



城市网络模型与算法上机实习参考

城市网络模型与算法上机实习



CONTENT

- 01 实习一：ArcGIS网络分析
- 02 实习二：最短路径算法实现
- 03 实习三：关阀分析
- 04 作业上交及其它注意事项



01

实习一：ArcGIS网络分析



High-performance Spatial Computational Intelligence Lab @ CUG



实习数据：1.武汉市路网数据

2.武汉市部分kfc位置数据

3.武汉市部分高校教学楼数据

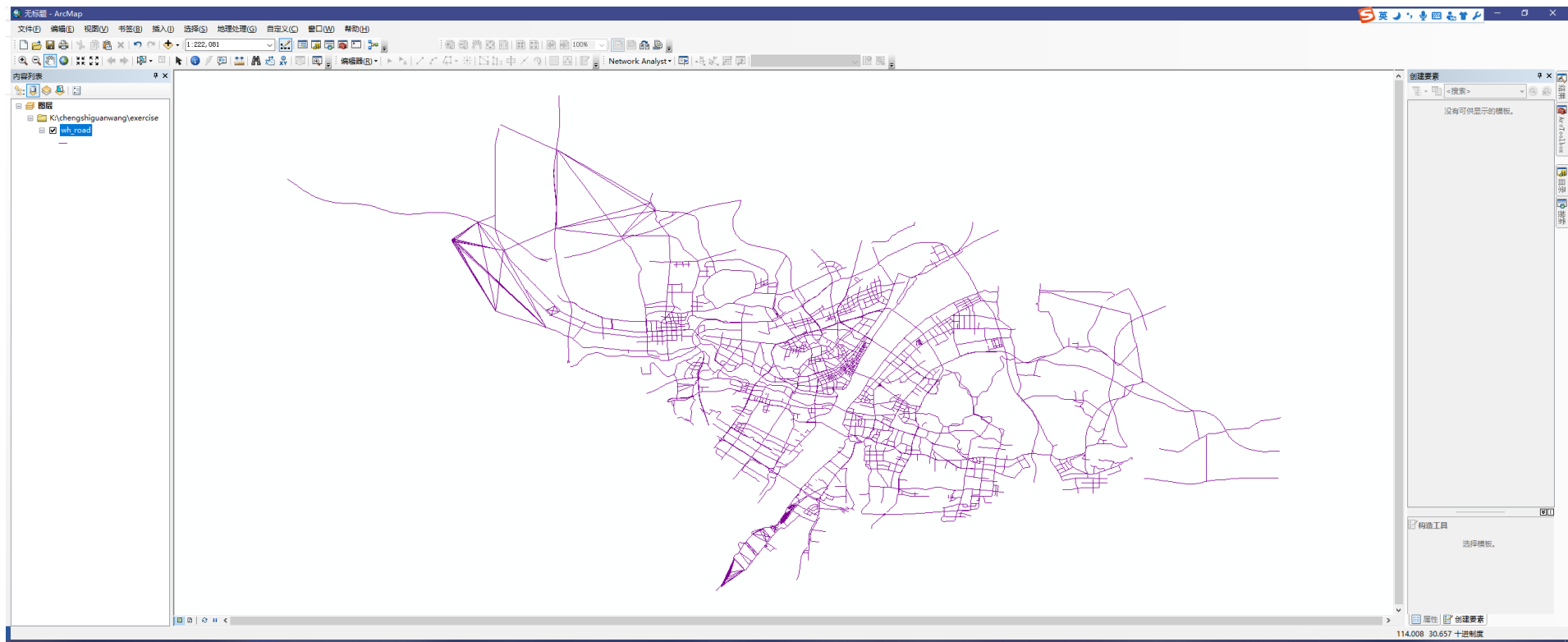
实习内容：1.最短路径分析，在武汉市内随意选取一点以及10家kfc。分别计算该点至选取kfc的最短路径。

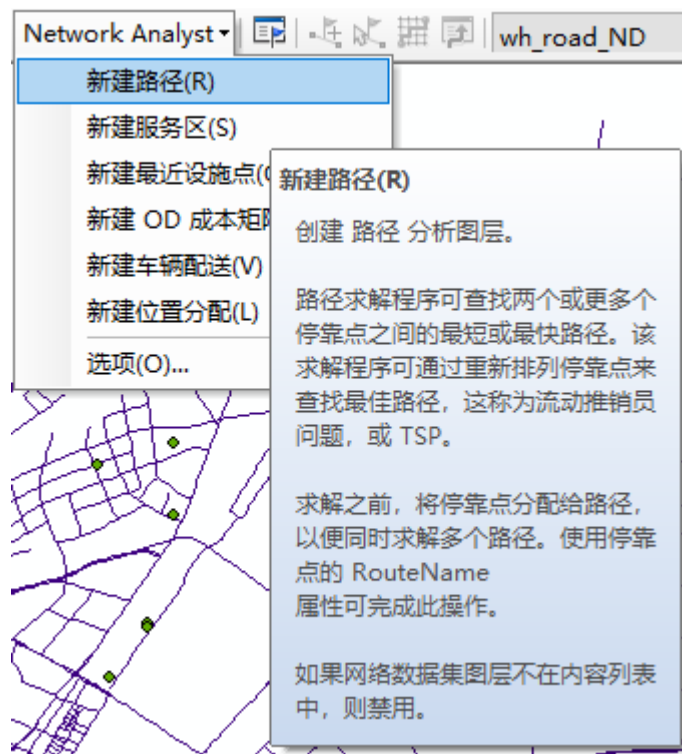
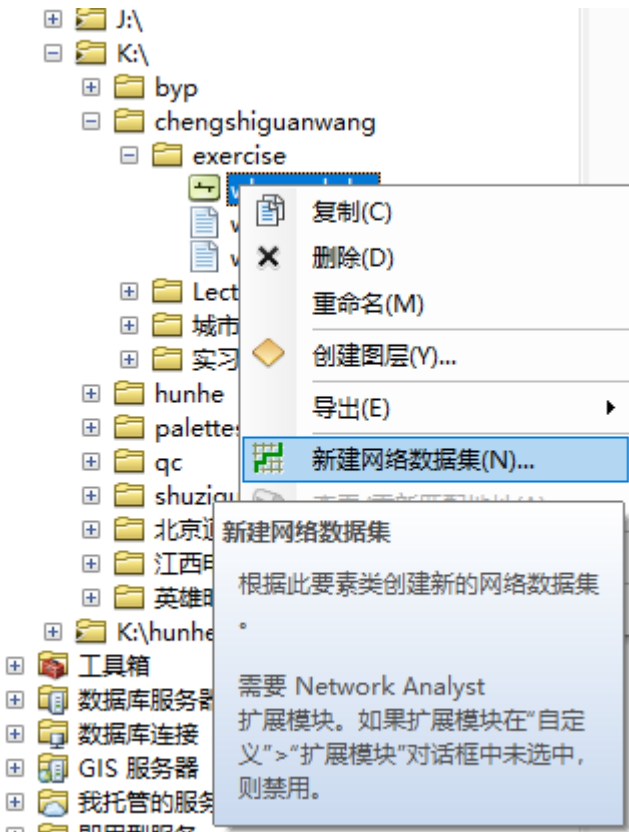
2.查找教学楼数据中所有教学楼最近的kfc位置。

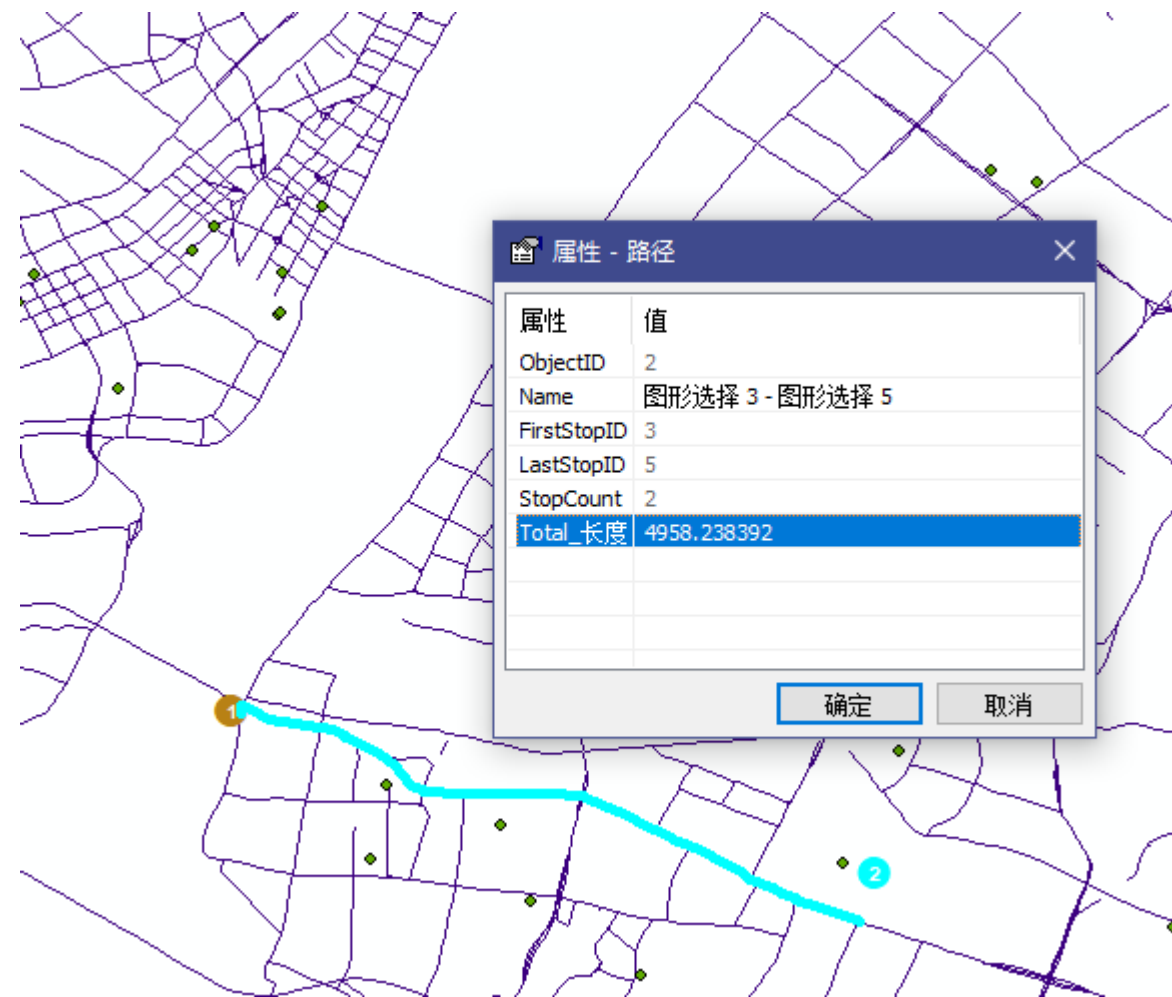
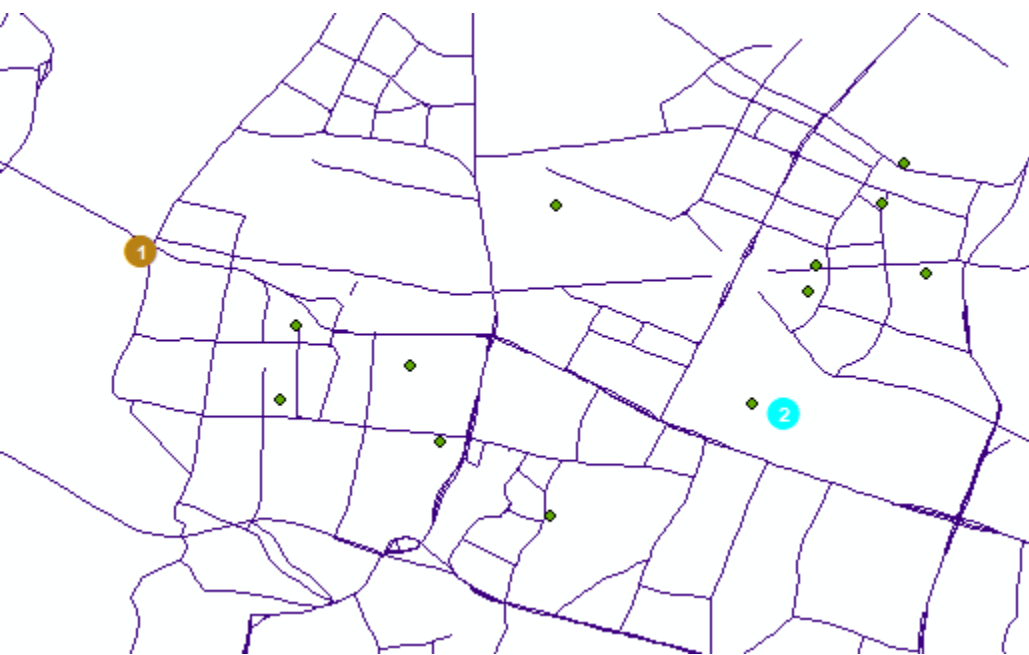
3.计算武汉市内所有kfc沿路网800m、2000m的服务范围。

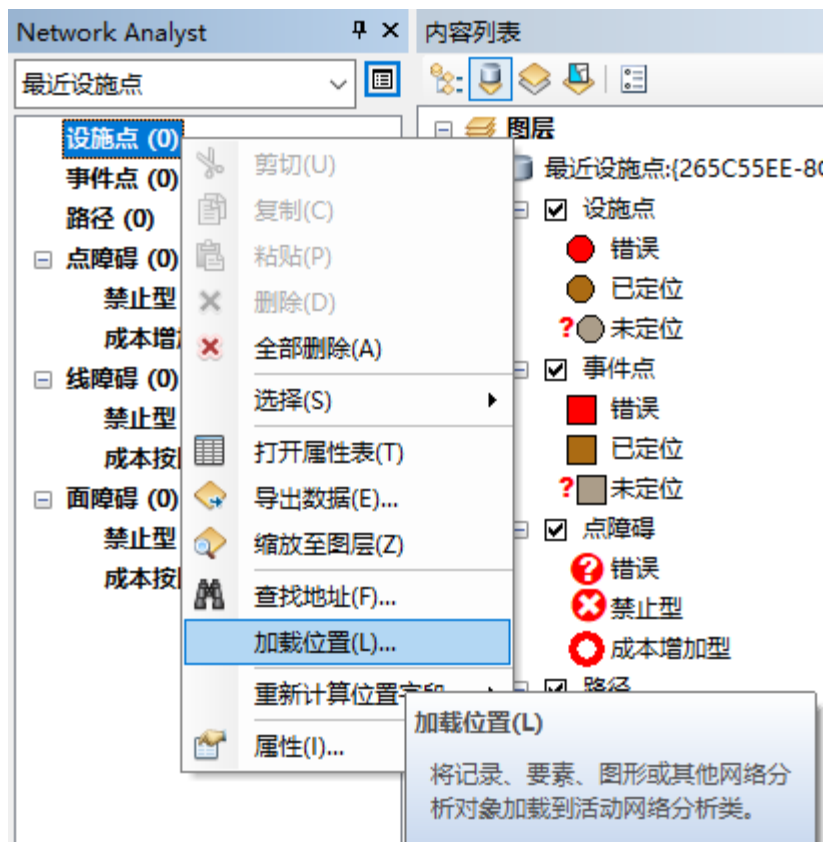
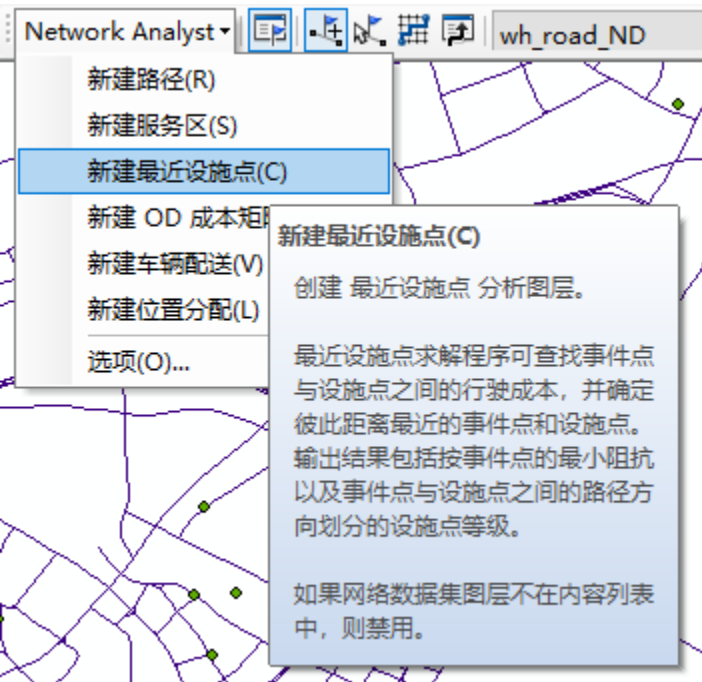
4.在武汉市范围内需要建立5个kfc配送仓库，每个配送仓库的服务半径为沿路网5000m。调整仓库位置，试让5个仓库能配送到尽可能多的kfc。输出所有kfc的配送仓库编号，以及配送OD矩阵，并试计算当配送范围为多少时，能够配送全市的kfc。

- 数据管理工具
 - LAS 数据集
 - 关系类
 - 几何网络
 - 分布式地理数据库
 - 切片缓存
 - 制图综合
 - 图层和表视图
 - 图表
 - 地理数据库管理
 - 子类型
 - 字段
 - 属性域
 - 工作空间
 - 常规
 - 归档
 - 打包
 - 投影和变换
 - 拓扑
 - 数据比较
 - 文件地理数据库
 - 栅格
 - 照片
 - 版本
 - 索引
 - 表
 - 要素
 - XY 转线



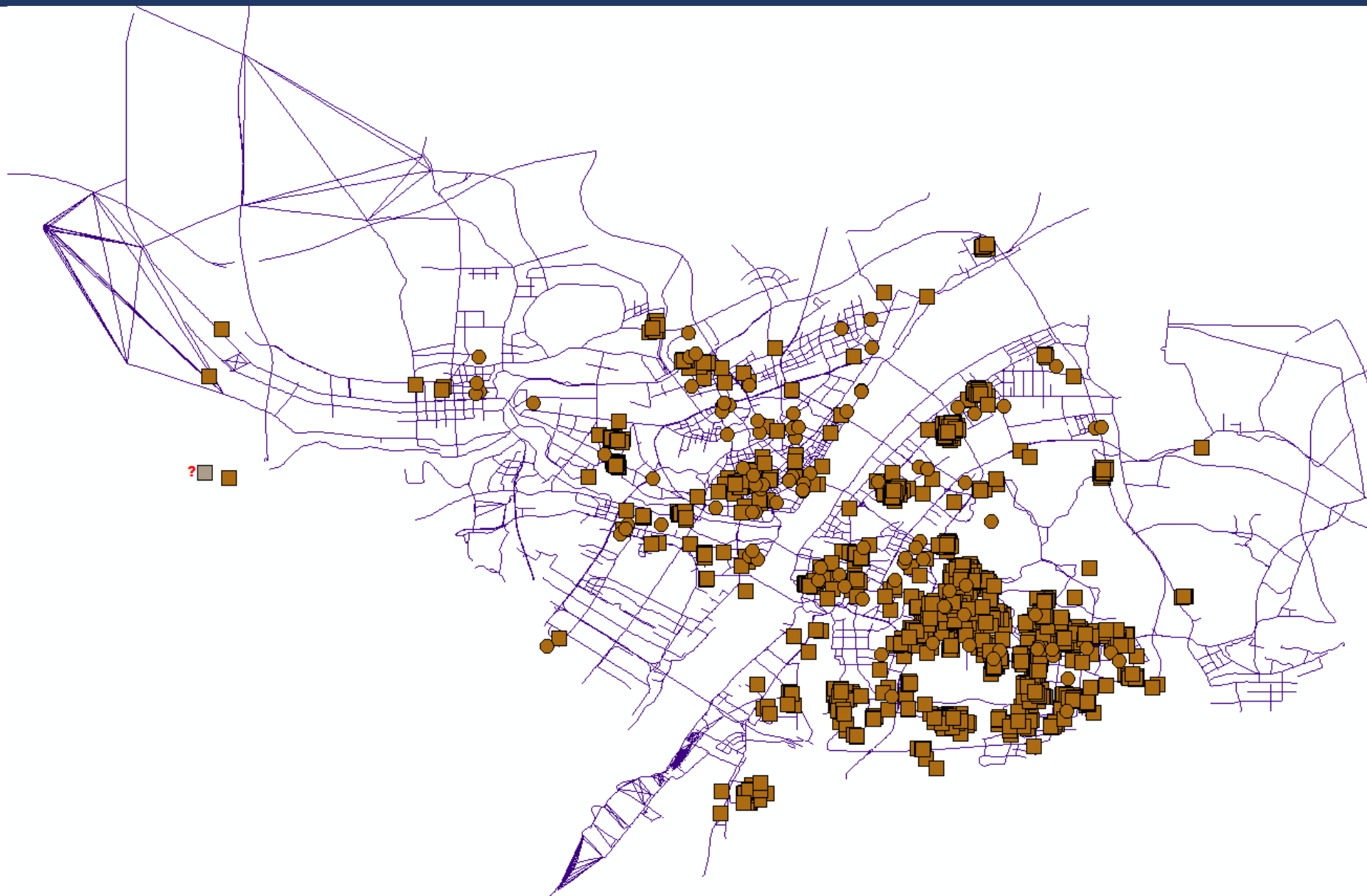








同理将school加载为事件点



图层属性

常规 图层 源 分析设置 累积 网络位置

设置

阻抗(I): 长度(米)

☐ 使用时间:

使用方法(U): 开始时间

时间(T): 8:00

☒ 星期(W): 今天

☐ 具体日期(D): 2019/ 4/15

默认中断值(C): 5000

要查找的设施点(F): 3

行驶自(T):

☒ 事件点到设施点(N)

☐ 设施点到事件点(Y)

交汇点的 U 形转弯(U): 允许

输出 Shape 类型(S): 具有测里值的实际形状

☐ 应用等级(H)

☒ 忽略无效位置(V)

限制

方向

距离单位(E):

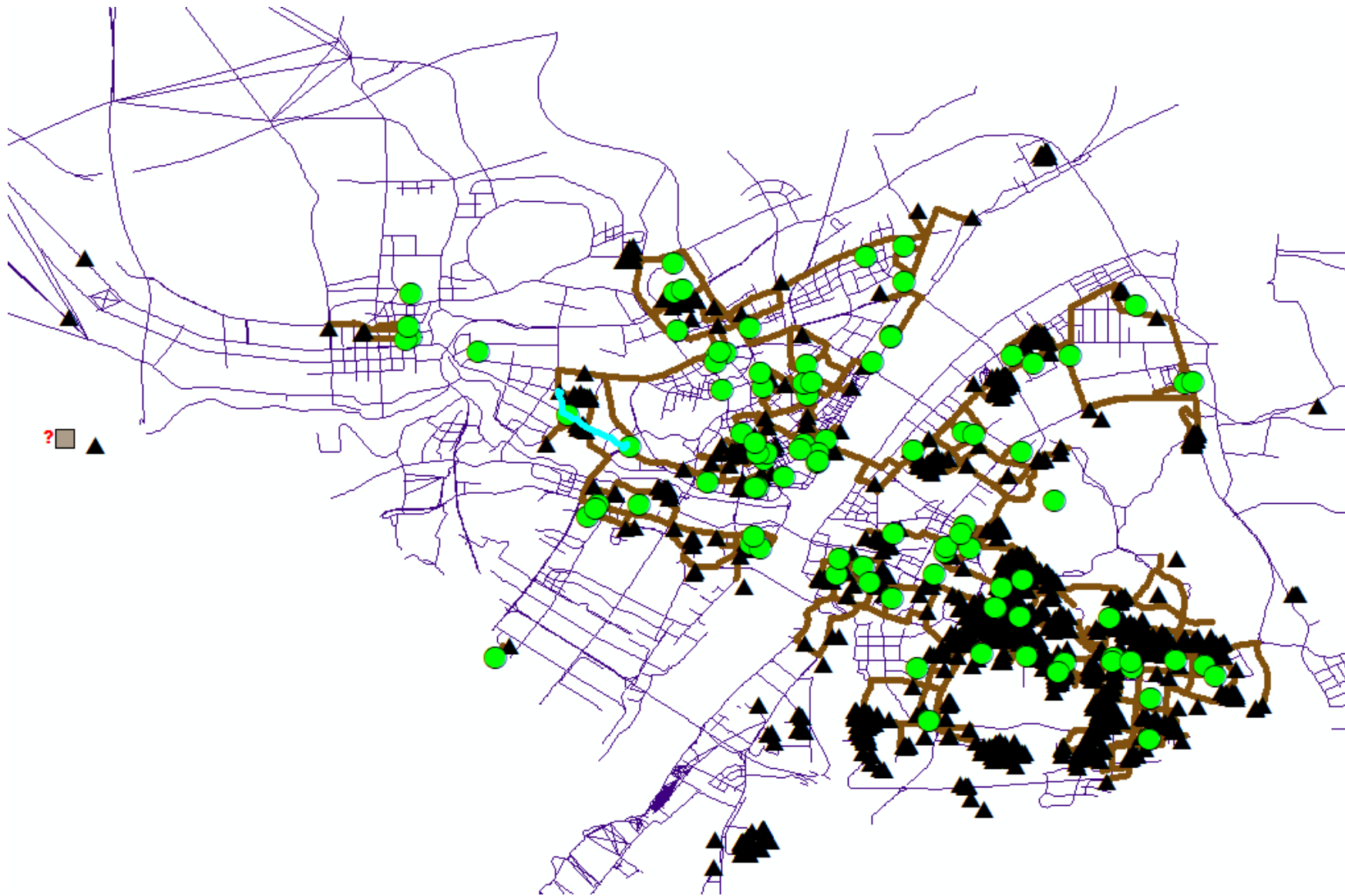
☐ 使用时间属性(M)

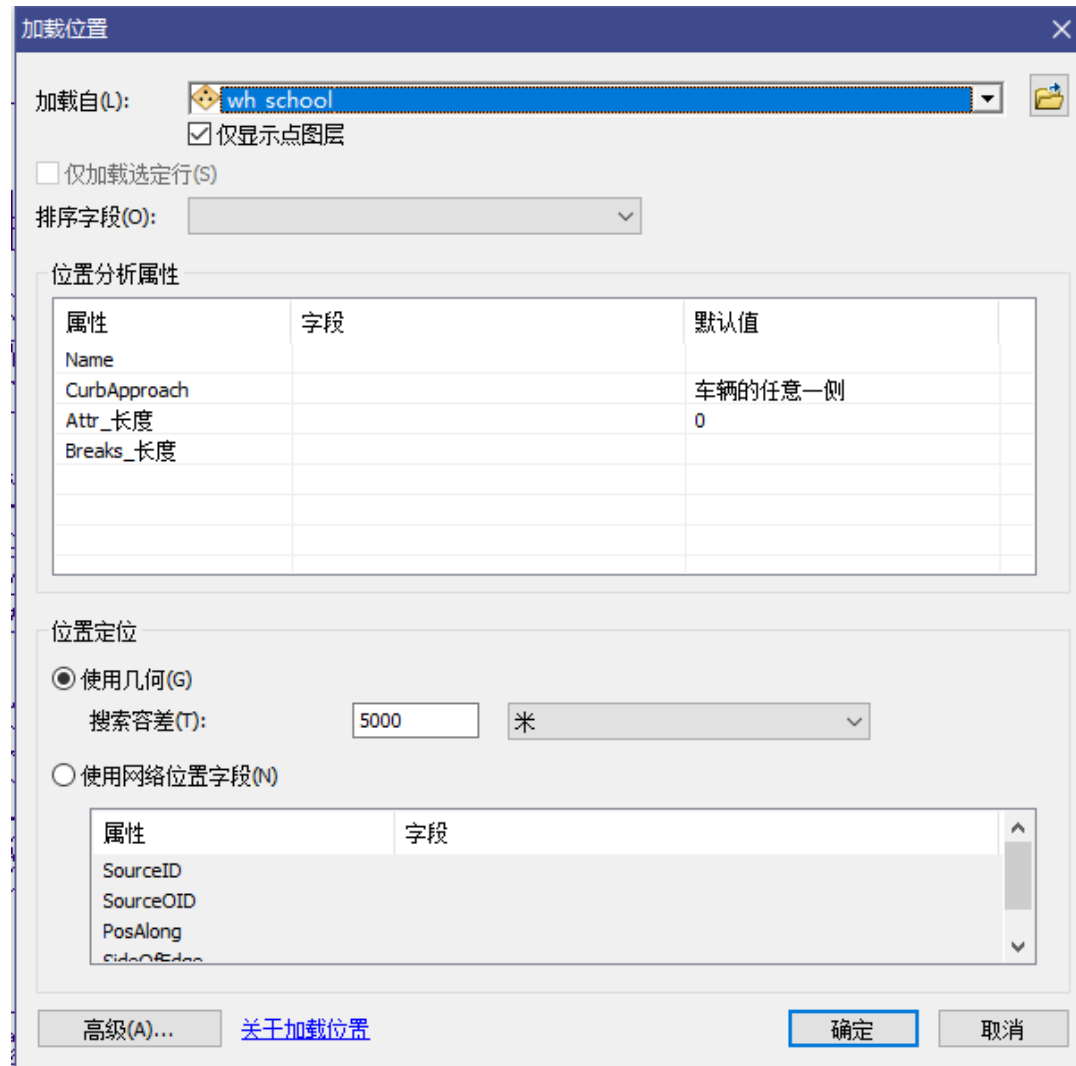
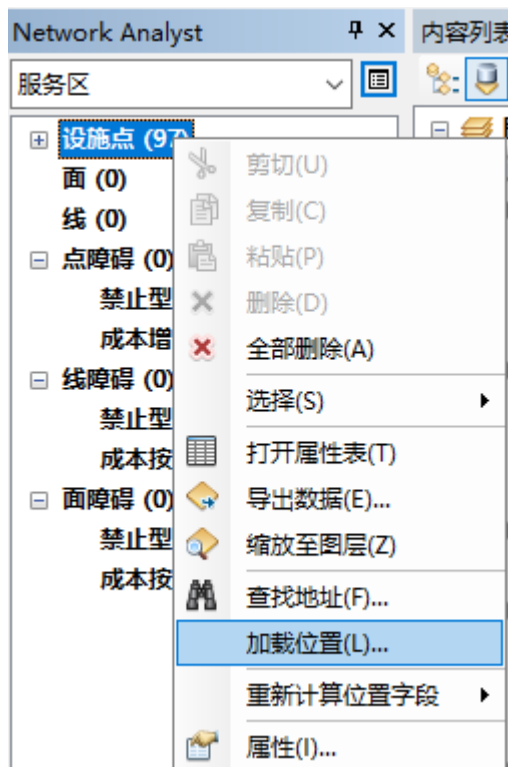
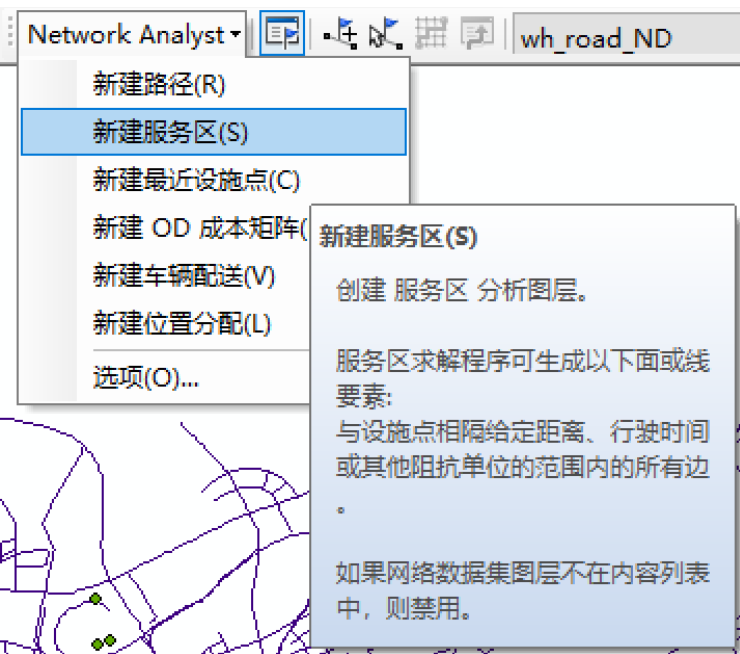
☐ 自动打开方向窗口(P)

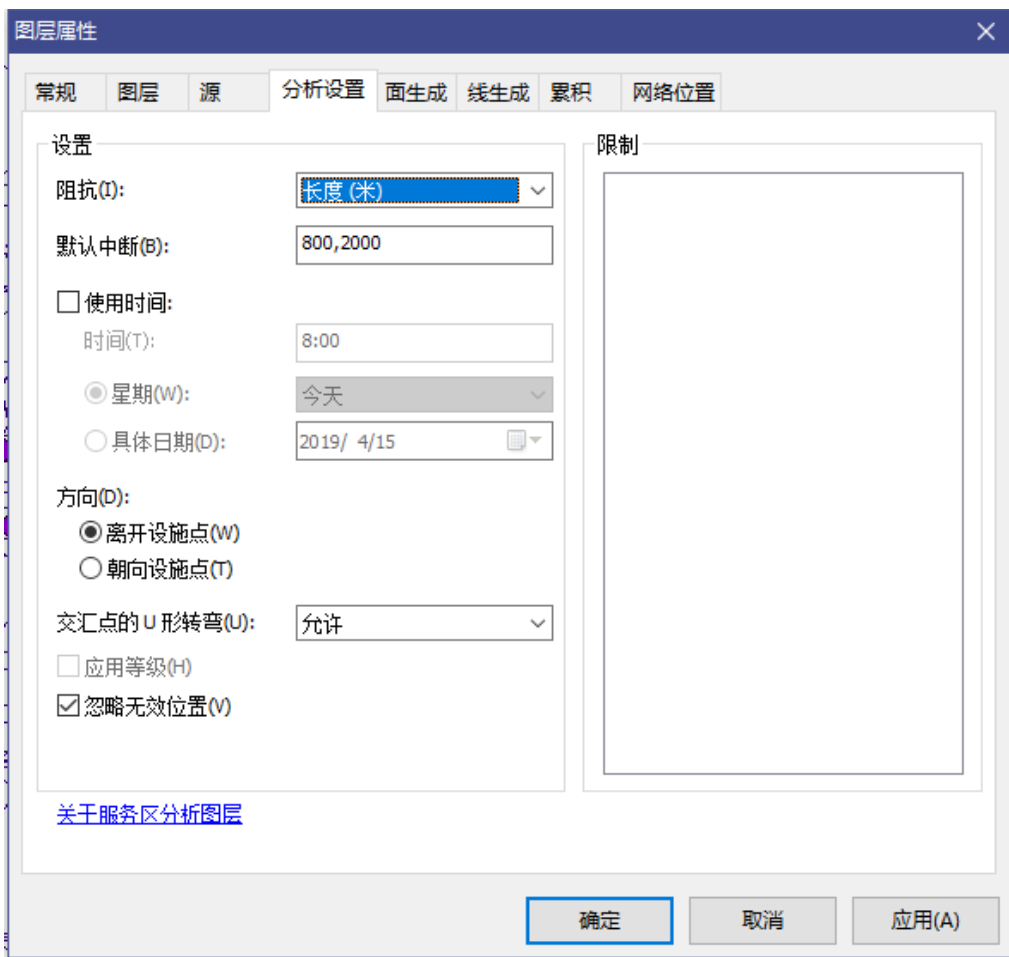
[关于最近设施点分析图层](#)

确定 取消 应用(A)

- 1.默认中断值：即最大搜索范围，超出最大搜索范围之外的设施点将被忽略。
- 2.要查找的设施点：即最多搜索离事件点最近的n个设施点，此时默认中断值仍然适用。即使默认中断值内仅有n-1个设施点，仍将结束搜索。
- 3.此处请自行设置参数



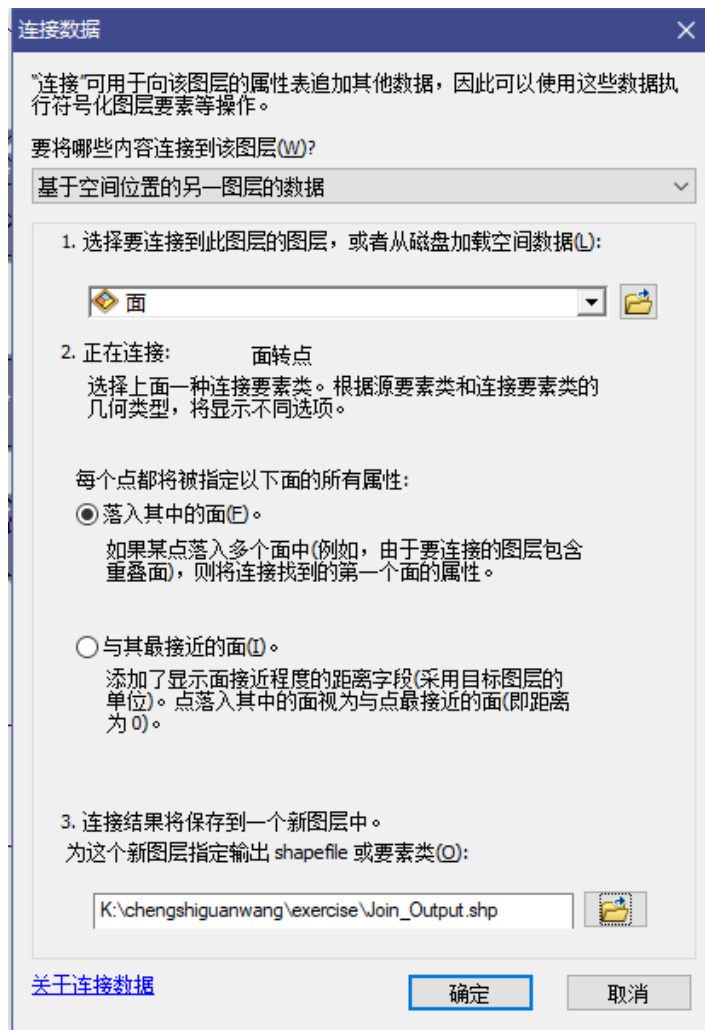
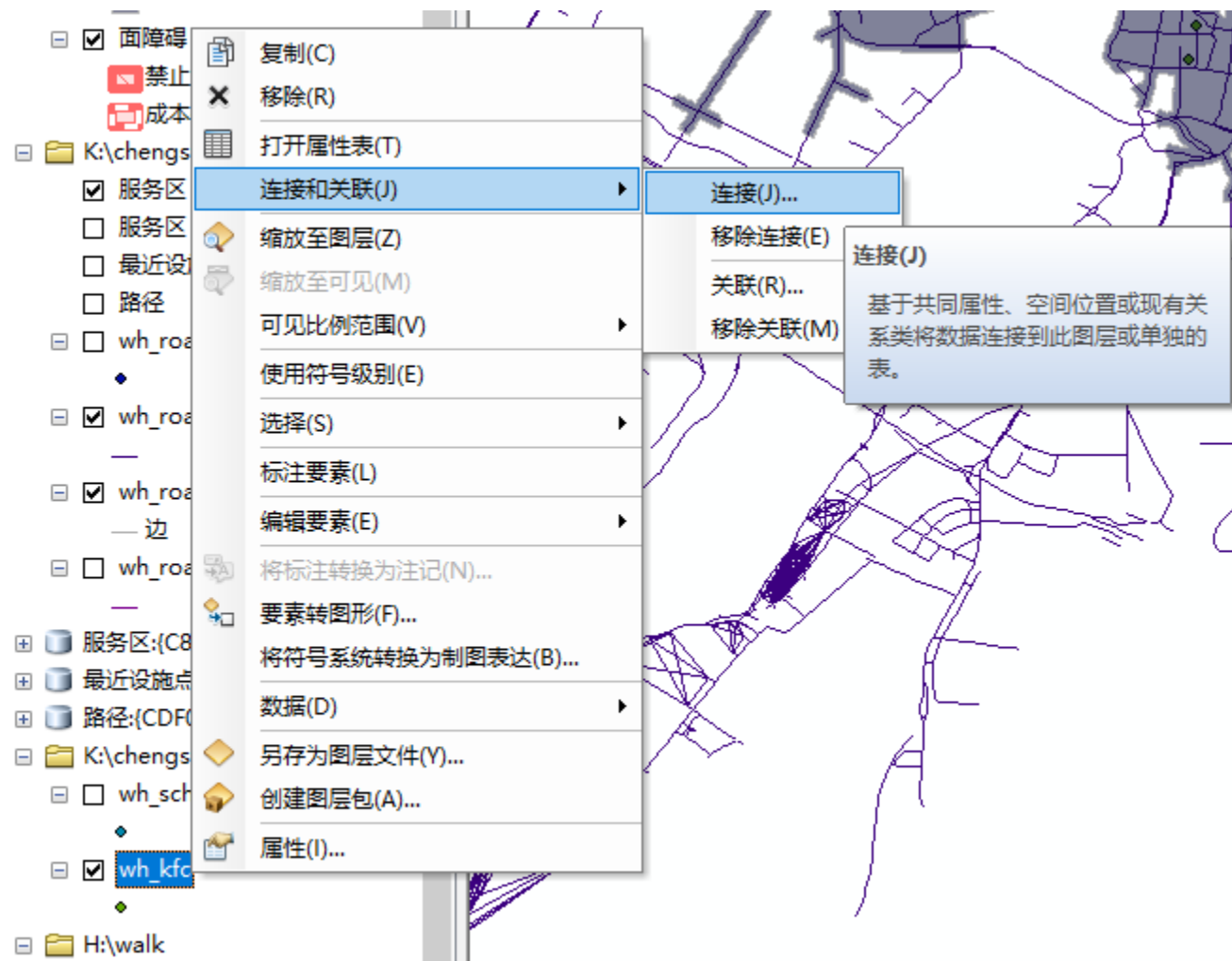




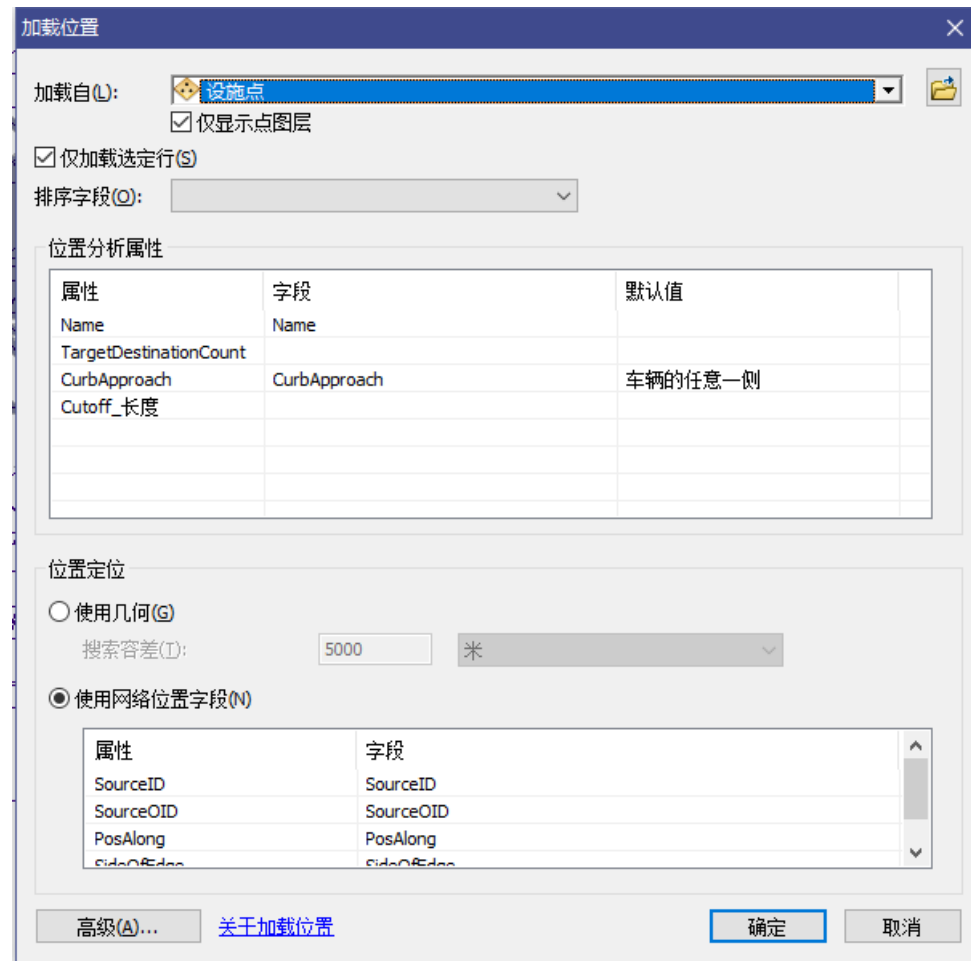
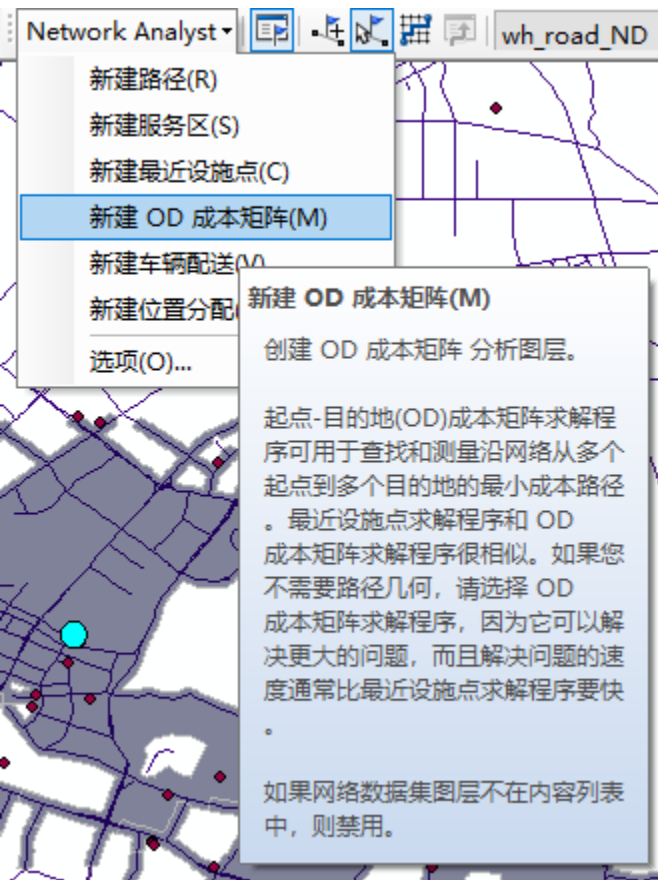


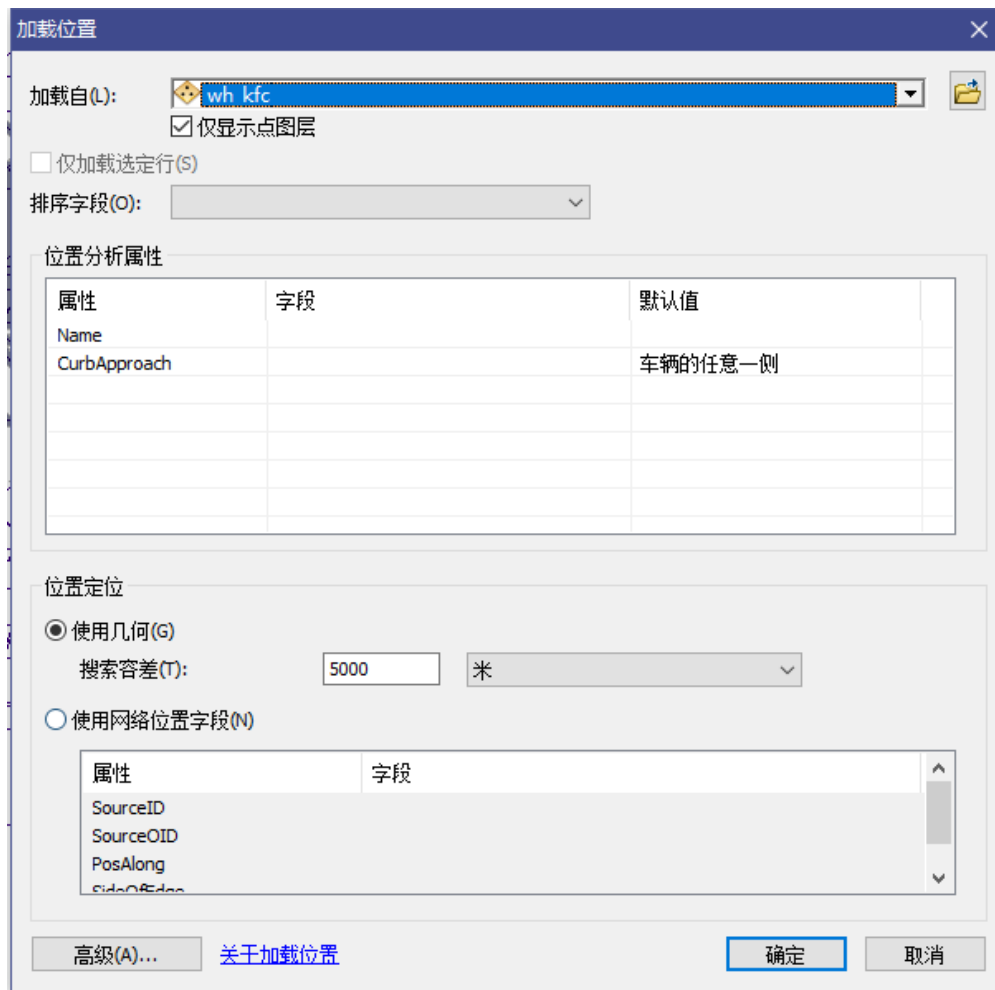
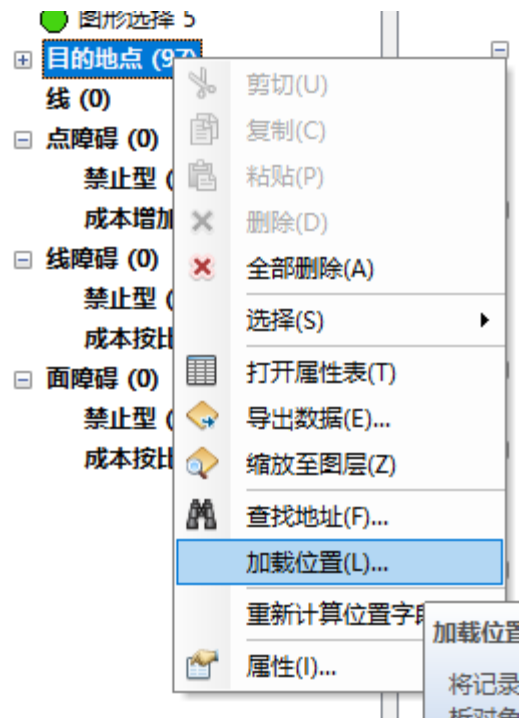
依照上一题的步骤 创建一个配送仓库并构建道路缓冲区 调整仓库位置 使仓库覆盖尽可能多的Kfc。

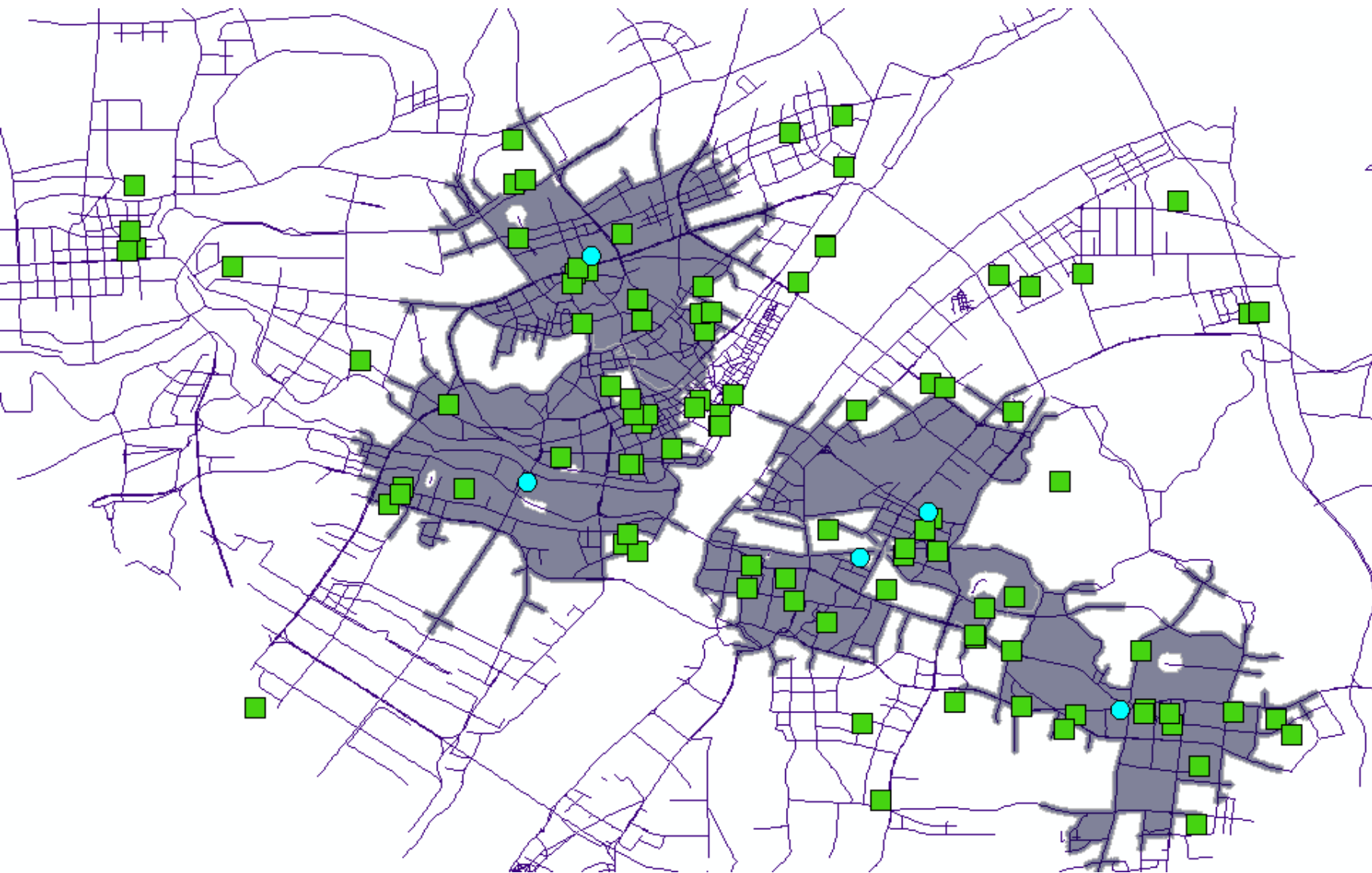


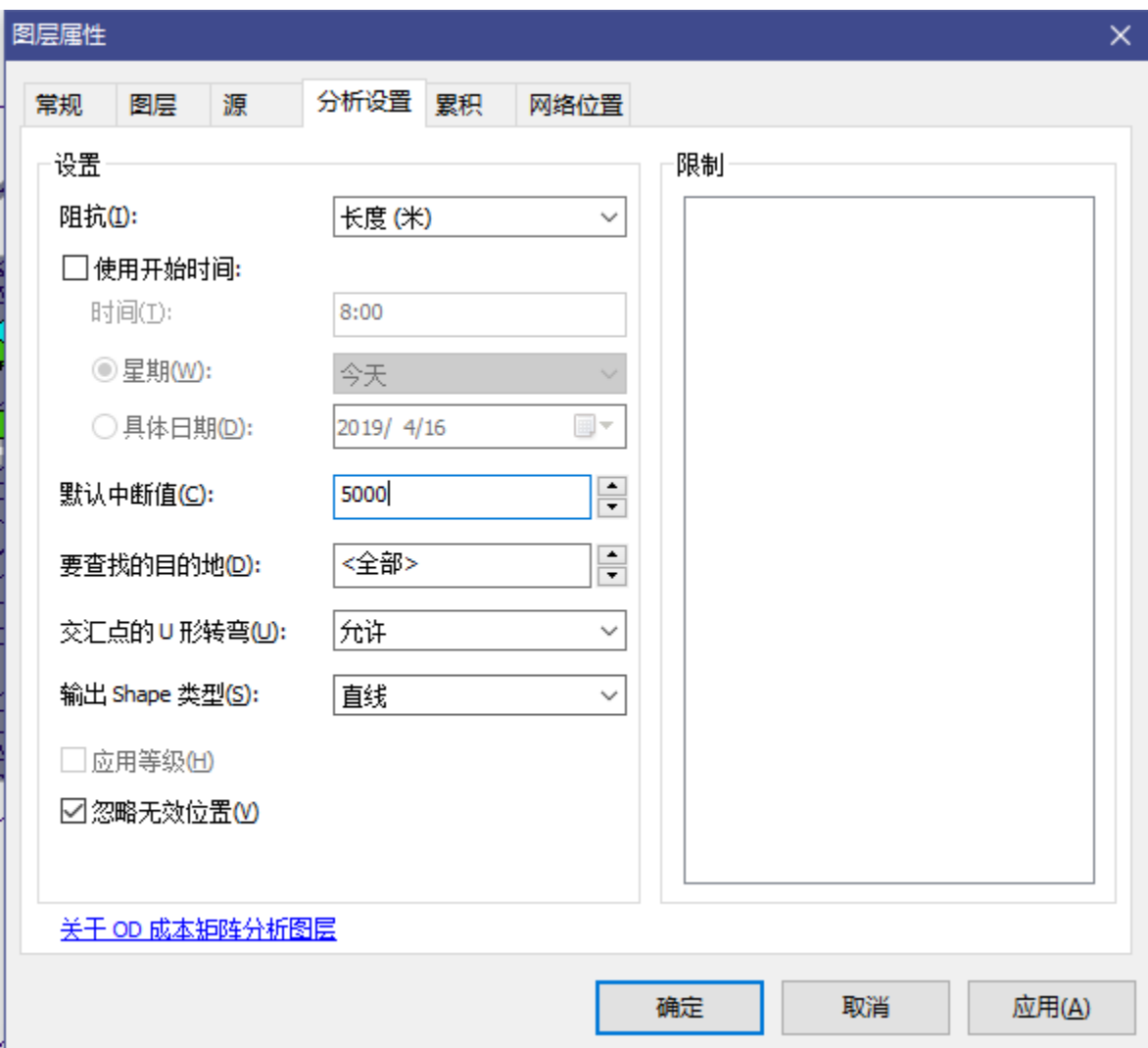


导出属性表 存为join.txt
范围内的点数量 并给出落在配送









ObjectID	Shape	Name	OriginID	DestinationID	DestinationRank	Total_长度
1	折线	图形选择 1 - 位置 81	2	81	1	2379.748236
2	折线	图形选择 1 - 位置 69	2	69	2	2912.387845
3	折线	图形选择 1 - 位置 19	2	19	3	3002.307113
4	折线	图形选择 1 - 位置 58	2	58	4	3181.454252
5	折线	图形选择 1 - 位置 27	2	27	5	3556.803595
6	折线	图形选择 1 - 位置 75	2	75	6	3681.344751
7	折线	图形选择 1 - 位置 66	2	66	7	3843.090532
8	折线	图形选择 1 - 位置 74	2	74	8	3928.043768
9	折线	图形选择 1 - 位置 6	2	6	9	3985.060386
10	折线	图形选择 1 - 位置 29	2	29	10	4010.818674
11	折线	图形选择 1 - 位置 13	2	13	11	4040.631842
12	折线	图形选择 1 - 位置 9	2	9	12	4099.455832
13	折线	图形选择 1 - 位置 93	2	93	13	4260.007969
14	折线	图形选择 1 - 位置 59	2	59	14	4311.226394
15	折线	图形选择 1 - 位置 43	2	43	15	4397.287883
16	折线	图形选择 1 - 位置 17	2	17	16	4977.246582
17	折线	图形选择 2 - 位置 11	3	11	1	2136.992979
18	折线	图形选择 2 - 位置 85	3	85	2	2612.209284
19	折线	图形选择 2 - 位置 65	3	65	3	3030.58287
20	折线	图形选择 2 - 位置 1	3	1	4	3160.249387
21	折线	图形选择 2 - 位置 63	3	63	5	3162.430011
22	折线	图形选择 2 - 位置 7	3	7	6	3519.568302
23	折线	图形选择 3 - 位置 22	4	22	1	134.614695
24	折线	图形选择 3 - 位置 92	4	92	2	610.265488
25	折线	图形选择 3 - 位置 37	4	37	3	1154.052018
26	折线	图形选择 3 - 位置 21	4	21	4	1326.218624
27	折线	图形选择 3 - 位置 64	4	64	5	1360.466674
28	折线	图形选择 3 - 位置 24	4	24	6	2919.770977
29	折线	图形选择 3 - 位置 28	4	28	7	3398.291718
30	折线	图形选择 3 - 位置 32	4	32	8	3564.70174
31	折线	图形选择 3 - 位置 82	4	82	9	4244.808567
32	折线	图形选择 3 - 位置 23	4	23	10	4385.692782
33	折线	图形选择 3 - 位置 91	4	91	11	4395.481865
34	折线	图形选择 3 - 位置 56	4	56	12	4396.708942
35	折线	图形选择 3 - 位置 89	4	89	13	4436.272912
36	折线	图形选择 3 - 位置 72	4	72	14	4641.100839
37	折线	图形选择 3 - 位置 54	4	54	15	4772.914135
38	折线	图形选择 4 - 位置 20	5	20	1	629.208818
39	折线	图形选择 4 - 位置 90	5	90	2	1070.500846
40	折线	图形选择 4 - 位置 76	5	76	3	1567.138272
41	折线	图形选择 4 - 位置 52	5	52	4	1784.103362
42	折线	图形选择 4 - 位置 45	5	45	5	1834.966288

1. 请将kfc店名与位置相关联后，将OD矩阵另存为OD5000.txt导出。
2. 试调整仓库配送范围，使仓库能配送全市范围的kfc。给出配送范围建议（不同仓库配送范围可不同），结果od矩阵导出为OD_all.txt。

02

实习二：最短路径算法实现





实习数据：1.la-ext.sp

2.node.txt、edge.txt、union.txt

1、2数据仅格式不同，若使用1数据，请参考参考代码进行读取

实习内容：1.实现一种最短路径算法。

2.基于给出的数据，输出由指定点i至网络中前10000个点的最短距离。结果存为txt。

点i：班级序号+班号。如1班01号，实习题目就是输出id为101的点到网络中前10000个点的距离

结果保存格式：start_id,end_id,dist

101,1,1436410

101,2,1302440

.....

3.实现最短路径的输出，从i点出发，任选3个点，输出最短路径以及经过的节点。保存程序截图、或另存一个txt均可。



```
请输入你要搜索的最短路径的起始点：101
请输入终点：1
你所搜索的最短路径长度为：1436410
期间会经过的点为：1->101043->101054->101050->101042->101038->101033->100890->101030->100889->100838->100835->100829->100
832->100811->100798->100803->100801->95576->95573->95575->95570->95551->95564->95562->95548->95537->95535->95534->95470-
>95468->95448->95447->95439->95438->95433->143061->143060->192036->143053->192107->143052->189303->189302->189301->19017
0->190650->95230->143041->143043->143042->95219->143040->143039->95201->95200->144839->144838->95191->95204->144842->951
49->190986->95146->95131->95134->95130->189258->189257->189256->191230->87473->87470->87462->87430->87429->87426->189349
->87421->87415->87416->87315->87314->87313->87312->87310->87306->87305->87297->87296->189660->87292->87291->87263->87253
->87134->87239->142872->142871->87235->87234->87102->84058->84057->84056->84055->190153->84031->84030->84029->84014->840
22->84025->84023->83968->83967->83965->83957->83953->83951->83962->83944->83943->83942->83939->83938->83936->83934->8393
3->83658->83657->83656->83655->83641->83638->83627->190710->190709->190712->190711->188996->187999->187998->83616->83618
->83564->187997->187996->187995->187994->187993->187992->187991->187990->187989->187988->187987->187986->187985->187984-
>187983->187982->189618->189617->189621->189620->189619->187981->189624->189623->189622->189625->189627->189626->189629-
>189628->189632->189631->189630->82738->82717->82737->82736->82745->82735->82734->82733->82724->82726->149613->82729->15
0481->82722->82721->82720->82465->82464->82463->82462->82461->82458->82456->82455->82454->82450->82452->82429->82428->82
427->82426->82425->82424->82390->82417->82388->82416->82374->82415->82414->82357->82356->82355->82325->82353->82324->823
51->82345->82322->82344->82320->82318->82342->82303->82333->82332->82294->82292->82290->82289->82328->82287->82327->5272
4->52756->52722->52752->52751->52749->52717->52701->52730->52700->52699->52696->52726->52666->52665->52662->52663->52661
->52658->52657->52656->52655->52654->52647->52646->52645->52644->52610->142150->52637->52364->52339->142149->52363->5236
2->52332->52361->52326->52359->52358->186562->186561->186560->52309->52305->52344->52342->52341->52304->52303->52253->52
252->52259->52251->52247->52246->52245->52244->52243->52241->52240->52216->52184->52212->51223->51245->51233->51207->512
32->51205->51227->51226->51224->51166->51140->51164->51163->51126->51149->51143->50814->50813->50811->50806->50805->5074
9->158698->187907->142124->50729->50728->50269->50178->186069->186070->142115->142114->50129->50127->50097
->50085->49980->185762->185763->49972->49960->46747->46748->46744->46697->46698->142078->46677->46667->46668->45499->454
97->45498->45494->45463->45458->45456->45440->45447->45404->45395->45396->185786->185787->45356->45360->45359->45355->45
354->45287->45285->45283->186917->186918->45168->45056->45057->45051->45041->45031->155649->187895->187894->155636->1847
71->185647->44605->44535->44533->44532->44527->44526->44522->44523->43695->43692->43693->43688->43686->43682->43680->436
39->43641->43637->43615->43612->43611->43610->43602->43593->43592->43590->43588->43589->43527->43516->161384->161385->28
022->28020->28019->28017->27993->28004->27991->27989->27983->27940->27981->27742->27736->27700->27693->27672->27665->276
60->27661->27659->27619->27618->27523->27608->27605->27603->27604->27602->27496->20857->20759->20834->20827->20826->2078
9->20788->20702->20675->20686->20673->20672->20641->20637->20639->20640->20636->20532->20165->20164->20161->20159->17704
4->177045->141553->20121->20120->20112->20101->175306->20098->193985->175305->19921->193978->19865->19853->19863->19521-
>19522->18144->18143->18140->18135->18122->18120->18115->18069->18070->18068->18065->18064->18062->18054->18051->18050->
17987->177948->17876->17875->177914->177915->177921->177922->177952->17858->175278->175279->193948->193946->193947->1778
6->17775->17771->17766->17767->17764->17762->17760->17017->177860->177858->177859->16998->158380->16876->16877->16856->1
6849->14399->11596->11595->11597->11594->177419->11592->11577->177420->177440->177441->177422->11502->11550->11516->1151
4->178215->178216->11497->175833->175834->175835->11481->171941->172153->172150->156992->156993->11472->11348->11347->14
1340->11342->148032->10883->10881->10877->149774->149775->10840->10841->172655->172656->141304->141305->172662->194717->
194718->155044->155041->627->626->194640->624->621->156144->156145->156148->148819->601->589->581->590->580->173593->173
597->574->552->553->172488->172489->172490->150542->150543->539->540->522->523->172678->172679->172676->172677->348->172
672->172680->172682->172681->174129->174128->150384->158046->156933->164->163->172428->80->166503->166502->150317->15031
3->150312->150319->150316->172425->172426->172427->150288->150289->126->127->128->134->170038->1
50513->150->158042->168370->168371->173->172->171->170->169->150331->194184->194183->150390->158040->101->请按任意键继续
. . .
```

03

实习三：关阀分析中的可达性分析





在自来水或煤气等类型管道发生爆管事故或进行设备检修时，需要通过关闭若干相关阀门使现场停水（停气）。

关闭阀门的方案可以有很多种，但存在一个最优方案，使得：

- 1、关闭阀门的数量最少；
- 2、停水（气）范围最小。

一般说来，关阀分析不仅要给出最优的关阀方案，还需要给出停水（气）区域，列出受影响的用户。



实习数据： 1.la-ext.sp

2.node.txt、 edge.txt、 union.txt

1、2数据仅格式不同，若使用1数据，请参考参考代码进行读取

实习内容： 1.基于给定的搜索半径100000，输出离指定点i距离为100000内的所有点。结果存为.txt。

点i： 班级序号+班号。如1班01号，实习题目就是离id为101的点距离为100000的所有点编号。

结果保存为txt或保留程序截图。

```
E:\vs\城市管网实习_1\x64\Debug\城市管网实习_1.exe
////////////////////////////////////
请输入要搜索范围的起始点: 101
请输入范围: 100000
该范围内的点为: 100 173646 141067 158040 99 173647 174 150390 173648 161958 150391 194183 102 156212 176 177 194184 187 1
94189 156213 175 150331 194204 191 103 169 188 189 194205 161956 148256 170 160237 194 161950 203 161957 148257 171 1619
60 168365 166928 208 161951 194191 204 161954 161942 161940 172 161961 161959 168366 225 207 161952 206 194190 161955 16
1941 148259 173 161963 178 168367 224 209 161953 194192 150333 148258 168371 180 161962 194181 168364 170017 239 238 205
210 150335 168370 179 183 181 170015 194182 150332 168369 170018 240 241 194196 194193 211 150334 162223 186 171732 184
182 170014 170016 160238 158039 168368 230 245 246 172920 194194 631 194195 212 215 162222 194188 158043 185 195 150388
158038 228 231 162228 244 248 247 194198 632 194197 194199 636 170895 213 214 162221 197 162219 196 150389 229 235 236
162229 249 168224 633 634 194208 217 637 638 162220 190 193 227 168864 232 237 162224 168223 168325 635 168211 642 216 2
19 639 641 192 226 168866 222 168865 173727 648 161943 659 672 218 242 221 220 643 649 162225 158044 651 673 253 243 252
644 640 646 654 171191 652 665 254 162234 645 647 650 655 255 162236 692 693 657 250 162235 251 257 请按任意键继续. . .
```

04

作业上交及其它注意事项





1. 实习一请保留所有结果部分截图，复制粘贴至word文档中，转成pdf格式上交。
2. 实习一——4，请将具体的搜索范围给出，并在图上注明。
3. 实习一还需上交OD5000.txt、OD_all.txt。
4. 实习二、实习三需上交所有源代码。
5. 实习二、三程序若保留程序截图，请分别保存，并转为pdf格式上交。
6. 实习上交格式：实习X+班级+学号+姓名；如实习一+11416X+201XXXXXXXXX+XXX
7. 三次实习请统一发至学委处，一并打包上传。
8. 所有实习作业上交截止时间：2019年5月13日17:00（超时扣分）
9. 严禁抄袭



城市网络模型与算法上机实习

袁泽皓

yuanzehao_o@foxmail.com

谢谢!



High-performance Spatial Computational Intelligence Lab @ CUG