参赛密码 ________

（由组委会填写）

“华为杯”第十四届中国研究生数学建模竞赛

学　　校	上海交通大学
参赛队号	10248229
	1.田一丁
	2.陈浩坤
	3.褚幸

参赛密码 ____________

（由组委会填写）

“华为杯”第十四届中国研究生

数学建模竞赛

题 目 **构建地下物流系统网络**

摘 要：

众所周知，目前我国主要的物流手段还是地上运输。由于我国人口众多，每日物流量巨大，再加上我国道路基础建设不完善，使得本来就拥堵的交通状况更加恶劣。恶劣的交通状况不仅给人们的出行来带不便，还会造成巨大的经济损失和资源浪费。不仅是我国，世界范围内的各个国家都面临物流带来的交通压力问题。为了改善地面交通的状况，减小物流对地面交通的影响，各国科学家纷纷提出了地下物流系统（Underground Logistics System——ULS）的概念。

地下物流系统类似于地铁，运用地下通道实施城市内或城市与城市之间的物流运输。地下物流系统是一个高难度，高复杂程度的系统，它涉及到地上地下的运输、运输点的选取、地下线路建设与已有地下设施的冲突等问题。同时，地下物流系统还是一个高造价的系统，在设计和建造过程中需要严格把控成本。

本文研究了地下物流网络系统的构建，并以最优化理论为基础，给出了一套通用的设计地下物流系统的方法，即在给定任意的物流需求分布后，都可以用我们的方法找到一个覆盖所有地区、成本相对较低、可大幅降低地面交通拥堵情况的地下物流系统设计方案。在研究过程中，我们由浅入深，由局部到整

体地分析并解决了地下物流节点的设置、物流节点之间通道建设等问题，使用聚类分析得到地下网络节点的合理架设位置，同时利用最优化和贪心法的思想对网络节点的连接进行了最有效的分析，保证满足所有条件的情况下成本最小。最后通过程序进行模拟，证明了我们方法的可行性。

我们的工作主要有三点贡献：1. 对于成果甚少的地下物流系统研究进行了积极的探索和尝试。 2. 给出了一套有效的方法，针对不同地区的不同物流需求进行地下物流系统的设计。 3. 证明了地下物流系统的可行性和有效性。

关键字：地下物流系统 最优化理论 聚类分析 道路交通改善 最小生成树

I. 目录

1. 背景介绍......6
2. 任务定义......7
3. 模型假设及符号说明......7
3.1. 符号说明......7
3.2. 模型假设......7
4. 解决方案......8
4.1. 问题一思路及解决方案......8
4.1.1. 问题描述......8
4.1.2. 模型建立与求解......8
4.1.3. 解决思路的拓展......12
4.2. 问题二思路及解决方案......12
4.2.1. 问题描述与解决思路......12
4.2.2. 简化问题......13
4.2.3. 模型建立......13
4.2.4. 解决方法......14
4.2.5. 计算结果与分析......15
4.3. 问题三思路及解决方案......16
4.3.1. 问题描述与解决思路......16
4.3.2. 解决方案......16
4.4. 问题四思路及解决方案......18
4.4.1. 问题描述......18
4.4.2. 模型建立与求解......18
4.4.3. 扩容分析和方法......19
5. 总结和展望......20
6. 参考文献......20
附录......21

A. 各地区所需要转入地下物流系统的流量 21

B. 一级区域之间的通道设计以及通道的实际流量表 22

构建地下物流系统网络

1. 背景介绍

交通拥堵是世界大城市都遇到的“困局”之一。 2015 年的全球最拥堵城市排名中，我国大陆有十个城市位列前三十名。交通拥堵的危害远比我们想象中的严重，除了给人们出行带来不便，还有可能给国家和人民造成巨大的经济损失。据中国交通部 2014 年发布的数据，我国交通拥堵带来的经济损失占城市人口可支配收入的 20%，相当于每年国内生产总值(GDP)损失 5-8%。同时大量研究表明：“时走时停”的交通导致我国原油消耗占世界总消耗量的 20%。2015 年城市交通规划年会发布数据显示：在石油消费方面，我国交通石油消费比重占到了消费总量的 54%，交通能耗已占全社会总能耗 10%以上，并逐年上升。除此之外，交通拥堵带来的额外高能耗也带来了高污染和高排放。

导致城市交通拥堵的主要原因是交通需求激增所带来的地面道路上车辆、车次数量巨增，其中部分是货物物流的需求增长。尽管货车占城市机动车总量的比例不大，但由于货运车辆一般体积较大，启动、行驶都较缓慢，往往在造成道路拥堵上占有重要原因。按常规的车辆换算系数（不同车辆在行驶时占用道路净空间的程度），货运车辆所占用的道路资源达 40%。可想而知，如果能够减少地面道路的货运车数量，交通拥堵情况将得到巨大改善。另外一方面，大量实践证明，仅通过增加地面交通设施来满足不断增长的交通需求，既不科学也不现实，地面道路不可能无限制地增加。因此，必须通过其他方式来代替地面的货运车辆进行物流运输。为此，各国的科学家都纷纷提出了地下物流系统（Underground Logistics System——ULS）的概念[1][2][3][4]。

地下物流系统是指城市内部及城市间通过类似地铁的地下管道或隧道运输货物的运输和供应系统。图 1 展示了地下物流系统的基本构造。它不占用地面道路，减轻了地面道路的交通压力，从而缓解城市交通拥堵；它采用清洁动力，有效减轻城市污染；它不受外界条件干扰，运输更加可靠、高效。地面货车的减少同时带来巨大的外部效益，如路面损坏的修复费用，环境治理的费用等等。然而地下物流系统的研究与实践还刚刚兴起，全球范围内尚无成熟、成功的案例可供借鉴，并且地下物流系统结构复杂，造价高，风险大，所以在建造前，必须经过细心的设计和多方面模拟测试。

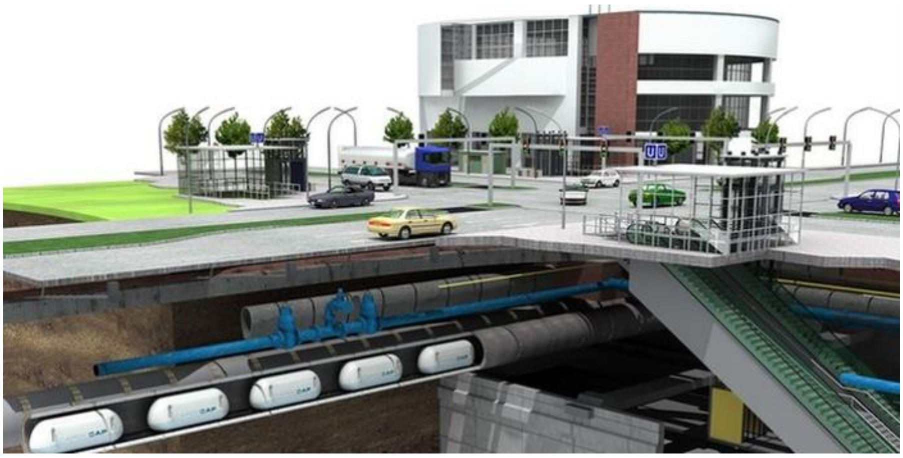


图 1. 地下物流系统示意图

在本文研究的问题中，地下物流系统由运输节点和运输通道两种主要元素构成，系统中可以运行两种车辆，分别载重 10 吨和 5 吨。另外，节点与节点之间可以建造两种

通道，分别是双向双轨通道和双向四轨通道。节点又分为一级节点和二级节点，两种节点的区别主要在于造价不同，与地面进行货物交互的能力不同以及连接方式不同。具体来说，一级节点之间可以通过运输通道互相连接，二级节点只能与一级节点连接，也就是说二级节点之间运输货物只能通过一级节点之间的连接进行。一级节点与其相连的二级节点共同构成一级区域。

文章的后续部分将如下安排：我们将在第二章中详细定义构建地下物流系统的任务，并叙述各个步骤和子任务；在第三章中我们将叙述在整个解决问题的过程中的一些宏观假设并且给出本文中所有涉及到的符号及其含义，方便读者查阅；我们将在第四章中针对定义好的问题和任务进行数学建模并提出解决方案；在第五章中我们将总结我们的工作，并分析方法的不足及指出未来改进的方向。

2. 任务定义

一个完整的地下物流运输系统主要由两部分构成：与地面进行交互的运输节点以及在地下进行运输的运输通道（包括运输车辆）。建造地下物流运输系统前我们应该设计好各运输节点的修建位置以及节点之间通道的连接情况。

另外，由于建设整个地下物流系统需要花费很长的时间，为了尽快将系统投入使用并改善地上交通状况，应当将整个系统分块建造，并且将建好的部分尽早投入使用。怎样制定建设计划使得在最短的时间内能够达到最优的地上交通改善，也是一个重要任务。由此，我们研究并解决了以下几个问题：

问题一：地下物流节点选取。

问题二：地下通道网络设计。

问题三：从全局出发进行系统改进。

问题四：建设时序设计。

3. 模型假设及符号说明

3.1. 符号说明

M_i	为满足交通基本畅通的需求，地区 i 所要转入地下物流系统的货物收发最小总流量
R_i	经线性模型计算后，地区 i 所要转入地下物流系统的货物收发总流量
X_{ij}	地区 i 经由地下物流系统发送到地区 j 的流量
$Dist_{ij}$	i 所代表位置与 j 所代表位置之间的距离
Y_{ij}	一级区域 i 从地下发往一级区域 j 的货物量
$I(x)$	指示函数，当 x 为 0 时返回 0，否则返回 1

3.2. 模型假设

在本问题中，为了简化计算，进行了如下假设：

1. 由于地区货运量与地区面积之比都比较小（大概在10^{-3}数量级），故假设地下运输节点的服务范围覆盖地区中心时视为覆盖了整个地区。

2. 假设货物从地下运输节点到地上之后使用小型车辆或人力进行运输，不影响城市的拥堵程度。

3. 不考虑地下运输通道之间因为空间有限而导致的交叉或冲突。

4. 地区的拥堵指数只与地区的货运量相关。

4. 解决方案

4.1. 问题一思路及解决方案

4.1.1. 问题描述

问题一要求对地下物流节点进行选择，使得各地区的地面交通状况至少达到基本畅通（拥堵指数达到 4 以下），同时使得进出 4 个物流园区的货物尽最大可能放入地下运输。

4.1.2. 模型建立与求解

4.1.2.1. 解决思路

为解决各个地区的交通问题，我们从各个地区的货物出入量情况着手进行分析。通过各地区拥堵指数和其货运总量，可以计算出它们为达到交通基本畅通所需要转入地下运输的货物出入总量，由此知道了各个地区的地下物流流量的最小需求。只要建立一系列运输节点，使其能够负荷所有这些转入地下的流量，则各个地区的交通问题都能得到改善。

通过上面的计算我们可以获得各个地区需要转移到地下进行运输的最小物流流量，但是各地区之间的物流运输是相互关联的，因此我们还需要进一步计算地下物流网络中每个地区与其他地区之间的具体物流流量。由于每个地区所需转入地下的最小物流流量已经求出，因此可以将最小流量需求作为问题求解的限制条件。计算各地区需要转移到地下运输的物流流量可以视为一个最优化[5]问题。

为使得进出 4 个物流园区的货物尽最大可能放入地下运输系统，我们把进出 4 个物流园区的货物总量作为优化目标，结合前面提到的限制条件，我们得到了一个最优化模型，求解该模型，可以得到地区与地区间的地下物流收发流量，使得各地区的最小流量需求得到满足，同时使得进出 4 个物流园区的货物量尽可能大。

因此，解决该问题可以分两步进行，也就是先通过求解一个最优化问题得到地下物流系统的物流矩阵，接着再根据求解出来的地下物流流量矩阵，使用改进的聚类模型对地区进行聚类，最终确定一二级节点的位置以及服务地区。最优化模型和聚类模型的详细内容将在下面两节给出。

4.1.2.2. 最优化模型的建立

由地图可知，图中共有 4 个物流园区和 110 个地区，为了描述方便，我们将 4 个物流园区的集合记为A_{lp}（area of logistic park），把非物流园区地区集合记为A_{city}。由地区 i 经由地下物流系统发到地区 j 的货物流量记为X_{ij}，地区k（根据提设

不考虑四个物流园区）的最小地下物流需求（为满足交通基本畅通需求该地区的货物收发总流量）记为$\mathrm{M_k}$，进出 4 个物流园区的总货物量记为T。

则根据交通基本畅通、地区发送到地区的流量应大于等于零这两个条件可得如下的最优化问题：

$$\mathrm{argmax}_{\{X\in[0,+\infty]\}} \sum_{\mathrm{i}\in \mathrm{A_{lp}}, j\in A_{city}} X_{ij} + X_{ji}$$

$$\mathrm{s.t.} \quad \sum\nolimits_{\mathrm{j}\in\{\mathrm{A_{city}},\mathrm{A_{lp}}\}} \left(X_{ij} + X_{ji}\right) \geq M_i \text{ , } \mathrm{j} \in \mathrm{A_{city}}$$

$$\mathrm{X_{ij}} \geq 0, \qquad i,j \in \{A_{city}, A_{lp}\}$$

4.1.2.3. 最优化模型的求解优化

由上一小节可知，在模型的限制条件下最大化目标函数T可以通过利用最优化库进行求解，考虑到该模型的变量数目相对较多，求解时间可能较长，因此我们对该模型进行简化。

决策变量、约束条件和目标函数是最优化问题的三个要素。在本问题中，由于决策变量的个数是由地区到地区间所有可能的边数决定的，而约束条件限制使得交通拥堵情况应达到哪种程度，优化这两个要素可能使得模型不满足条件，因此我们着眼于从目标函数 T 对模型的求解复杂性进行优化。

对于最优化问题，目标函数的变量数目越大，则问题的求解复杂度越大，因此，我们考虑减少目标函数中的变量数目去求得一个目标函数的近似解。

我们的目的是使得由园区经由地下物流系统到非园区地区的货物总流量最大，而由题设已经知道园区与非园区地区间的实际流量，直观上看，两地区间的实际流量越大，那么应该转移到地下系统进行运输的流量也应越大，原因有两个，第一，实际流量越大，拥堵程度高的可能性越大，因此需要转入地下的流量也应越大。第二，地下物流系统的建设成本较高，因此，它所连接的两个地区间若没有足够的实际流量那将造成极大的浪费，因此实际流量大的两地区间建立有地下物流通道的可能性大，转入地下的流量也应越大。

因此，我们可以筛选出地区间实际流量最大的前 n 个，把它们所对应的地下物流流量相加作为我们的优化目标，这样我们将计算出满足已知约束条件下的一个近似解。该解所需要的时间复杂度远远小于最优解的时间复杂度，且是给定时间条件下趋于最优的一个解[1]。

4.1.2.4. 聚类模型的建立与求解

由最优化模型可以求解得到符合题设要求的一个地下物流矩阵 X，根据这个矩阵我们可以计算出每个地区所需要转入地下物流系统的流量。把地区 i 所需要转入地下物流系统的流量记为$\mathrm{R_i}$，则$\mathrm{R_i}$满足：

$$\mathrm{R_i} = \sum_{j\in\{A_{city}, A_{lp}\}} (X_{ij} + X_{ji})$$

[1] 最终求解得到的地下物流系统物流矩阵尺寸为 114×114，由于太大无法展示，将与代码包一起提交。

可以将 R 对应到地图上显示出来，部分地区的中心点分布以及所需要转入地下物流系统的流量如图 2 所示。

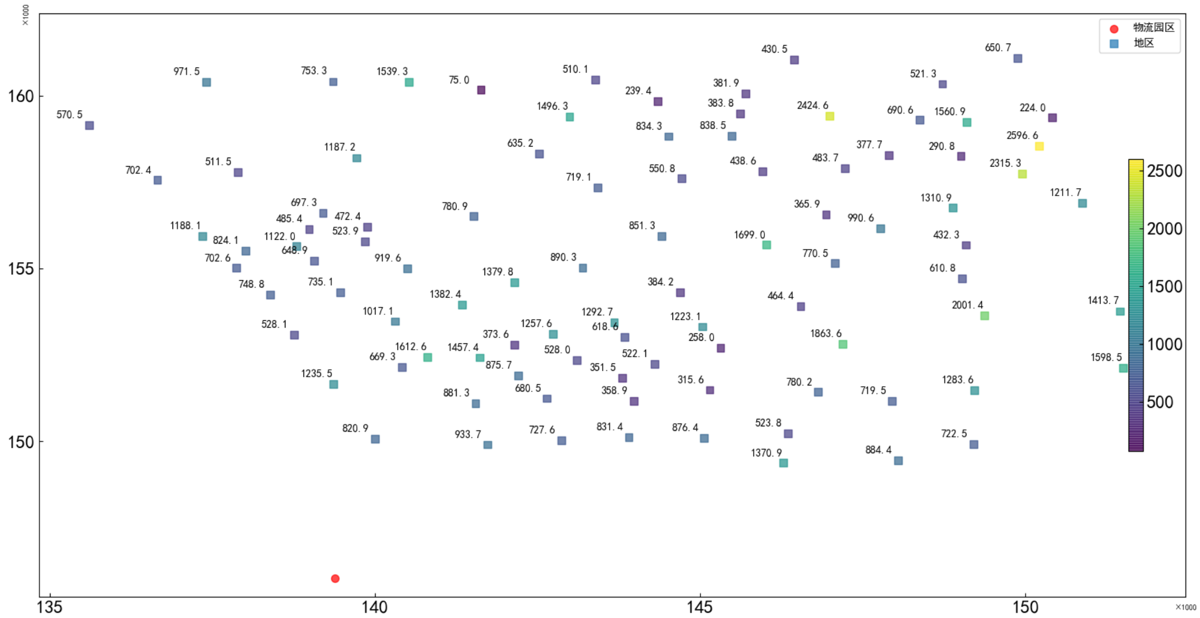


图 2. 部分地区中心点分布及需转入地下的最小流量

所有地区的中心点分布以及所需要转入地下物流系统的流量图在附录 A 中。接下来考虑节点的数量，位置和服务范围。每个地区需要转入地下的物流需要由一级或者二级节点代为转入或者转出地下，因此，如果每个节点能服务更多的地区，那么总的节点数将更少，从而所有节点的建设成本将更低。但是，每种节点从地面收发货物的总量是有上限的，根据题设，一级节点从地面收发货物总量为 4000 吨，二级节点的为 3000 吨。

节点的位置和服务范围可以通过聚类算法在图一上进行聚类而得到。常用的聚类算法是 k-means 算法[6]。k-means 算法源于信号处理中的一种向量量化方法，现在则更多地作为一种聚类分析方法流行于数据挖掘领域。k-mean 算法的目的是：把 n 个点（可以是样本的一次观察或一个实例）划分到 k 个聚类中，使得每个点都属于离他最近的均值（此即聚类中心）对应的聚类，以之作为聚类的标准。这个问题将归结为一个把数据空间划分为 Voronoi cells 的问题。

但是，前面提到节点从地面收发货物的总量是有上限的，也就是，节点的服务范围内所需从地面收发货物的总流量不能超过特定值。传统 k-means 算法不能解决这个问题，因此，我们拓展了 k-means 算法来作为我们的聚类算法，算法描述如下，该如何使用这个算法得到最终的聚类结果将在下文详细描述。

输入: 各地区的中心点位置及最小流量值，聚类的数目 k，聚类区域最大总流量 f，最大迭代次数 m
输出: 是否分类成功，如果是，同时输出核心点的位置以及由其聚合起来的地区的中心点位置集合

1: 在图中撒 k 个核心点，点的位置随机;
2: 对于每个核心点，对应到一个它所聚合的地区点集合，初始化为空集合;
3: 对于每个核心点，计算离它最近的未被其他核心点聚合的地区点，若该地区点距离核心点的距离小于 3000，并且该地区点加入该核心点对应的聚合地区点集合后，集合中流量值之和小于 f，则加入，否则，进入下一步;
4: 对于每个核心点，计算其对应的聚合地区点的位置的几何中心位置，并将核心点的位置更新为该几何中心位置;
5: 如果迭代次数大于 m，输出聚类失败，算法结束，否则，进入下一步;
6: 如果存在地区点没被聚合，回到第 3 步，否则输出聚类成功，并输出最终聚类数目，各核心点的位置以及由其聚合起来的地区的中心点位置集合，算法结束。

算法 1

由于不知道算法的 k 值应该取多少，因此我们从一个较大的值开始（初始化为图中点的总数目），接着利用二分查找算法去逼近 k 的合适取值，从而得到我们最终的聚类结果，二分查找 k 值的过程描述如下：

输入: 聚类数目的最大值 n，各地区的中心点位置及其最小流量值，聚类区域最大总流量 f，最大迭代次数 m
输出: 输出最合适的聚类数目，同时输出核心点的位置以及由其聚合起来的地区的中心点位置集合

1: 初始化最小聚类数目 l=1，最大聚类数目为 r=n，当前聚类数目为 c=(l+r)/2;
2: 利用改进的 k-means 算法（即算法一），k 值设置为 c，执行该算法;
3: 如果算法一结果为聚类成功，则更新 l=c，否则，更新 r=c;
4: 如果 m 不等于 l，更新 m=(l+r)/2，回到第 2 步，否则，进入下一步;
5: 输出最近一次成功聚类的 k 值，同时输出核心点的位置以及由其聚合起来的地区的中心点位置集合，算法结束。

算法 2

聚类算法 1 在进行聚类的时候，会限制节点服务范围内的转入地下的货物总流量不超过 4000 吨，因此，最终出来的聚类结果符合前面提到的节点从地面收发货物的上限限制。

利用算法 1 和 2，可以得到最终的节点数量，位置和服务地区。必须指出，我们这里把节点由地面收发货物的上限定为 4000，也就是一级节点的上限，这样做的好处是使得最终的类别数量更少，从而降低建设成本。

当聚类完成后，我们计算每个节点服务地区内从地面收发货物的实际总流量，对于流量超过 3000 吨的节点，我们把它视为一级节点，对于流量在 3000 吨以下的节点，我们把它降级为二级节点。

利用改进的 k-means 算法，二分查找 k 值算法和一二级节点划分准则，我们得到了最终的一二级节点数量，位置和服务范围，最终结果如图 3 所示，方块点表示地区的中心，圆圈代表每个节点的服务范围，圆的圆心就是节点的位置，另外，一二级节点

由不同颜色区分，一级节点的服务范围为红色圆圈，二级节点的服务范围为蓝色圆圈。

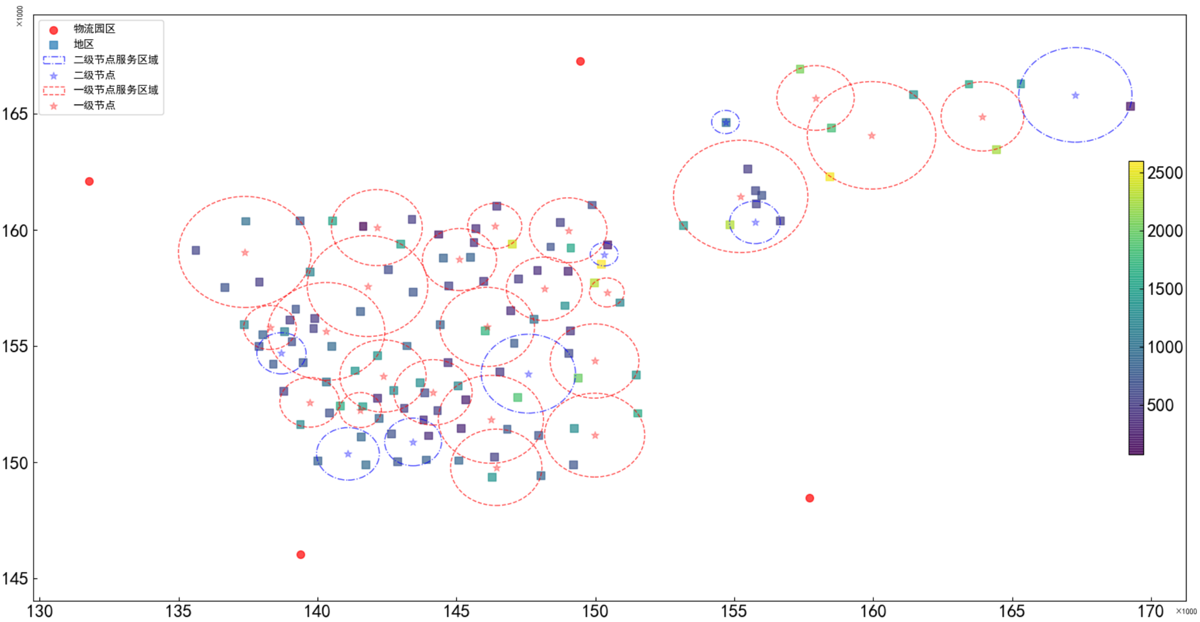


图 3. 一级节点和二级节点的分布

4.1.3. 解决思路的拓展

前面一节给出了问题一的一个很好的解决方案，即是通过建立一个最优化模型和一个聚类模型在满足限制条件去对地区进行聚类，也已经给出了一个比较好的聚类结果。在这一节，我们描述一下我们在解决第一问过程中的另外一些有趣的思路，作为问题一解决思路的拓展。

考虑到问题一有两个目的，一个是使得各地区的交通达到基本畅通，另一个是使得进出 4 个物流园区的货物尽最大可能放入地下运输。通过对数据进行分析，我们得到一个有趣的结论，也就是，四个物流园区的货物收发总流量占所有货物总流量的四分之一以上（经计算约为 25.175%）。因此，如果我们把所有进出园区的流量都转移到地下物流系统，那么将同时在问题一的两个目标上都有很大的收益。由于有四分之一的流量被转入到地下，因此各个地区的交通拥挤情况将得到极大改善。另外，因为把进出四个园区的货物全部放入了地下进行运输，因此，第二个目标也就得到了满足。

但是，考虑到这种解决思路先入为主地把所有园区流量转入地下，带有较强的前提假设，因此我们最终采用了上一节中所提到的更为合理的方案并且也求出了较好的结果。本节的思路由于时间关系没有进行实验验证，因此在此仅作为另一种解决思路提出来。

4.2. 问题二思路及解决方案

4.2.1. 问题描述与解决思路

在问题一中我们已经对地下物流节点进行了选取和分类，接下来我们要在已有的一级节点和二级节点或一级节点之间建立运输通道。运输通道的建立要保证每天需要运输的货物能够全部运输完成，在此基础之上要让运输和通道建设成本尽可能最小。我们同样可以用最优化模型来建模此问题。

4.2.2. 简化问题

由题意知运输成本与货物的运输量与运输通道的长度相关，通道的建设成本与通道的类型和通道长度相关。对于二级节点，由于其只能与一个一级节点相连，即每个二级节点连接的通道数必定为 1。因此，与二级节点相连的通道货物运输量只取决于二级节点的运输量，只需找到与二级节点距离最近的一级节点并与之相连，就可以保证这条通道的运输成本和建设成本最小。

这样一来，问题便简化为一级节点与它们对应的二级节点所构成的一级区域之间的通道建立问题。每个一级区域的货物流量等于一级节点本身的货物流量加与其相连的所有二级节点的货物流量的总和。我们将物流园区视为一种特殊的一级区域，一同参与建模。

4.2.3. 模型建立

由上一问的结果可知共有 23 个一级区域，加上 4 个物流园区，共有 27 个一级区域，各一级区域的地下运货量为其所包含一级节点及二级节点之和。

我们设定任意两一级区域之间地下货物流量为变量，则变量可表示为 27×27 的矩阵 X，X_{ij}表示从一级区域 i 到一级区域 j 的地下运输量。如果X_{ij}和X_{ji}都为零，则表示i，j两个一级区域之间不需要有运输通道连接。我们需要优化的目标是每天的总成本（总成本=运输成本+折旧成本）。其中运输成本为 1 元/吨·公里，折旧成本包括运输节点的折旧成本一级运输通道的折旧成本，年综合折旧率为 1%。表 2 展示了不同运输通道和不同节点的造价。

运输通道	双向四轨（10 吨）	5 亿元/公里
	双向四轨（5 吨）	3.5 亿元/公里
	双向双轨（10 吨）	4 亿元/公里
	双向双轨（5 吨）	3 亿元/公里
	一级节点	1.5 亿元/个
	二级节点	1 亿元/个

表 2. 不同运输通道和运输节点的造价

每两个一级区域之间的流量受它们之间的运输通道最大运输能力限制。对于物流园区与一级节点之间的通道，使用双向四轨通道，每天运营 18 小时，每小时发 5 量 8 节载重 10 吨的车辆，每天最多可运输 18×5×8×10×2=14400 吨货物。对于非园区和一级节点之间的通道，由于只能使用 5 吨载重的车辆，所以每天最多运输 7200 吨货物。

根据以上知识，我们可以将运输通道网络的设计抽象为解决如下的最优化问题：

$$\mathrm{argmin}_{\{Y\in[0,+\infty]\}} \sum_{i,j\in\{A_{city},A_{lp}\}} Y_{ij}Dist_{ij} + 0.05I(Y_{ij})Dist_{ij}/365$$

$$\text{s.t.} \quad Y_{ij} \le 7200 \ \ (ij \in A_{city})$$

$$Y_{ij} \le 14400 \ \ (i \in A_{lp}, j \in A_{city})$$

$$Y_{ij} \le 14400 \ \ (i \in A_{city}, j \in A_{lp})$$

$$Y_{ij} \ge 0 \ \ (i,j \in \{A_{lp}, A_{city}\})$$

$$\sum_{j\in\{A_{city},A_{lp}\}} Y_{ji} = \sum_{j\in\{A_{city},A_{lp}\}} X_{ji} + \sum_{j\in\{A_{city},A_{lp}\}} Y_{ij} \quad (i,j \in \{A_{lp}, A_{city}\})$$

由于此优化问题变量较多，且指示函数不是连续可导函数，故无法对此问题进行直

接优化。同样地，我们采用另一种基于贪心法的算法来求此问题的近似解。

4.2.4. 解决方法

在 4.2.2 中我们已经讨论过，由于二级节点只能通过一级节点和其他地区进行货物运输，所以要把每一个二级节点连在某个一级节点上，根据地下流量表所知，所有二级节点的收发货物总量都小于 3000 吨，完全可以在一个双向双轨或者双向四轨的的隧道中进行运输，则每一个二级节点只需要连接一个一级节点。为了保证最终成本最小，则二级节点应该连接距离最近的一级节点上。

二级节点连接后我们可以知道每个一级节点所能服务的地区（包含自己的地区和连接的二级节点对应的地区），为了保证所有所有地区都能够进行货物运输，则要保证所有地区地下连通，在不考虑四个园区的情况下可以把整个一级节点看成全连通图，而要保证建设总成本最小同时所有节点连通，则求出这个全连通图的最小生成树即可。再求出的最小生成树中再判断通道流量限制，对于流量超载的通道进行贪心选择另外的节点进行连接。具体算法如下：

输入: 所有节点的位置，地区之间的货物运输量，节点覆盖地区的信息

输出: 节点的网络连接，总成本

1: 对于所有的二级节点连接距离它最近的一级节点;
2: 更新所有一级节点的服务地区;
3: 选择标号为 1 的一级节点为初始点，选择距离最近的一级节点连边;
4: 当前选择节点数目加一，再计算距离已有节点最近的节点并连边;
5: 重复上一个步骤，知道所有一级节点都被选择;
6: 对于每一个节点，计算节点成本，节点实际货运量，节点的通道成本，通道的实际流量;
7: 节点的实际流量为当前结点需要从地上转入地下的所有流量，根据为一级或者二级节点计算成本;
8: 对于节点通道流量，计算此节点服务的所有地区，运输以连接节点根的子树所对应的所有城市的货运量，根据通道流量选择双向双轨或者双向四轨的物流隧道进行建设。

算法 3

现在所有的非园区的网络已经设计完成，再考虑园区的通道。首先为了保证转运率的要求，每个园区都与且仅与最近的一个一级节点相连。首先园区到一级节点的双向四轨地下隧道最大能够支撑 14400 吨的货物运输，考虑进出 4 个园区的货物尽可能最大的放入地下运输，通过最优化的解法得到园区和节点的货物运输量最大不超过 10000，所以一条双向四轨地下隧道完全能够满足运输需求，而需要考虑的就是园区和最近的一级节点连接之后，地下货物运输量增大之前节点之间的通道运输是否还满足流量需求。

这样把问题转化为：有四个源点分别连接 4 个一级节点，节点之间是连通的，通过扩大隧道为双向四轨或者新增加一条边使得所有通道的运输货物量不超过上限。转化为一个搜索剪枝的问题，对于每一个节点有以下两种选择：1）将已有的双向双轨变成双向四轨，2）这条边不变，再找一条距离最短的边建一条双向双轨的隧道。因为这两种方法所能提供的增加的货运量是一样的，但是第一种成本更低，所以优先选择第一种方法。这样对于下一个节点类似进行考虑。因为初始点的选择会影响隧道的建立方式，同时影响成本，所以遍历不同初始点选择方式一共是$2^4 = 16$种，时间复杂度完全可以接受。

4.2.5. 计算结果与分析

网络构成如图 4 所示，红色原点表示园区，红色星星一级节点，蓝色星星表示二级节点，每个园区分别连接在距离最近的一级节点上，二级节点同样连接在最近的一级节点上。其他的一级节点构成一颗最小生成树保证所有节点连通且建设和运输成本最小。

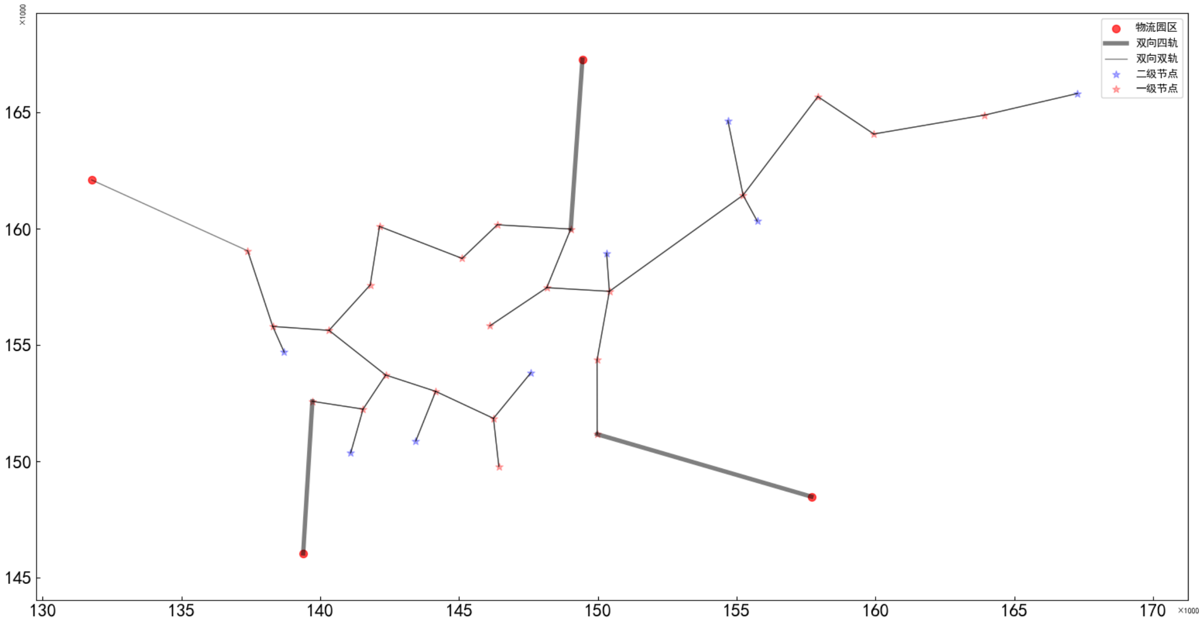


图 4. 运输节点之间的连接结构

四个园区连接的一级节点的位置，以及通道之间的位置和通道的流量如表 3 所示[2]。由于节点个数过多，只显示部分数据，其中 0 表示两个节点之间没有直接通道相连，否则表示某一个通道的实际流量。

园区通道	连接节点	通道流量	成本							
1	2	9225.773	211944.4							
2	11	9116.875	267147.6							
3	31	9712.575	234949.4							
4	24	4156.326	142843.9							
节点货运量（吨）	2949.897	3449.976	3285.376	2818.79	3881.8	985.3188	……	3620.662	3737.499	3423.63
通道流量（吨）	1	2	3	4	5	6	……	29	30	31
1	0	0	0	0	0	0	……	0	0	0
2	0	0	0	0	0	0	……	0	0	0
3	0	0	0	0	0	0	……	383.9338	0	0
4	0	0	0	0	0	0	……	0	816.948	0
5	0	0	0	0	0	0	……	0	91.1208	0
6	0	0	0	0	0	0	……	0	33.78838	0
7	0	0	0	0	0	0	……	0	0	0
29	0	0	109.2955	0	0	0	……	0	0	0
30	0	0	0	44.71132	38.40291	0.850705	……	0	0	0
31	0	0	0	0	0	0	……	0	0	0

表 3. 部分通道之间流量

计算得到园区和一级节点连接的成本和运输成本为 880149.15 元/天，节点成本以及之间连接成本和运输成本为 931307.17 元/天。那么总成本为 1811456.32 元/天

[2] 完整的数据由于太大无法在论文中展示，在附录 B 中给出截图。

（181 万/天）。这样即可以保证总成本最小，又能够保证所有通道的流量上限不超过最大可通过流量。

最终节点和通道的构成如图 5 所示。

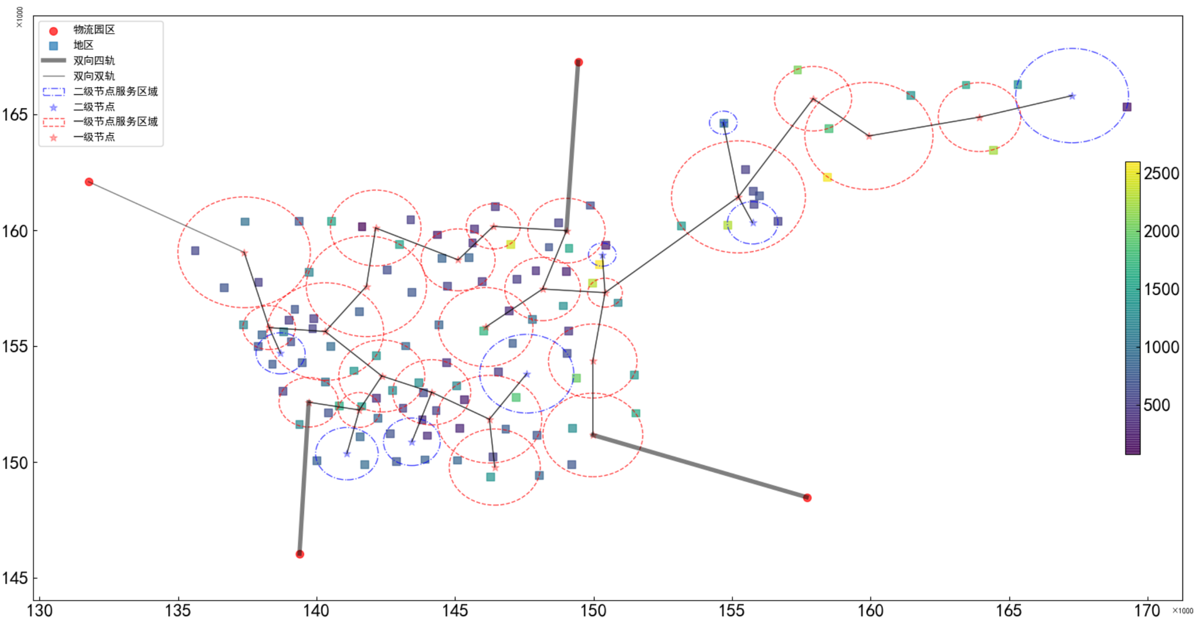


图 5. 最终节点与通道分布图

4.3. 问题三思路及解决方案

4.3.1. 问题描述与解决思路

我们知道地下物流系统主要由两部分构成，地下运输节点和地下运输通道。在前两个问题中我们将这两个部分分开进行设计，然而在实际建造过程中，确定运输节点与设计运输通道之间有一定的互相制约关系。

举例说明，要想使物流园区到一级节点的地下运货量尽可能大，需要考量一级节点之间运输通道的分布情况，以确保与物流园区连接的一级节点有足够的能力将货物继续运输下去。如果只有一个一级节点与物流园区相连，而此一级节点又只与一个一级节点之间存在运输通道，那么所有物流园区的货物流都将经过这两个一级节点之间的通道。而一级节点之间通道的最大运输能力低于物流园区和一级节点之间通道的最大运输能力，因此在没有掌握到全局信息的情况下很容易出现运输通道超负荷的情况。

直观上的最好解决方案便是在设计初期同时对运输节点和节点之间的运输通道进行建模。

4.3.2. 解决方案

对于在设计初期便对运输节点和运输通道同时进行建模的想法，我们从可行性和复杂性两个角度来考量。

4.3.2.1. 可行性

从问题一中我们已经得知可以使用最优化模型来建模并解决各地区货物的地上运输和地下运输的分布，但是对于一级节点和二级节点如何分布的问题却不能用同样的办

法进行建模。原因是一级节点和二级节点的数量都是可变量，没有办法对运输节点的数量和坐标同时设置变量并求解。

由此，我们无法简单地将整个地下运输系统的建造抽象为一个最优化问题，但是由于我们已知运输节点总数是有上界和下界的（0 到地区总数 110），我们可以先固定节点数量，然后通过迭代建模和优化，对节点数量使用二分查找法确定使目标函数达到最优的节点数。

4.3.2.2. 复杂性

通过对问题进行些许变形，我们可以对整体问题进行建模和求解，接下来分析这种模型在求解时的复杂度。

我们需要对节点之间通过地下运输的货物量设置变量、对各地下运输节点的坐标设置变量以及对运输节点之间的通道流量设置变量，这使得变量数目巨大。对于本问题，在最差情况下有10^4数量级别的变量个数。巨大的变量个数将导致优化算法的效率急剧下降，在我们经过多次试验后粗略估计，使用序列二次规划法[7][8]解决此优化问题所花费的时间复杂度约为$O(n^4)$。

综合以上结论，我们可以得出直接对整体问题进行建模求解的想法并不现实，更加可行的办法是先分步确定运输节点和运输通道，然后对分两步解决得到的结果运用全局信息进行分析优化。虽然这种方法无法得到整个设计方案的最优解，但是可以在可容忍的运算时间内得到一个较好的局部最优解。

举例说明，图 6 中展示了使用分部求解的部分结果，我们可以发现，如果将图中的 12 号运输节点改为一级节点，并将物流园区与 2 号节点的通道改为与 12 号节点相连，可以降低每天约 16182 元的成本。

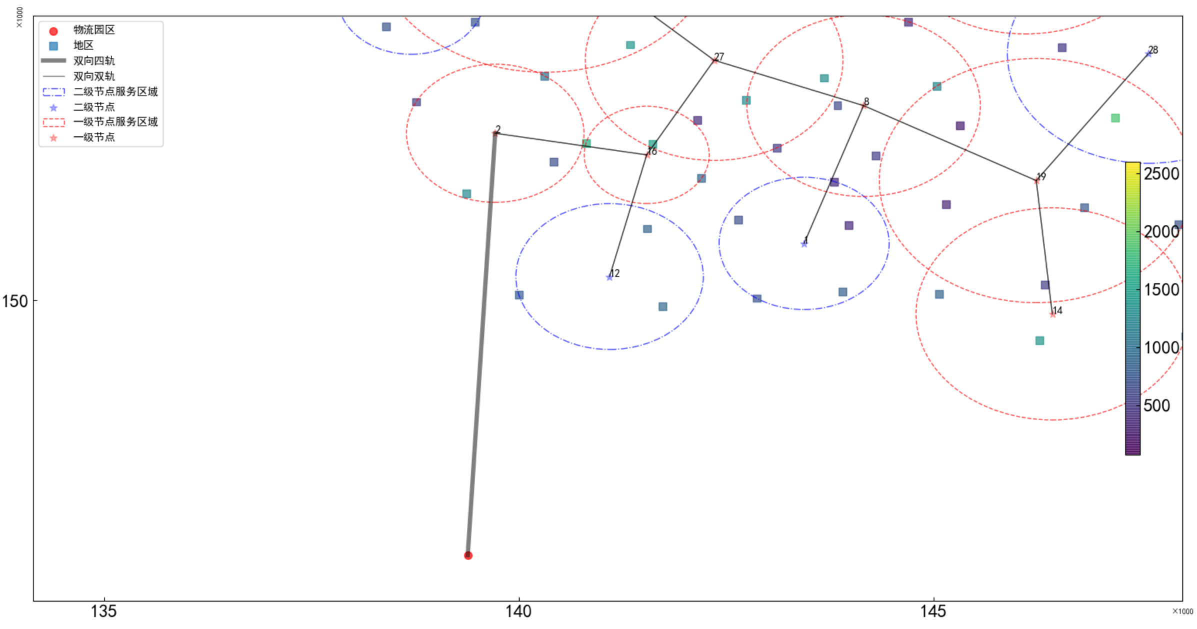


图 6. 部分节点与通道

4.4. 问题四思路及解决方案

4.4.1. 问题描述

考虑到地下物流系统造价高风险大，问题四的主要目的在于做好顶层设计，要求根据建设进度分八年完成地下物流系统的建设，并给出网络各个线路的建设时序及演进过程。

4.4.2. 模型建立与求解

4.4.2.1. 解决思路

网络边的建立是循序渐进的，早建设完毕的边就可以早投入使用产生收益，为使得经济效益最大化，我们可以从以下两个方面考虑一条边在什么情况下该被优先建设。第一，在边长度相同的条件下，如果一条边，建立之后能在原有网络的基础上容纳的流量的增量越高，则它越应该被优先建设；第二，在边建设之后能产生的增量收益相同的条件下，如果一条边，建立的时间越短，则它越应该被优先建设。

综合考虑以上两个应优先建设某一边的情况，我们可以用利用贪心算法来建立起我们的模型。我们把总收益设定为由第一年开始到第八年末该动态网络所能容纳的总流量。每当要建设一条边，我们去计算每一条未被建设的边若投入建设将产生的收益，然后按照收益排序，收益最高的那条边即作为我们要建设的下一条边，以此类推，直到所有边被建设完毕，那么网络各个线路的建设时序及演进过程也就明了了。详细的模型在下一节描述。

4.4.2.2. 模型建立

由于模型的建立依赖于贪心算法之上，因此最主要的任务就是如何去计算建设一条边将产生的增量收益。我们把建设一条边的增量收益设定为该边建设完成后相比于原网络在第八年末之前所能增加的收益。下面将对建设一条边p的增量收益进行推导。

记我们所设计的最终网络为G，当前已建设完毕的网络为G′（未建设完全的边图，不一定连通），目标是求建设边p的增量收益。

首先我们可以求出G中边的总长度，记为L，同样可以计算出G′中的边总长度，记为L′，记边p的长度为Lp。那么该边由建设完成到第八年末一共可以投入使用的年限t为

$$\mathrm{t}=\left(1-\left(\mathrm{l}^{\prime}\ +\mathrm{L_p}\right)\div \mathrm{L}\right)\times 8$$

接着，我们可以去求在网络G′的条件下，每年所能处理的流量。考虑到在问题二中，我们给出的网络结构是一个树结构，因此对于其中任一条边，记为(u,v)，它承担着将由u点过来的所有流量转发到v，同时将由 v 点过来的所有流量转发到u的任务，而问题二的解决已经保证了该边的容量不会超过上限，因此，对于G的子图G′，它的每一条边的容量都不会超过上限，也就是每条边的容量都是够用的，只要两个节点间有边连接，那么它们之间的流量就可以到达彼此，不用考虑边容量不够的问题。因此，对于网络G′，它所能处理的流量就是G′中连通的所有点对之间的流量总和，由问题一种求出来的地下流量矩阵可以很方便求得，记G′每年所能处理的流量为a。同理，G′加上边p之后每年所能处理的流量也易得，记为b。

那么，在原来的网络G′的基础上建设边p产生的增量收益$\mathrm{S_p}$为：

$$\mathrm{S_p}=(b-a)\times t$$

因此，我们可以从一个没有边的原始网络开始，对每条边计算其增量收益，取到最大增量收益的边即是下一条投入建设的边，于是将其加入原始网络，重复该过程就可以得到网络各个线路的建设时序及演进过程。

4.4.2.3. 模型求解

上一节已经对我们如何求解增量收益进行了详细的介绍，这一小节主要介绍一下模型的求解以及结果的展示与分析。

我们将模型的求解过程以算法描述的形式展示如下：

输入： 最终建设网络 G，地下物流矩阵 U

输出： G 中边的建设顺序

1: 初始化已完成建设的网络 G’为空集合:
2: 对于每条在 G 中不在 G’中的边，计算它的增量收益，记录增量收益最大的边;
3: 将 2 中找到的增量收益最大的边加入 G’中;
4: 如果 G’不等于 G，回到第 2 步, 否则，进入下一步;
5: 输出构建 G’过程中边的建设顺序，算法结束。

算法 4

通过计算边的增量收益，利用贪心算法一步步构建边，我们就得到了边的建设顺序，换句话说，我们得到了网络各个线路的建设时序及演进过程。

4.4.3. 扩容分析和方法

选择一条线路进行分析，例如 2 号节点（一级节点）和 11 号节点（一级节点）之间的通道初始流量为：2->11:3506, 11->2:3305。则它们之间建立的是双轨双向的隧道。双轨双向隧道种 5 吨的载重车每天最大流量为：$5\times8\times18\times5=3600$吨（分别代表汽车载重、车厢数、每天运营时间、每小时的发车数）。按照每年 5%的需求增长，因为隧道的双向尺寸一致，并且以单向流量较大的来设计，所以假设隧道两端的需求同步按照每年 5%增长，则由于$3500\times1.05=3675$，大于最大流量 3600，所以一年之后这条通道就需要进行扩容处理。

扩容优先选择升级隧道，由双向双轨升级到双向四轨隧道，则隧道的流量上限增到到 7200 吨。因为一年就需要改建，代价太大，所以再初始设计的时候就应该把这条隧道设计成双向 4 轨的大容量隧道。接下来，$3500\times1.05^{15}=7276.25$ 大于双向四轨的最大上限，则再经过 14 年，又需要对轨道进行扩容处理，这时有两种考虑方式：1）增加一个节点，2）不增加节点，只增加一条边分担现有的容量。下面分别分析这两种方式。

首先增加节点后一定要增加边，所以增加节点的成本高于只增加边的成本。所以尽量采用向现在的节点架设一条新的地下隧道的方式。而考虑到地上和地下的货物收发总量有限制（一级节点为 4000 吨，二级节点为 3000 吨），当地上底下的交互大于总量限制时只能选择增加一个新的节点并连接到现有的节点上。对于选择一级节点还是选择二级节点是这样考虑的，如果这个节点的流量大部分是在这个一级区域（一级节点和它包含的二级节点所对应的所有地区）内进行转移的，则建设一个二级节点，这样既能够降低成本，又能够缓解流量压力，而如果大部分的流量都是这个一级区域和其他区域之间进行转移，这个时候就需要建设一个一级节点来对这个一级节点进行分流，保证隧道的流量不超过上限。

5. 总结和展望

本文对地下物流运输系统的设计方法进行了详细的分析，并将此问题分解为多个子问题逐一解决。对于大部分子问题，使用最优化模型进行建模求解。本文不仅对分步建设地下物流运输网络提供了有效且普适的解决方案，还对综合全局信息对建设方案进行优化以及建设时序等顶层设计进行了积极的探索和讨论。

即便如此，我们的工作仍有许多不足和可改进之处。例如，对于那些利用最优化模型建模的问题，我们大都使用一种近似算法去求局部最优解，这导致我们最终产出的方案并不是最优方案。在未来的工作中，我们需要继续深入探索更好的求解最优化问题的算法，不断探索，最终找到问题的全局最优解。

6. 参考文献

[1] 周婷, 周爱莲. 基于时间成本的地下物流配送路线优化模型. 物流工程与管理, 38(8): 60-62, 2016.

[2] 马祖军. 城市地下物流系统及其设计. 物流技术, 2004 (10): 12-15, 2004.

[3] Ou-long H, Dong-jun G U O, Zhi-long C. Design of the Distribution Center of Underground Logistic System with SLP Method. Chinese Journal of Underground Space and Engineering, 2(1): 1-4, 2006.

[4] Boerkamps J, Van Binsbergen A. GoodTrip - A new approach for modelling and evaluation of urban goods distribution//Urban Transport Conference, 2nd KFB Research Conference. 1999.

[5] Wikipedia, Mathematical Optimization, https://en.wikipedia.org/wiki/Mathematical_optimization, 2017 09 16

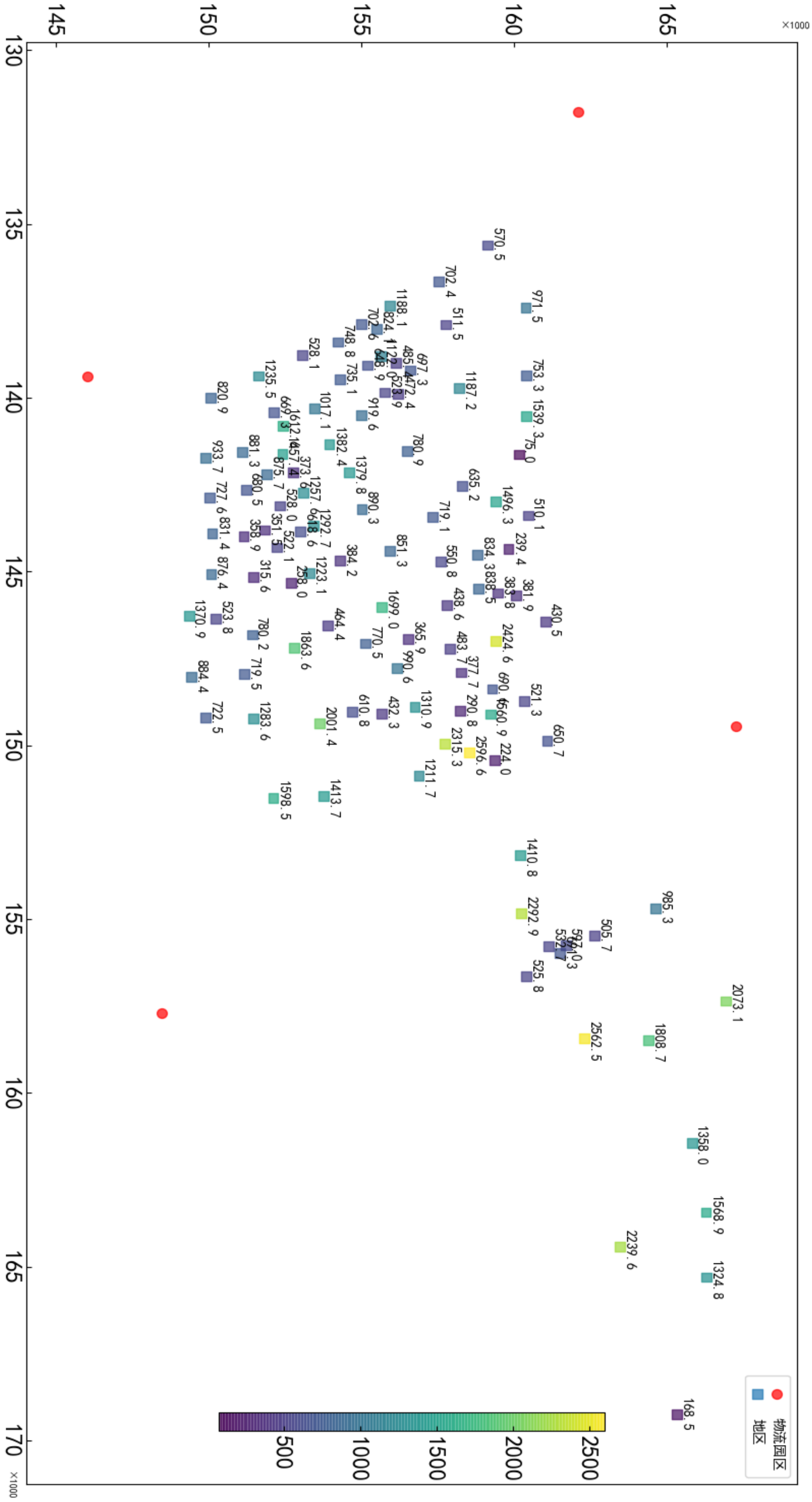
[6] Kanungo T, Mount D M, Netanyahu N S, et al. An efficient k-means clustering algorithm: Analysis and implementation. IEEE transactions on pattern analysis and machine intelligence, 24(7): 881-892, 2002.

[7] 钱令希, 钟万勰, 程耿东, 等. 工程结构优化设计的一个途径——序列二次规划 SQP. 计算结构力学及其应用, 1(1): 7-20, 1984.

[8] Nocedal J, Wright S J. Sequential quadratic programming. Springer New York, 2006.

附录

A. 各地区所需要转入地下物流系统的流量



B. 一级区域之间的通道设计以及通道的实际流量表

回区通道	连接节点	通道流量	成本																												
1	2	9225.773	211944.97																												
2	11	9116.875	267147.6																												
3	31	9712.5752	234949.36																												
4	24	4156.3265	142843.91																												
节点货运量（吨）	2949.8975	3449.9761	3285.3758	2818.7897	3881.7999	985.31883	3992.88	3603.6051	3619.6403	3453.6906	3604.6037	2635.8872	3527.0203	3655.5163	2820.569	3945.7049	2835.2838	1493.3054	3038.3774	3236.9407	3322.3443	3808.5281	3686.6963	3509.3059	3847.3907	3920.5682	3903.9137	2938.8976	3620.6615	3737.4987	3423.6302
通道流量（吨）	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	0	0	0	0	0	0	0	214.24613	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	3506.6326	0	0	0	0	7130.9862	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	257.89044	0	0	0	0	0	0	0	0	383.93385	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	816.94803	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	374.14354	0	0	0	91.120795	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	33.788381	0
7	0	0	0	0	0	0	0	0	284.16065	0	0	0	0	0	0	0	0	0	0	0	290.31982	0	0	0	0	0	257.19378	0	0	0	0
8	8.6678204	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	37.300431	0	0	0	0	0	0	0	504.6623	0	0	0	0
9	0	0	0	0	0	0	714.44866	0	0	0	0	0	0	0	0	0	63.197904	0	0	0	0	0	0	77.524422	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	61.917788	0	0	0	0	0	0	0	0	0	37.13241	0	0	0	0	0	0	0	173.90929
11	0	3305.3613	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7161.2733	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	406.87414	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	982.80336	0	0	0	0	4.1622472	0	0	0	0	0	0	0	0	0	0	0	0	0	0	193.66764	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	225.60208	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	118.16079	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	7105.525	0	0	0	0	0	0	0	0	0	6230.8035	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6270.0975	0	0	0	0
17	0	0	0	0	0	0	0	0	6785.5877	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	138.84971	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	531.06922	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10.255965	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	191.47621
21	0	0	0	0	0	0	357.69217	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	263.45854	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	77.531685	0	0	0	0	0	0	0	600.32835	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	770.88764	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	746.11847	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	389.42123	0	0	0	0	0	25.441038	0	661.82525	0	0	0	0	0	0	0	0	204.35688	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	752.33474	81.009469	0	0	0	0	0	0	0	158.94991	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	662.91965	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	109.29552	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	730.69095	0	0	0	0	0	0	0	0	0	0
30	0	0	0	44.711316	38.402913	0.8507048	0	0	0	0	0	0	603.18629	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	5205.2236	0	0	0	0	0	0	0	0	0	6374.023	0	0	0	0	0	0	0	0	0	0	0