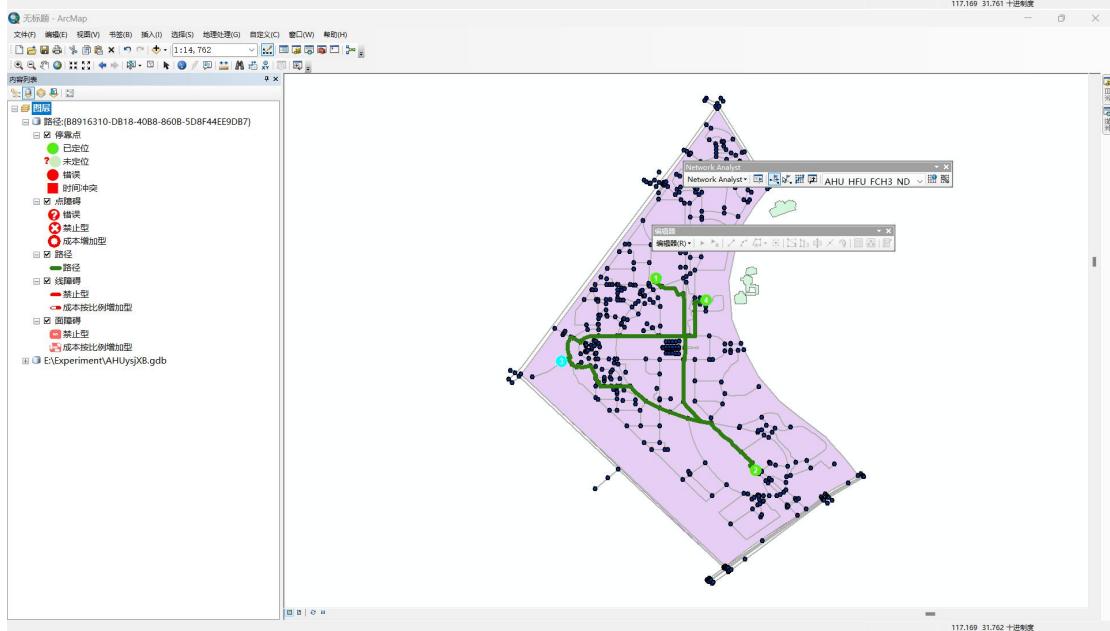
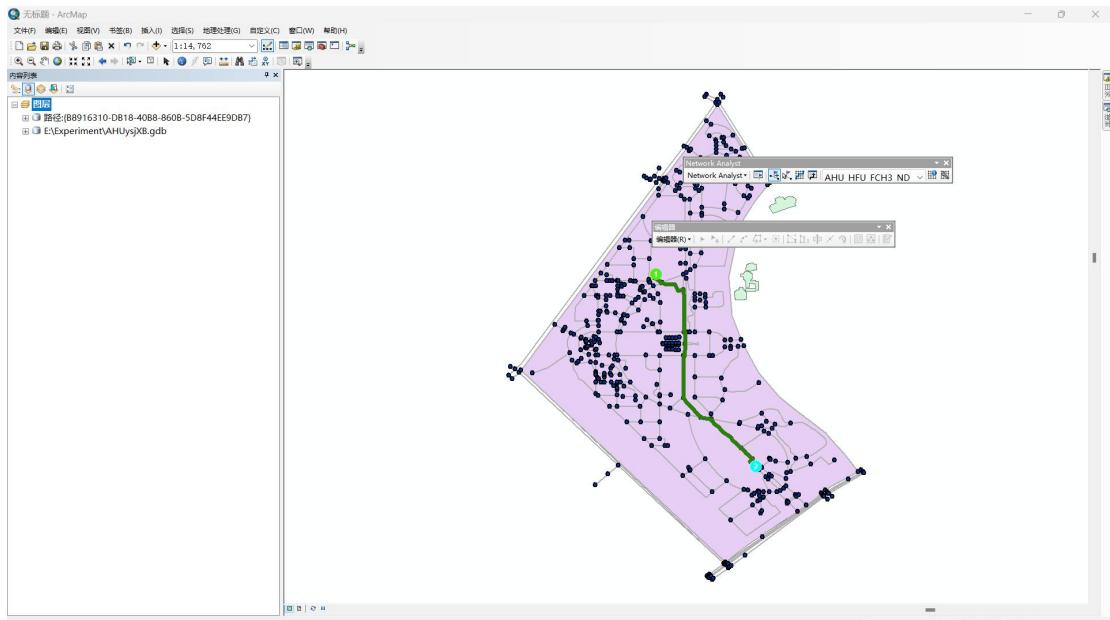


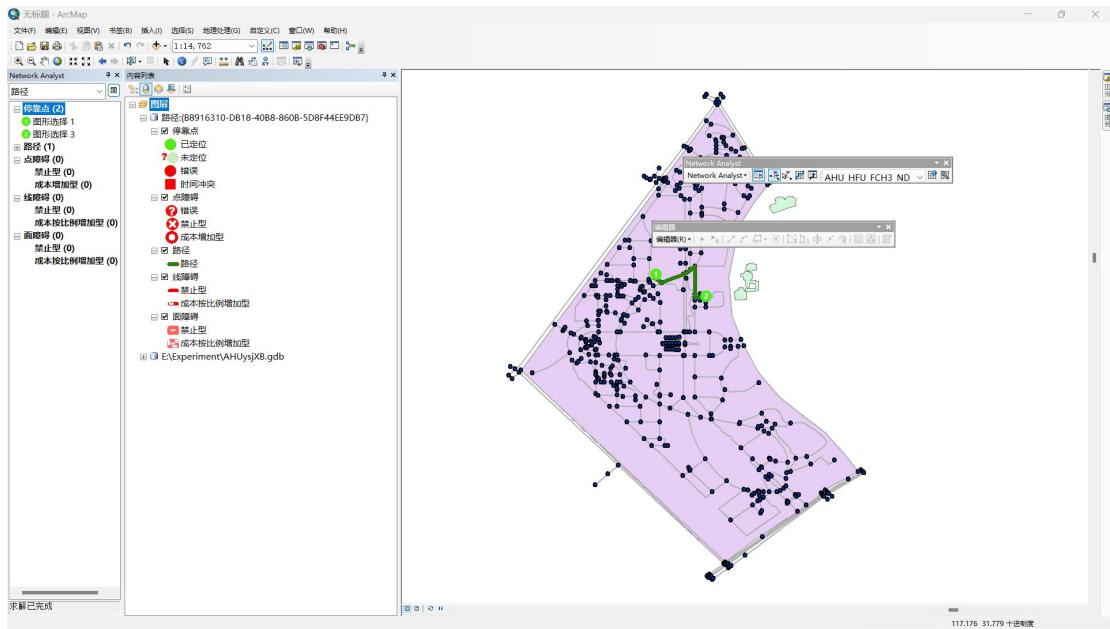












3.4 分析结果

在操作过程中，发现如果选点出现偏差，最终的操作结果将会有很大的差异，所以选点极为重要。在操作过程中，我也发现，为了能很好的求解出最短路径，建立一个合适的路网也是很重要的。最后成功实现了最短路径的求解。

四、任务3 基于ArcGIS API实现web端地图最短路径分析—二次开发

4.1 任务1

4.1.1 Arcgis server 环境安装

4.1.1.1 了解 Arcgis server

ArcGIS Server 是 Esri 公司开发的一款基于服务器的 GIS 软件，它允许用户在企业或组织内部部署 GIS 服务和应用程序。ArcGIS Server 提供了一系列功能强大的服务，包括地图服务、地理处理服务、空间分析服务等，这些服务可以通过网络提供给用户使用。以下是 ArcGIS Server 的主要特点和功能：

地图服务： ArcGIS Server 可以将地图数据发布为地图服务，用户可以通过 Web 浏览器或 GIS 应用程序访问这些地图服务，查看地图数据、进行地图交互操作，并与其他数据进行集成和分析。

地理处理服务： ArcGIS Server 提供了丰富的地理处理工具和功能，用户可以将这

些工具封装成地理处理服务，用于自动化地理空间数据的处理和分析。

空间分析服务：ArcGIS Server 支持空间分析功能，用户可以通过网络访问这些空间分析服务，进行缓冲区分析、空间查询、网络分析等操作。

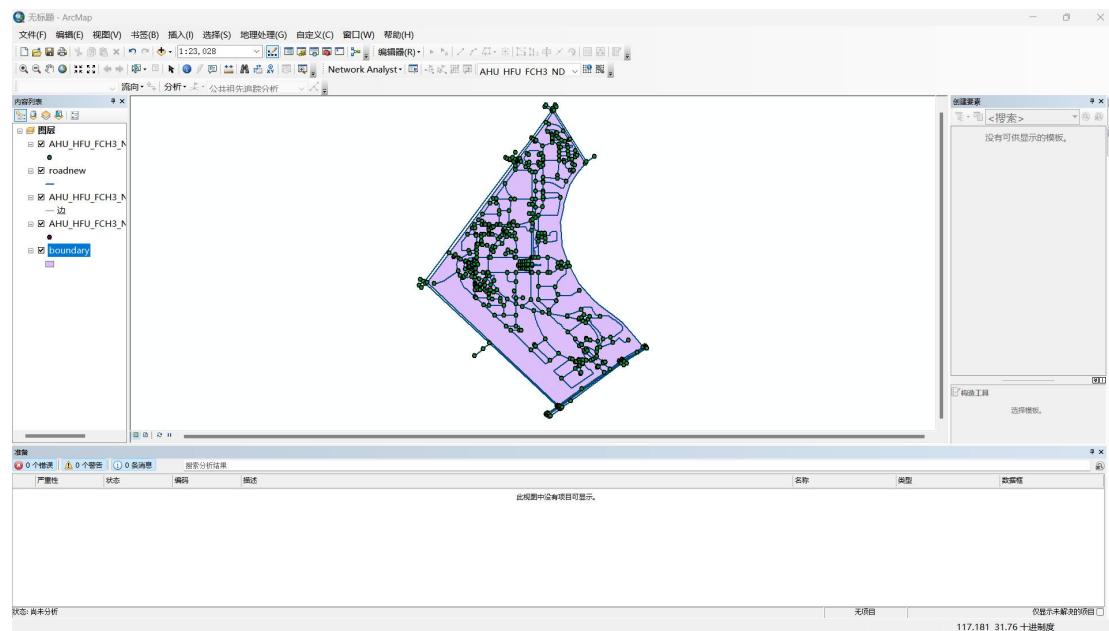
安全性和权限控制：ArcGIS Server 提供了完善的安全性和权限控制机制，用户可以设置访问控制、身份验证和授权策略，确保地理空间数据的安全性和保密性。

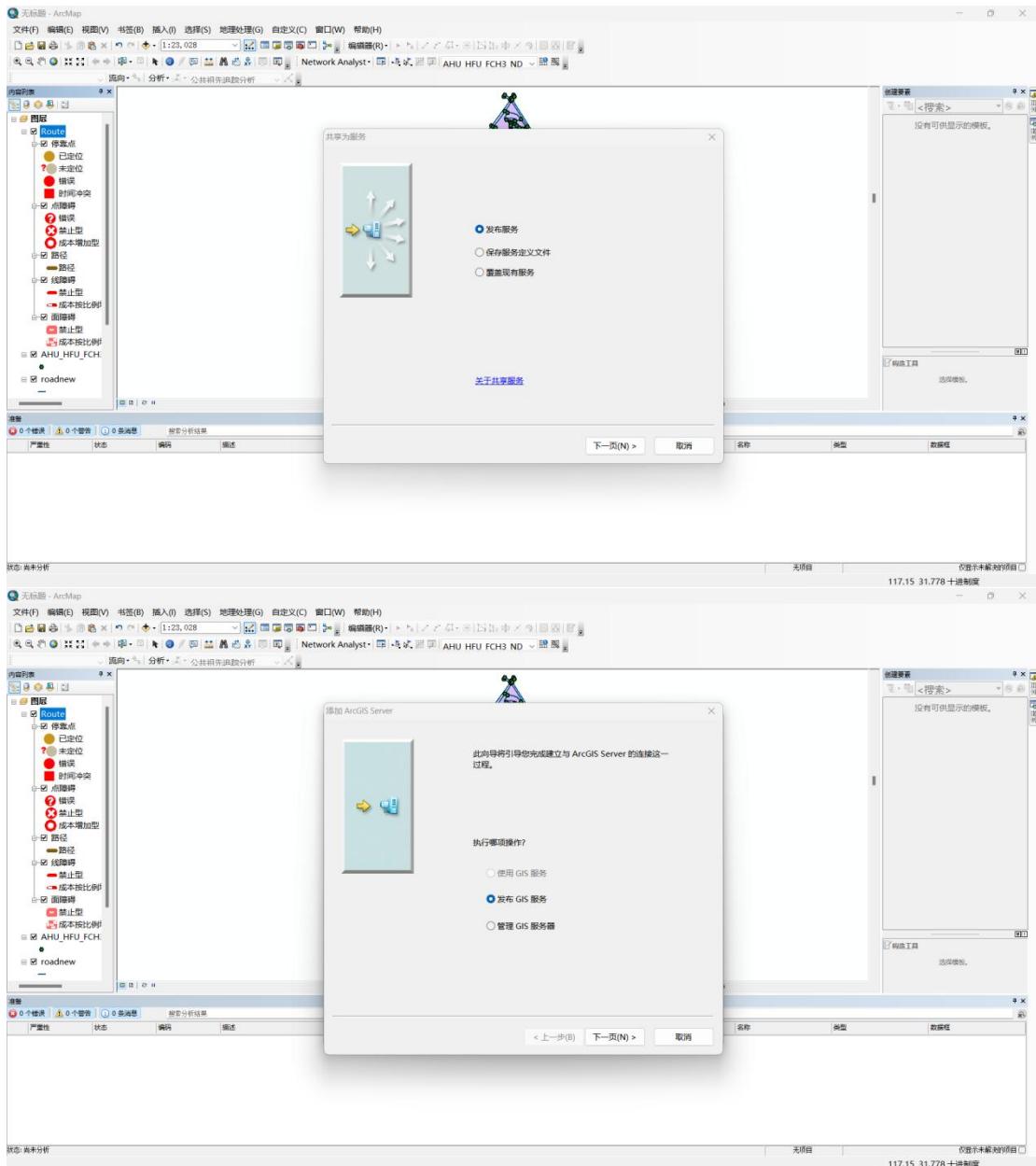
可伸缩性和性能优化：ArcGIS Server 具有良好的可伸缩性和性能优化能力，支持大规模地图数据的存储、处理和传输，可以满足不同规模和需求的 GIS 应用场景。

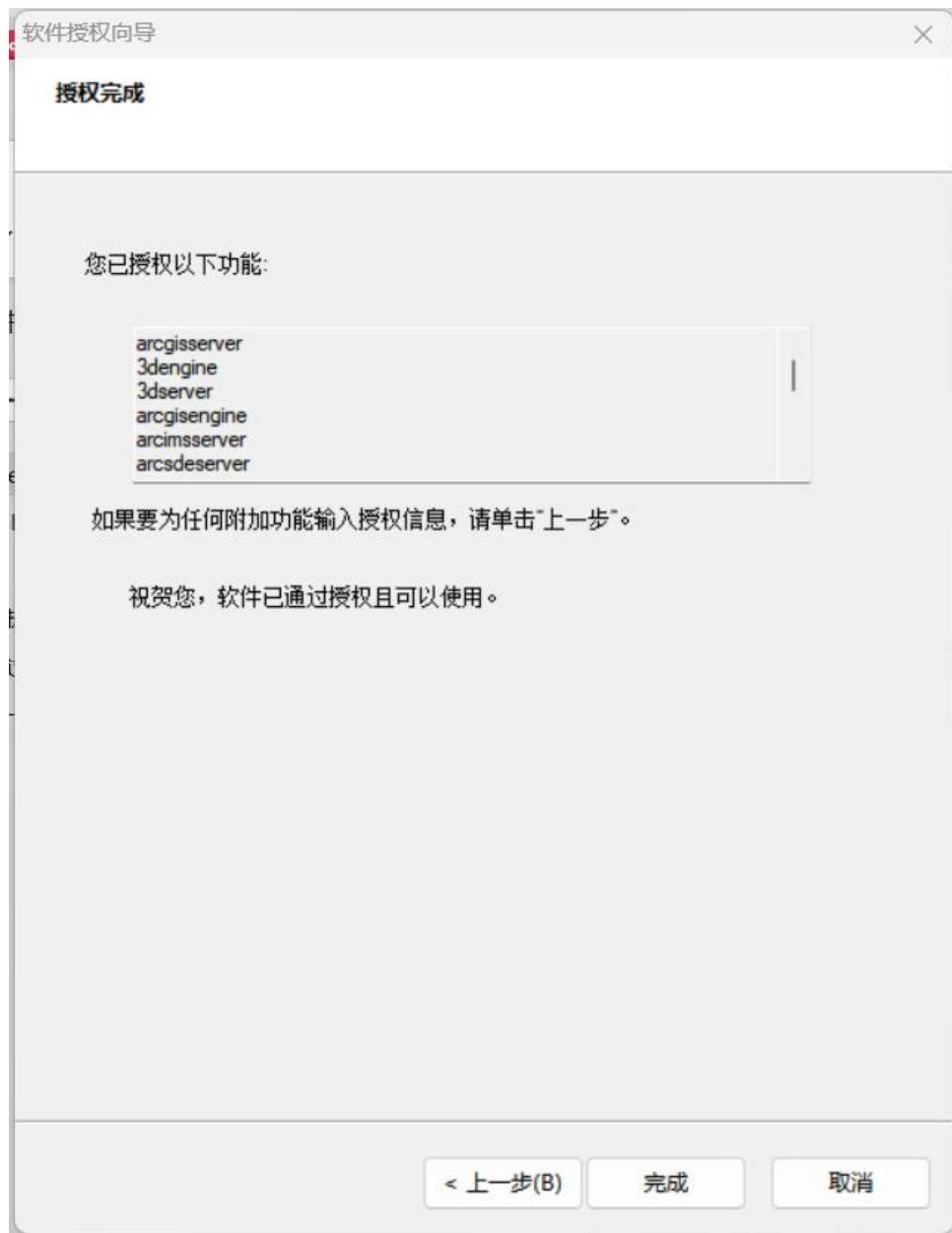
可定制化和扩展性：ArcGIS Server 提供了丰富的 API 和开发工具，用户可以根据自己的需求定制和扩展 GIS 应用程序，实现更多样化的地理空间数据处理和展示功能。

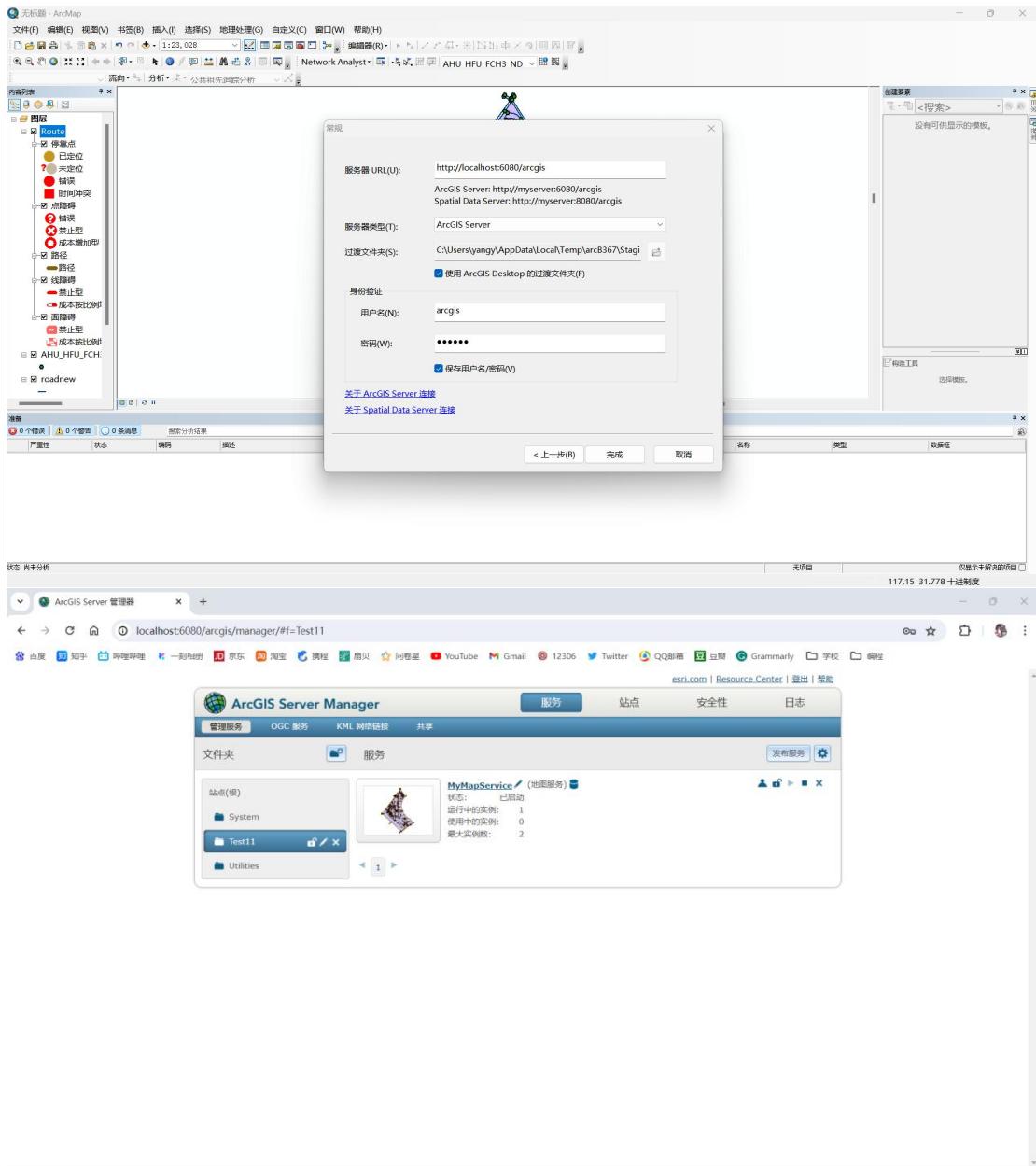
ArcGIS Server 是一款功能强大、灵活性高的 GIS 服务器软件，可以帮助用户构建、部署和管理各种类型的 GIS 服务和应用程序，实现地理空间数据的有效管理和利用。

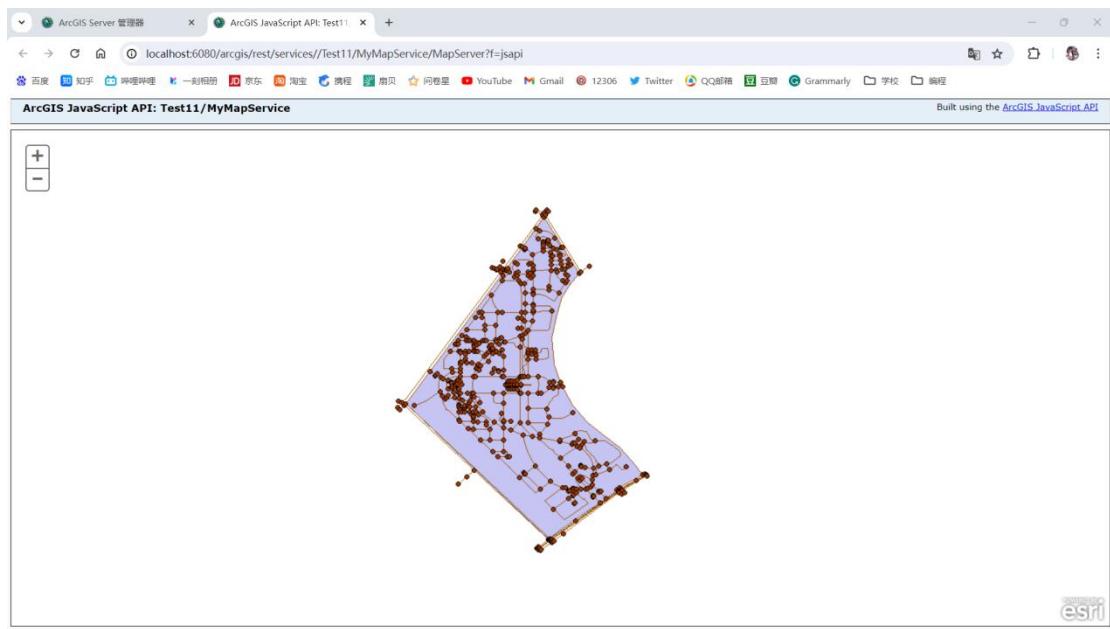
4.1.1.2 操作截图



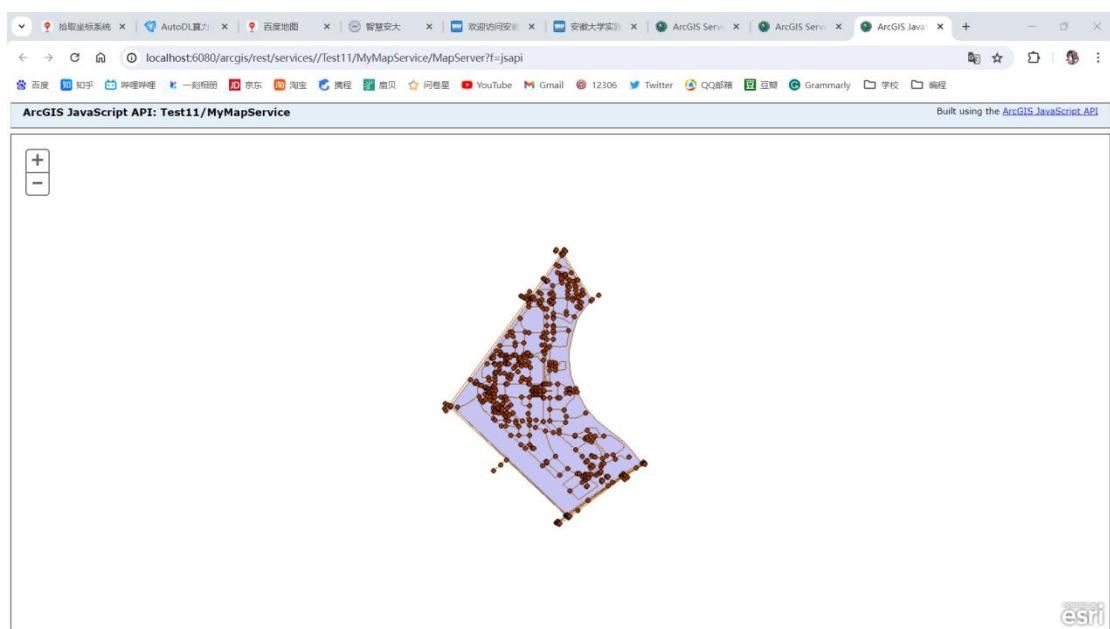








4.1.2 Arcgis server 地图发布;



4.1.3 基于 ArcGIS API 实现 web 端地图最短路径分析

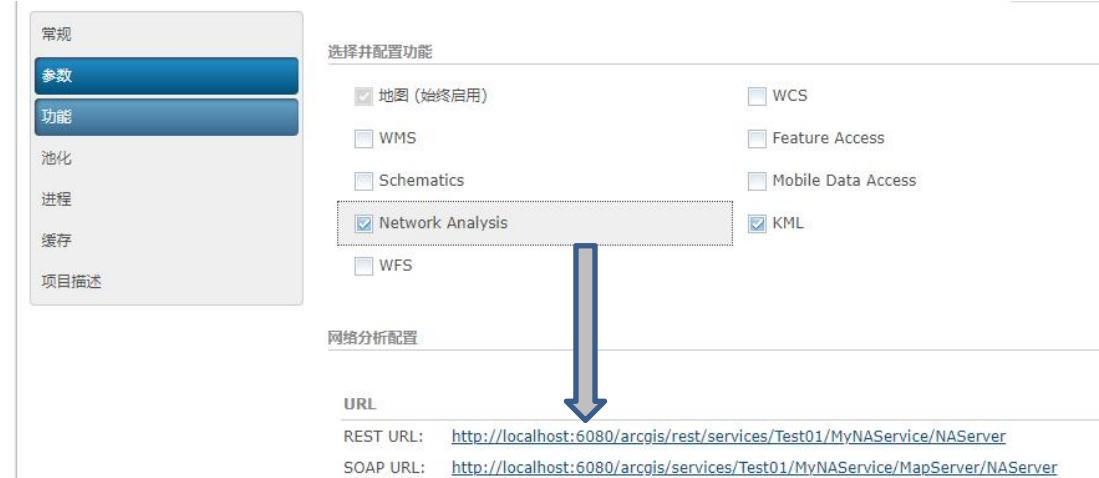
4.1.3.1 创建 html 文件

按照要求修改路径

下面的代码不要直接用，其中 **url** 的要修改，具体修改方式为：点击自己发布的服务，然后选择功能，复制其中的 REST URL，把 localhost 修改为自己的计算机名（右键“我的电脑”，属性，复制设备名称放到 localhost 的位置）



还有 `routeUrl` 也要修改，修改方式类似上一步，点击自己发布的服务，然后选择功能，点击 Network Analysis，复制其中的 REST URL，localhost 修改同上，与上一步修改不同的是，修改完这些之后，要在最后面加上“/图层名”（图层名就是在“2-0 地图发布教程”中 2.2.2 让你修改的英文名称）



修改完成：

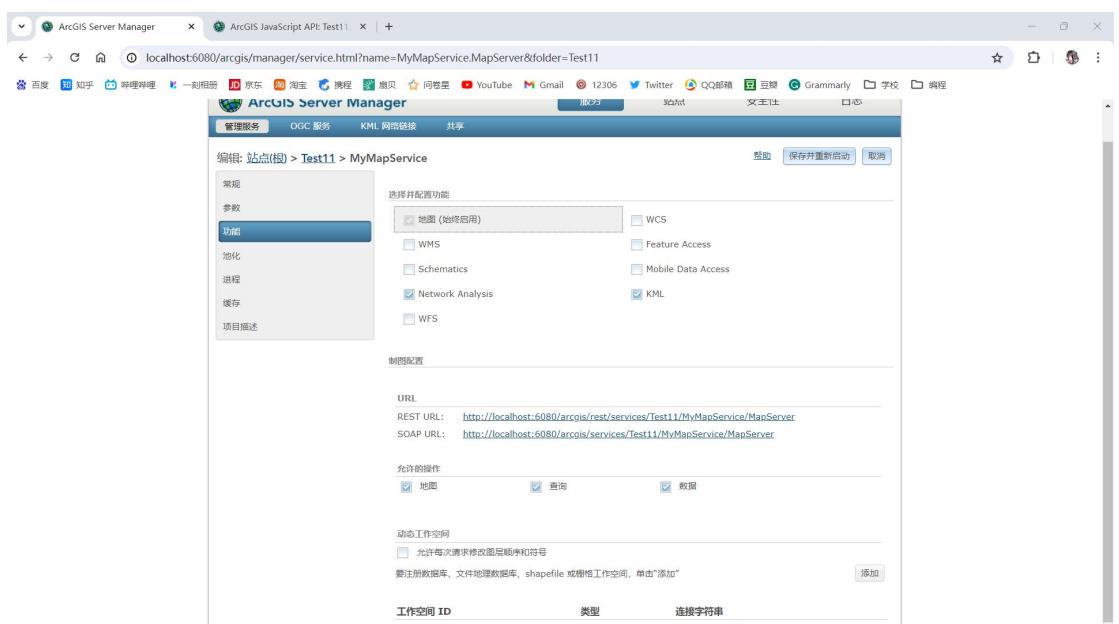
```

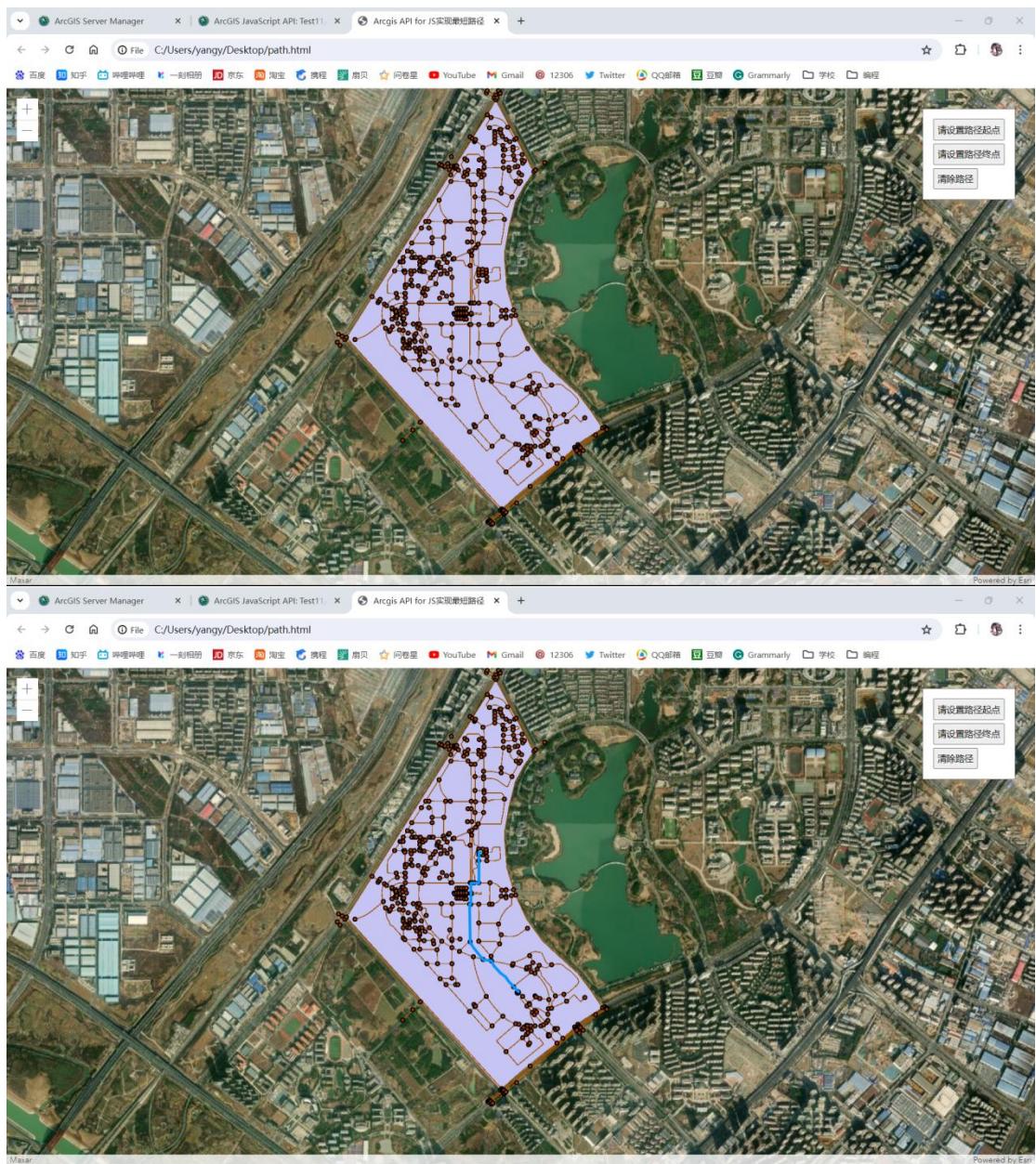
70   GraphicsLayer,
71   }
72   // 加载图层, GraphicsLayer是一种客户端图层（根据坐标以及形状生成相应的图形然后在地图显示出来），为了把我们的图层加载进去，看起来更方便——
73   const routeLayer = new GraphicLayer();
74
75   // 我们自己的图层，这个url可以在我们发布的服务中点击服务的图像查看（把后面的?f=json删掉），还可以在服务-功能-底图中复制REST URL——
76   const layer = new MapImageLayer({
77     url: "http://YoungBean:6080/arcgis/rest/services/Test11/MyMapService/MapServer",
78   });
79
80   // 创建map，创建一个地图，为了一会在地图容器中显示出来，basemap选择底图是什么样式的
81
82   const mymap = new Map({
83     // 可选择的属性有：arcgis-navigation, topo, streets, satellite, hybrid, dark-gray, gray, national-geographic,
84     // oceans, osm, terrain, dark-gray-vector, gray-vector, streets-vector, streets-night-vector,
85     // streets-navigation-vector, topo-vector, streets-relief-vector
86     basemap: "satellite", //Basemap layer service
87     layers: [routeLayer, layer]
88   });
89
90
91   // 创建地图容器，为了在网页中能显示我们的地图 视图有两种类型：MapView和SceneView。MapView以2D方式呈现地图及其图层。SceneView将这些元素呈现为3D。View是MapView和SceneView的基类。
92   const view = new MapView({
93     container: "viewDiv",
94     map: mymap,
95     center: [-118.24532, 34.05398], //Longitude, latitude
96     center: [117.19181, 31.77169], //Longitude, latitude
97     zoom: 15 //缩放层级
98   });
99
100  // 这里是我们自己布的arcgis Server 中的路网
101  const routeUrl = "http://YoungBean:6080/arcgis/rest/services/Test11/MyMapService/NAServer/Route";
102
103
104  // 监听button，设置点去按钮并添加监听事件
105  btn_start = document.getElementById("startBtn")
106  btn_end = document.getElementById("endBtn")
107  btn_clear = document.getElementById("clearBtn")
108  btn_start.addEventListener("click", function(event) {
109    first_click = true
110  })
111
112  btn_end.addEventListener("click", function(event) {

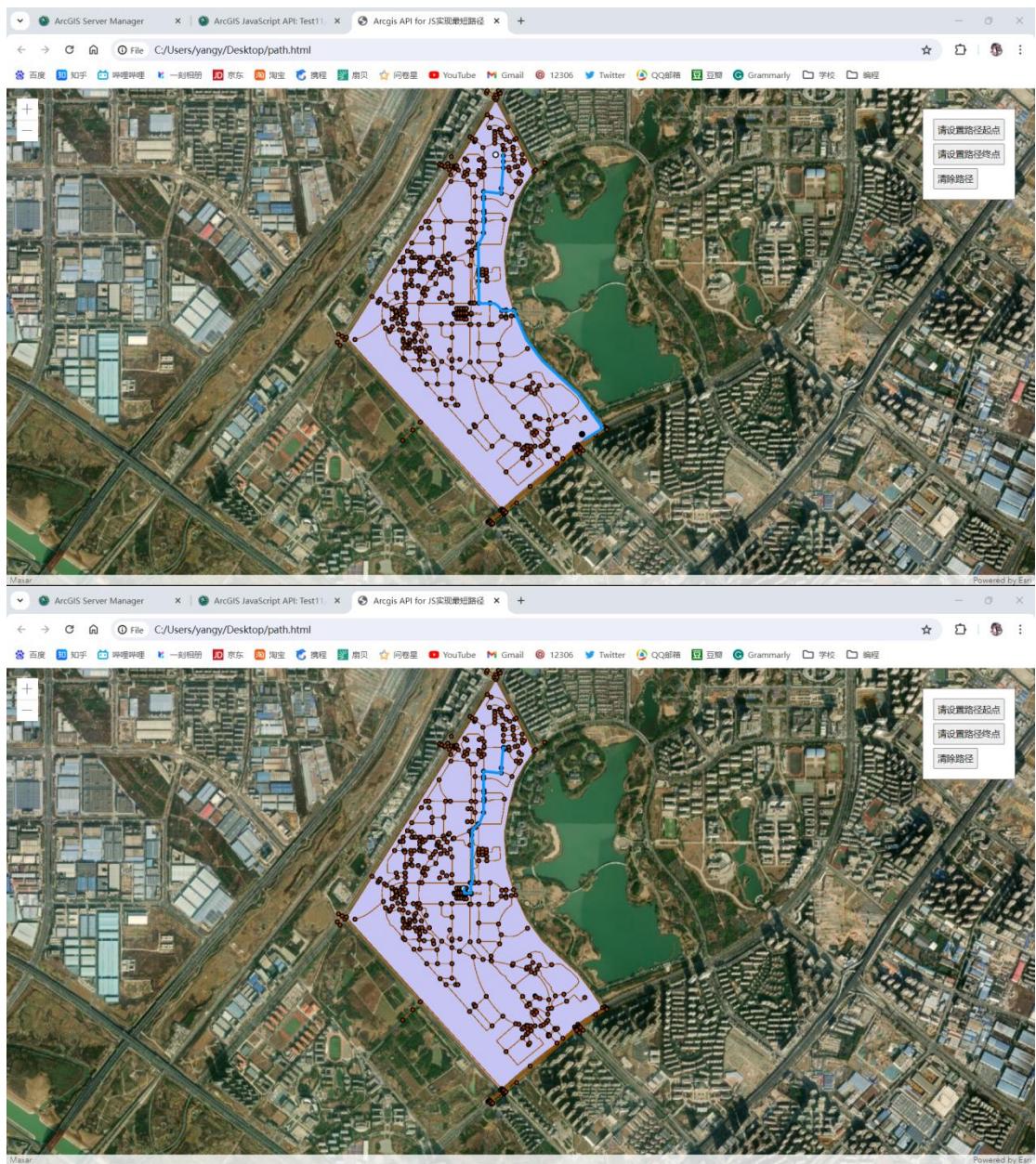
```

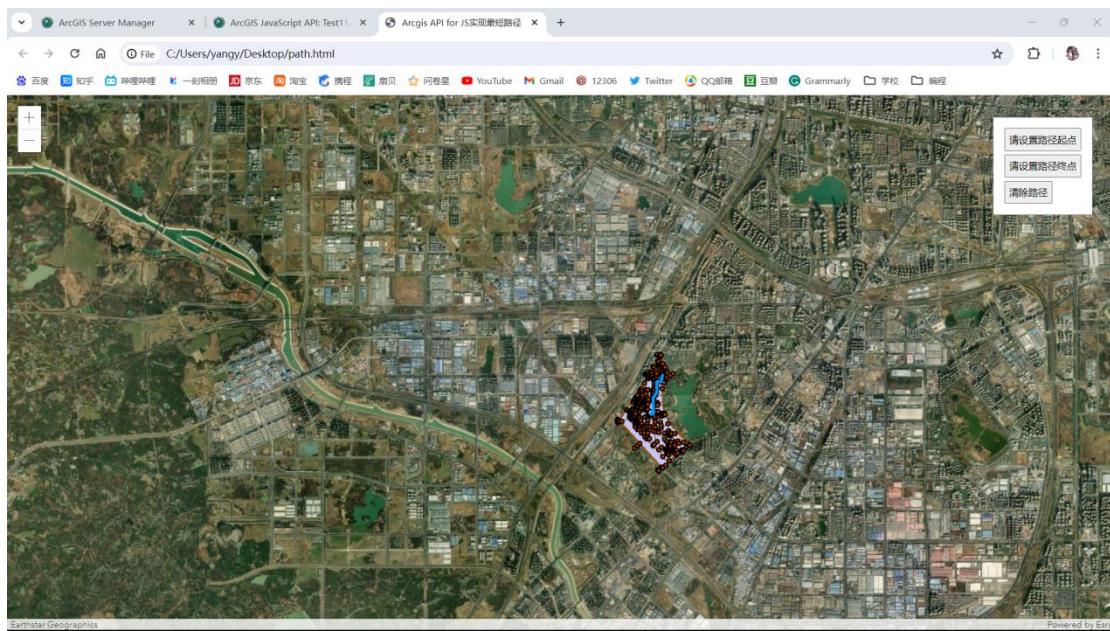
length: 8,318 lines : 189 Ln: 101 Col: 104 Pos: 4,300 Windows (CR LF) UTF-8 INS

4.1.3.2 结果截图







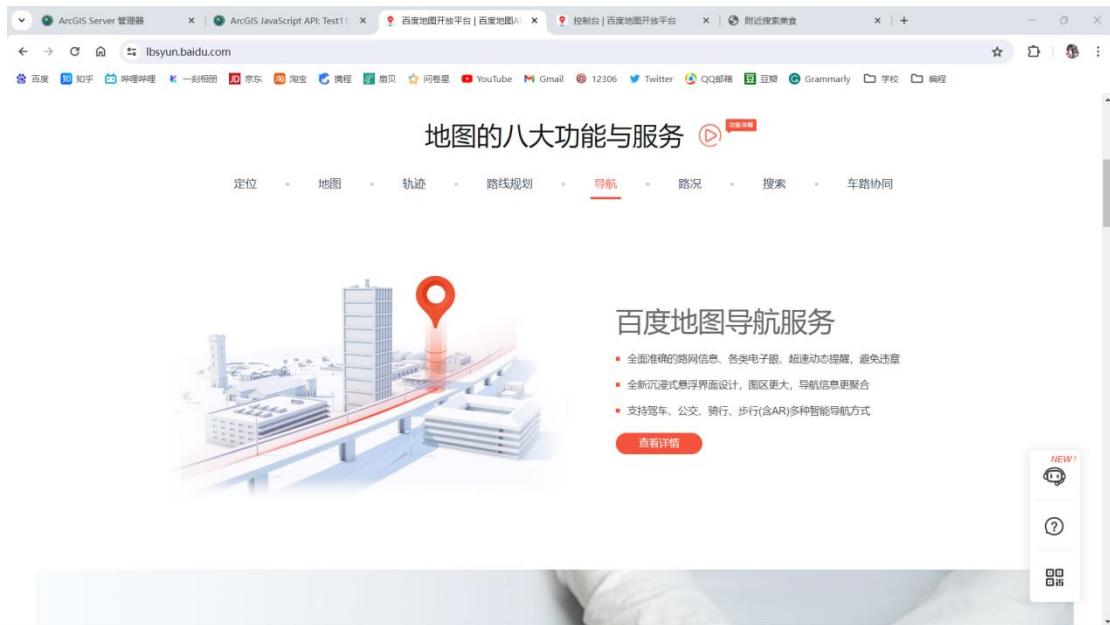


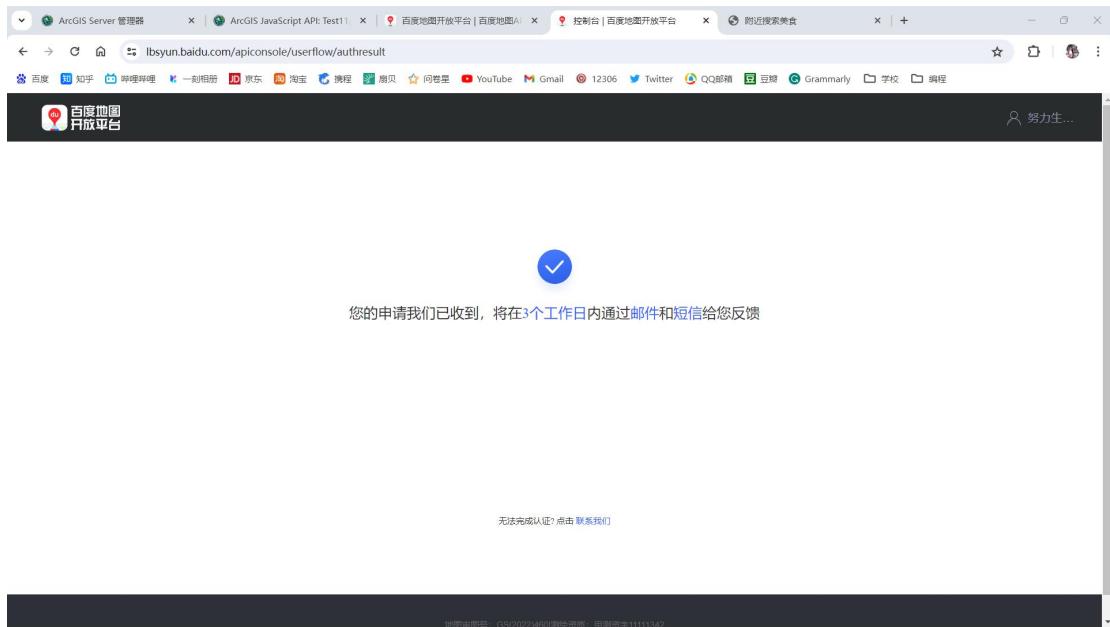
完成最短路径求解

4.2 任务 3：参考 Arcgis API 帮助文档，实现百度地图的附近搜索美食

4.2.1 准备工作

(1) 尝试使用百度地图的 JavaScript API





(2) 尝试其它方法：使用谷歌的 JavaScript API

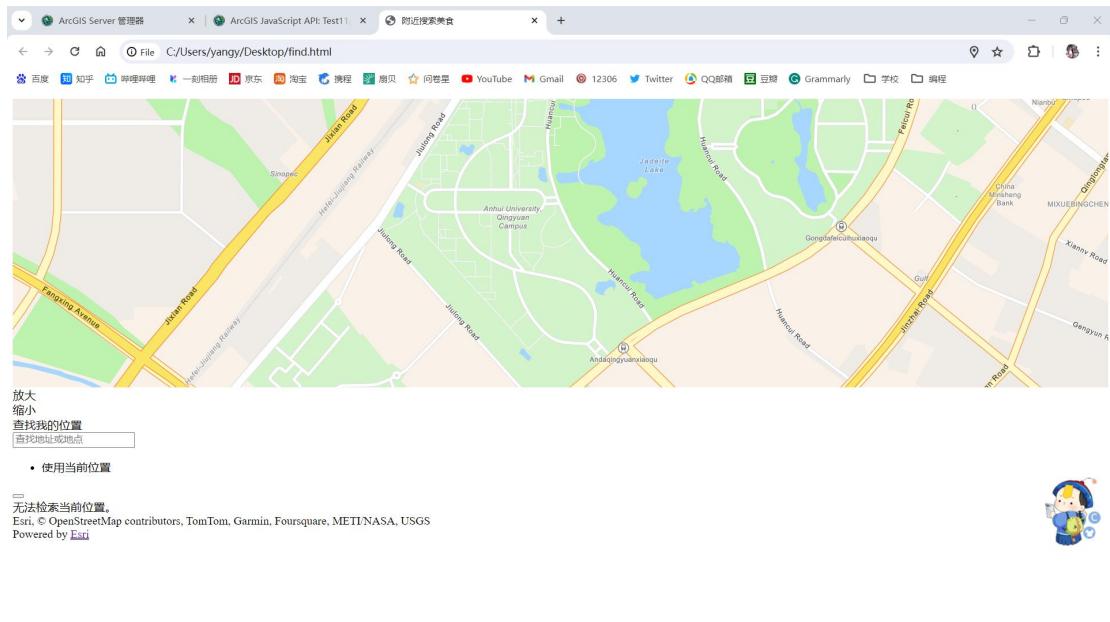
A screenshot of a Notepad++ editor showing a JavaScript file named 'find.html'. The code implements a search interface with a location button, a search input, and a nearby search function using ArcGIS services.

```

30 // 添加定位按钮
31 var locateBtn = new Locate({
32   view: view
33 });
34 view.ui.add(locateBtn, "top-left");
35
36 // 添加搜索框
37 var searchWidget = new Search({
38   view: view
39 });
40 view.ui.add(searchWidget, "top-right");
41
42 // 创建地理编码器
43 var locator = new Locator({
44   url: "https://geocode.arcgis.com/arcgis/rest/services/World/GeocodeServer"
45 });
46
47 // 添加附近搜索事件
48 view.on("click", function(event) {
49   locator.locationFromAddress(event.mapPoint)
50     .then(function(response) {
51       searchNearby(response.attributes.PlaceName);
52     })
53   });
54
55 // 附近搜索函数
56 function searchNearby(placeName) {
57   var featureLayer = new FeatureLayer({
58     url: "https://services.arcgis.com/V6ZHFr6zsdNUvG0/arcgis/rest/services/World_Food_Sites/FeatureServer/0"
59   });
60   featureLayer.definitionExpression = "CATEGORY = 'Restaurant'";
61   featureLayer.outFields = ["NAME", "ADDRESS"];
62   featureLayer.queryFeatures()
63     .then(function(results) {
64       var nearbyRestaurants = results.features.map(function(feature) {
65         return {
66           name: feature.attributes.NAME,
67           address: feature.attributes.ADDRESS,
68           distance: view.center.distance(feature.geometry)
69         };
70       });
71       nearbyRestaurants.sort(function(a, b) {
72         return a.distance - b.distance;
73       });
74     });
75   });
76
77
78
79
80
81

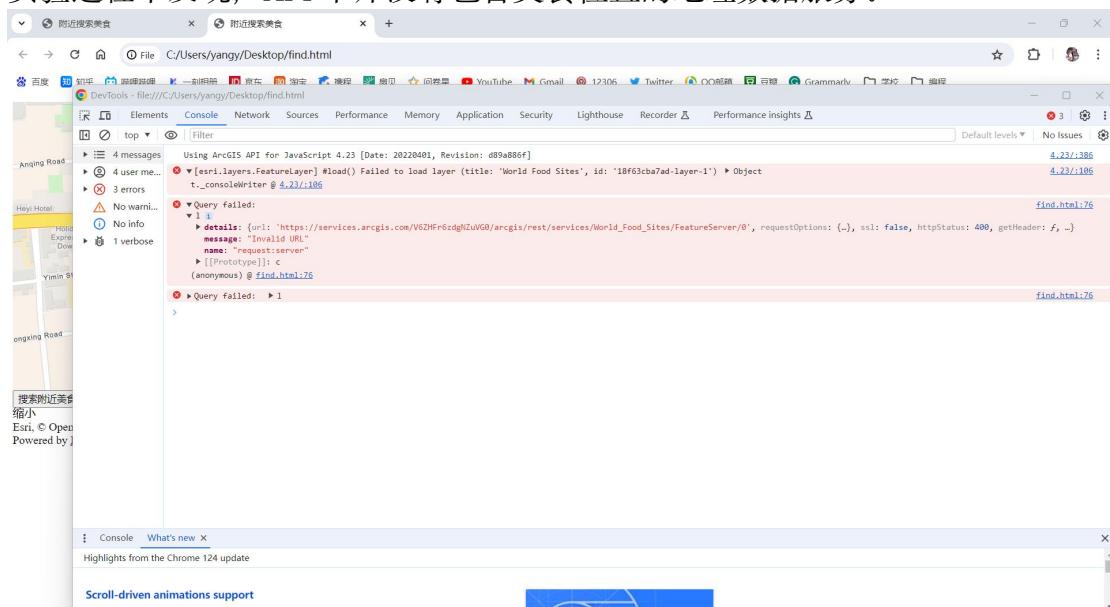
```

(3) 使用 ArcGIS API 中的 Query



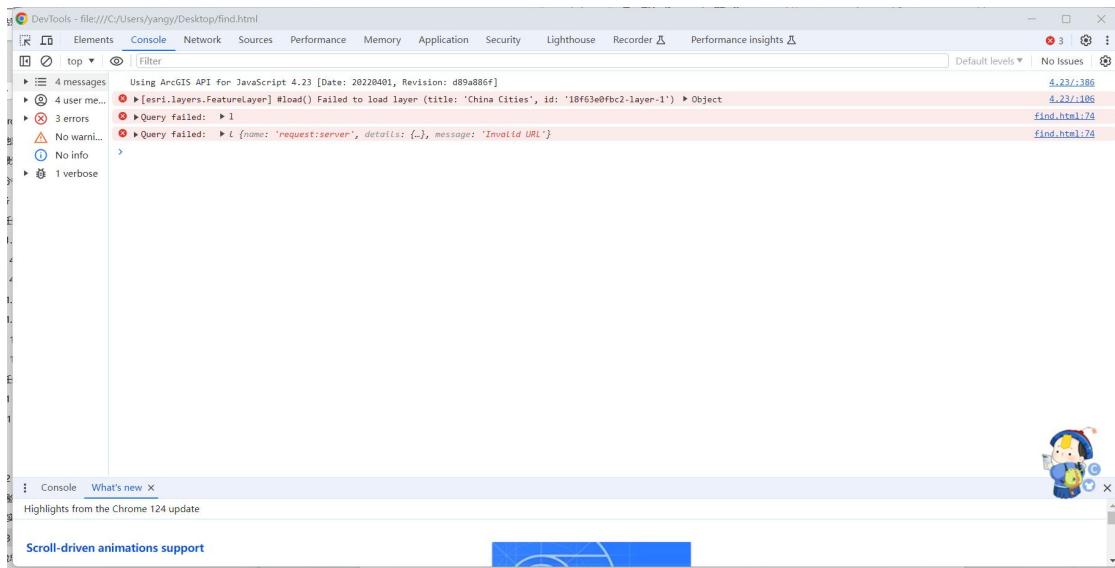
4.2.2 关于 ArcGIS API for JavaScript 的相关代码实现

实验过程中发现，API 中并没有包含美食位置的地理数据服务。



建立连接失败

要实际应用到美食搜索，需要替换示例服务的 URL，确保使用的是包含美食位置的地理数据服务。



尝试了好几个中国数据，都无法建立链接。

4.2.3 尝试修改目标

The screenshot shows the ArcGIS Developers website at developers.arcgis.com/javascript/latest/. The page title is 'ArcGIS Maps SDK for JavaScript'. The main content area features a map visualization titled 'Census Tracts' showing household income ranges across a geographic area. To the left is a navigation sidebar with links to Overview, Key features, Get started, Tutorials, Release notes, Components (beta), Maps (2D), Scenes (3D), Layers, Query, Edit, Visualization, and Geocoding. To the right, there's a sidebar titled 'On this page' with links to Where to start, Sample code, Showcase, Tutorials, and Blog. A 'Looking for 3.x?' link is also present. At the bottom, there's a feedback section asking 'Was this page helpful?' with 'Yes' and 'No' options.

通过在官网上的搜寻，我决定尝试使用包含美国城市数据的示例服务来修改代码。使用名为 "USA Major Cities" 的示例服务，该服务包含美国主要城市的位置数据。以下是这个服务的 URL：

https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/USA_Major_Cities/FeatureServer/0

同时修改目标为：搜索附近美国城市

C:\Users\yangy\Desktop\find.html - Notepad++

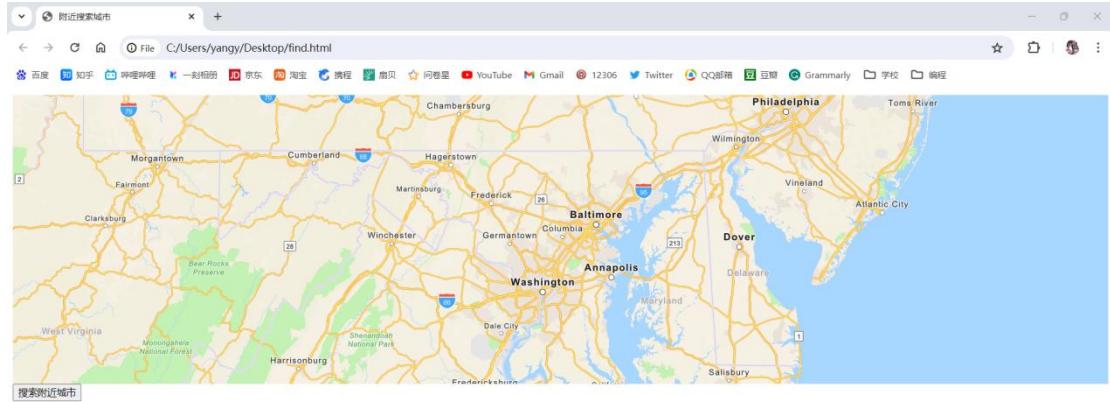
```

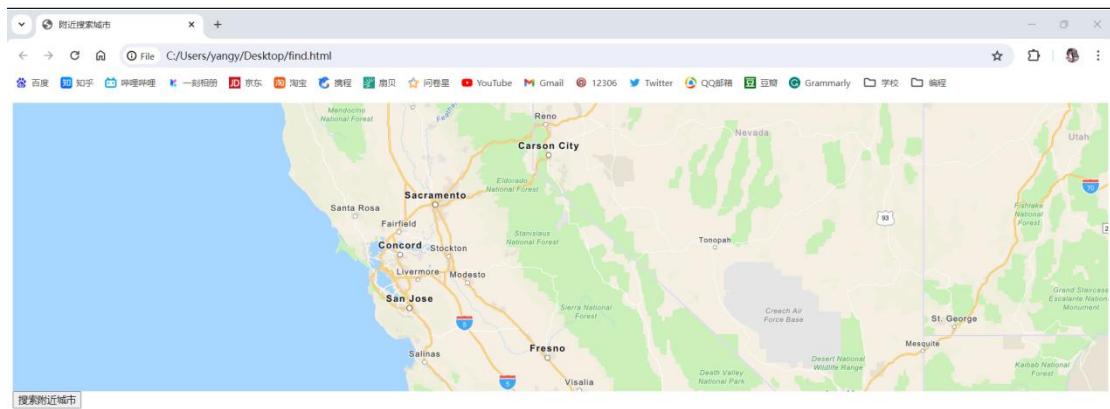
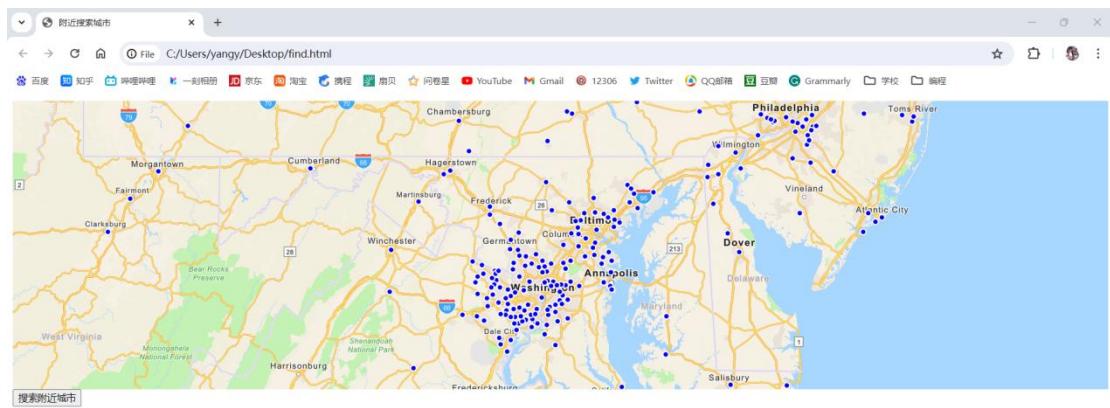
40 var graphicsLayer = new GraphicsLayer();
41 map.add(graphicsLayer);
42
43 var featureLayer = new FeatureLayer({
44   url: "https://services.arcgis.com/P3ePLMyz2RVChkxJ/arcgis/rest/services/USMajor_Cities/FeatureServer/0"
45 });
46
47 document.getElementById("searchButton").addEventListener("click", function() {
48   var circle = new Circle({
49     center: view.center,
50     radius: 2000000 // 单位默认为米
51   });
52
53   featureLayer.queryFeatures({
54     geometry: circle,
55     spatialRelationship: "intersects",
56     returnGeometry: true
57   }).then(function(response) {
58     console.log("Found " + response.features.length + " features");
59     graphicsLayer.removeAll();
60     response.features.forEach(function(feature) {
61       var graphic = new Graphic({
62         geometry: feature.geometry,
63         symbol: {
64           type: "simple-marker",
65           color: "blue",
66           size: "6px",
67           outline: { color: "white", width: 1 }
68         }
69       });
70       graphicsLayer.add(graphic);
71     });
72   }).catch(function(error) {
73     console.error("Query failed: ", error);
74   });
75 });
76
77 </script>
78 </body>
79 </html>
80
81

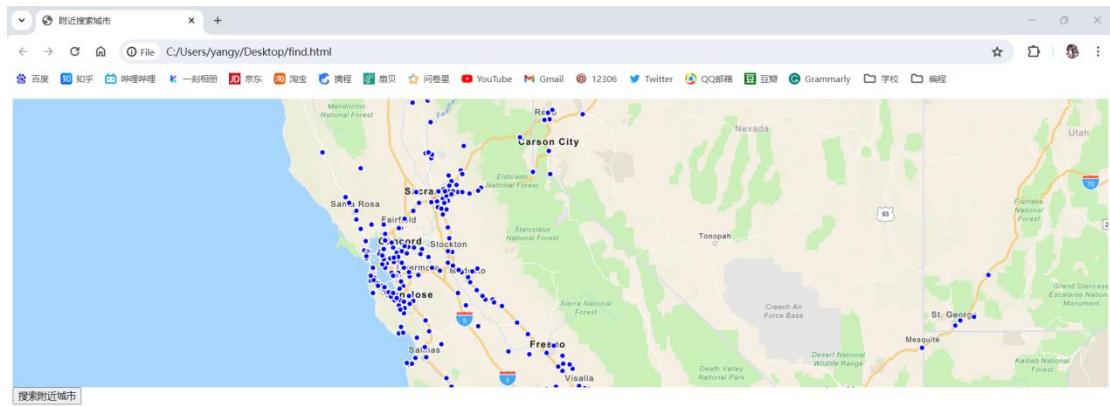
```

Hyper Text Markup Language File length: 2,311 lines: 81 Ln: 81 Col: 1 Pos: 2,312 Windows (CR LF) UTF-8 INS

同时将范围扩大 1000 倍，为附近 2000km 的美国城市







成功实现

证明代码是正确的，无法建立 URL 链接，找不到合适的 API 是主要问题

五、实验总结

这次实验耗时较长，除了程序设计最短路径的练习，主要针对 ARCGIS 的相关软件进行使用练习。实验围绕图状结构和最短路径算法展开，通过编程实现了基于旅游景点图的最短路径计算，以及在 ArcGIS 平台上进行了最短路径分析的操作。同时，还尝试了基于 ArcGIS API 实现 web 端地图最短路径分析和百度地图的附近美食搜索，以及在实践中遇到的问题和解决方法，为加深对地理信息系统的理解提供了实际操作的机会。

在实验过程中，我深刻认识到了选点的重要性，在进行地图分析时需要仔细选择起点和终点，以及在 ArcGIS API 中要确保使用正确的地理数据服务 URL。虽然在尝试美食搜索时遇到了链接建立失败的问题，但通过不断尝试和探索，成功利用美国城市数据服务进行了附近城市搜索。

这次实验不仅提升了我对图状结构和最短路径算法的理解，还让我熟悉了 ArcGIS 软件的使用和 API 的调用。尤其是在解决问题的过程中，锻炼了我解决实际工程问题的能力，提高了我的动手能力和逻辑思维能力。

在今后的学习和工作中，我将继续加强对地理信息系统和相关技术的学习，不断提升自己的能力，为更好地应对各种工程挑战做好准备。