

生产企业原材料的订购和转运方案

摘要

本文针对生产企业原材料的订购和转运方案的特性，考虑供求关系、配比关系、经济性以及企业自身协议完成度等因素，通过对四个问题的具体分析，适当地合理假设得到每个问题的解决方案。

依据近五年企业总需产能选择供应商（问题 1）对现有的原始数据进行预处理，通过 EXCEL 筛选除去供货极低的供应商，获得较为规整且有价值的数据。利用冒泡排序，将近五年 240 周每家供应商对企业每周的供货量按由大到小依次排序，各家供应商得到 240 周每周排名序列，再对其求平均值，最后根据所得平均值由小到大进行排序，确定前 50 家供应商作为最重要的供应商为企业尽可能多地提供原材料，见表 4。

依据企业总产能上限筛选最优订购转运方案（问题 2）对 402 家企业近 5 年的总产能进行排序，在总产能误差为 0.3% 的限定条件下筛选，最终选择 37 家供应商，见表 5。将附件 1 中 240 周 402 企业的订货量与供货量的差取绝对值，依次对上述连续两周所求的绝对值求均值后从小到大排序，选取位于前 12 位的连续两周，即选定 24 周，并得到订购方案见附件 A。通过附件 2 中数据分析各转运商的损耗率，并将损耗率为 0 的转运商去除，将转运数据规整化、标准化、有效化。将转运商的平均损耗率由小到大进行排序，计算出 24 周每一周转运商数量见表 7。根据订货量与供货量的匹配程度，分析方案的实施效果、转运商的转运率，见图 3、图 4。对订购方案中订货量与供货量的差值分析，发现 2.9% 的供应商每周为企业提供的供货量略微多余企业的实际订购量，只有 0.2% 的供应商每周的供货量达不到企业订购需求，订购方案较优，且每一个转运商的转运效率都在 70% 以上。

依据对不同原材料的倾向要求限筛选最优订购转运方案（问题 3）对生产 A、B、C 三种原料的供应商进行分组，在每一组内对供货商按照每周平均供货量从 A 到 C、近五年每周的产能由大到小排序。根据多 A 少 C 的原则，以企业最大产能为标准，筛选出 32 家供应商企业见表 9。利用问题二中同样的方法制定出新的订购方案及转运方案见附件 A 和 B。方案的实施效果、转运商的转运率，见图 6、图 7。

依据供应商的材料供应量选定企业扩大的最大产能方案（问题 4）通过计算 402 家企业年平均产能，选出最大的年份的平均产能作为提高产能计算的依据，计算出每周的产能最大值为 40525.5 立方米，提高了 12325.5 立方米，增长率为 43.6%。利用问题二中方法确定新的订购方案和转运方案，见附件 A、附件 B。

关键词：订购方案，转运方案，实施方案，随机算法

一、 前言

1.1 研究背景

建筑装饰板材生产商在产品生产的过程中，朱崇贤^[1]认为供应链的整个体系一定程度上可以控制产品质量，还能在增加企业的生产效率的同时降低企业的生产成本，因此需要保证供应链体系的完整性、系统性、高效性。叶红霞^[2]提出库存管理可以直接影响存货周转率，因此高效、合理的库存管理能加快存货周转率，周转率提高后也能相应减少库存占用资金从而提高企业的运营能力。张雨晨^[3]在材料运输方面进行研究，通常材料运输的相关费用是项目总成本的重要组成部分，因此原材料的运输费用影响着整个项目的总成本费用支出。

陶连城^[4]认为在方案的选择中无论是资源分配、状态变更、路径选择等各个方面的因素对任务动态规划都会涉及过多或是过少的影响。在规划的过程中，本身就存在数量和范围的问题，还要对动态变化的事件目标和资源情况做进一步的满足。

木质纤维和其他植物素纤维材料供应商作为生产商的上流合作企业，在与供货商每周原材料订购方面，尽可能满足企业的最大生产需求可以降低企业维持经营的消耗。再交由转运商将每周的货物转运到仓库进行归库储备。无论货物储备还是货物运输方面，确定最优方案均可以较大程度上推进企业生产制备，降低成本，提高企业显隐性资产。

1.2 问题重述

基于上述研究背景，本文需要研究完成以下任务：

某建筑和装饰板材的生产企业可以使用 A、B、C 三种木质纤维和其他植物素纤维材料原材料企业中每立方米需要消耗 0.6 立方米 A 类原材料，或者 0.66 立方米 B 类原材料，或者 0.72 立方米 C 类原材料。该企业的每周产能最大值为 2.82 万立方米，每年按 48 周安排生产，需要提前制定 24 周的原材料订购和转运计划，企业需要根据自身的产能极限，合理地向原材料供应商订购原材料。每周所需订购地原材料地数量称为“订货量”，每周供应商提供地原材料称为“供货量”。在订购完成之后，企业需要联系第三方物流公司对原材料转运到企业仓库。其中，本公司所需材料具有特殊性，供应商的供货量会与企业的订购量由一定的差别。通常企业会尽可能保持不少于未来两周生产的原材料库存量，一般来说企业将收购供应商提供的全部原材料。

在运输的过程中，原材料会有一定程度的损耗，损耗率为损耗量占供货量的百分比，每家转运商的运输能力为 6000 立方米/周。规定在通常情况下，每家供应商应尽量选取一家转运商进行原材料的运输。

原材料的采购成本很大程度影响企业的生产效益，实际中 A 类原材料的采购单价比 C 类原材料高 20%，B 类原材料的采购单价比 C 类原材料高 10%。三类原材料运输和储存的单位费用相同。要求结合相关附件进行分析。

- (1) 对供货商供货特征进行量化分析，并确定 50 家重要供货商。
- (2) 参考问题一，确定供应商数量最大程度上满足企业生产需求。为企业制定未来 24 周每周最经济的原料订购方案，制定损耗最少的转运方案，并对

- 其效果进行分析。
- (3) 压缩企业的转运以及仓储成本，计划尽量多的采购 A 类，尽可能少的采购 C 类原材料。同时降低转运商的转运损耗量，制定有关订购和转运的方案，分析其实现效果。
 - (4) 根据现有原材料以及供应商的供应特点和转运商的实际情况，确定该企业的周产能可以提高多少，并制定未来 24 周的订购方案和转运方案。

二、 问题分析

2.1 概论

这是一个选择与优化分析问题，根据原料供应商的每周供货能力，选择更加符合企业产能要求的供应商；根据转运商转运途中转运的能力，选择损耗率更少的转运商；根据不同材料的采购单价差异，选择更适合企业订购的配比。难点在于限定条件的选择，综合考虑以上的条件制定不同的选择方案、订购方案、转运方案，以及在选定这些条件之后进行模型优化，然后对每一种选择的效果进行分析。

2.2 问题一的分析

为了分析与企业每周产能最匹配的 50 家供应商的最优组合，建立模型，初步研究供应商每周的供应量。运用动态规划，对 402 家供应商进行排序，得出较为靠前和较为靠后的供应商。最后再利用企业所需要的总产能、每立方米分别所需的三种材料数量构建配比分析模型，定量得出供应商所生产的材料配比后，计算出前 50 家供应商企业。

再用排序和筛选的方法、穷举法对已经得出的结果进行验证，观察对比验证出的数据结果是否吻合。

2.3 问题二的分析

为了在第一问的基础上得出更精简的供应商名单，本节先进行数据的预处理，对整体的方向进行选择。以剩余企业最大的满足生产效能为标准，将其转化为需要剔除的数据条件最小的影响供应商，所能提供的产能为目标函数。主要建立模型，运用不同条件下进行取值，再将所设置的限定条件不断缩小区域，将不同限定条件下的结果进行比较，进一步的约束条件设置在为 0.3%，最终筛选出最少的剩余供应商数。针对已经筛选出的供应商中供货量与订货量进行比较，根据比值较小的供应商提供出 24 周的订购方案，再根据已经选定的供应商进行转运商的选择。

为了选择合理的订购方案，需要对附件 1 中的数据进行运算各个供应商各自每周的供应量与订购量的匹配度，通过计算得到每家供应商连续 2 周相应的 $\overline{\Delta d_{nm}}$ ，为了在合理的订购方案中制定最经济的转运方案，依照每一家转运商每

周周转运途中的平均损耗率，计算未来的转运商家数。

最后对设计的订购和转运方案实施效果分析，对现有结果进行图表分析并预测模拟未来 24 周的方案。

2.4 问题三的分析

为了压缩企业的生产成本，需要重新制定原材料订购和转运方案。在关于订购方案的选择方面，其中问题二与问题三的分析有不同和相似的部分。问题二更多的是依照最大产能为作为主要的限定条件，而问题三新增的限定条件为尽可能多的选择生产 A 的供应商，最大可能不选择生产 C 材料的供应商，企业的最大产能上限也是重要的参考条件之一。因此我们建立了动态规划模型，运用了拉斯维加斯算法，二者的相似之处在于对于订购方案以及转运方案的的选择问题中，与前文中所用的方案相似，都是剔除掉停止供应周数下的转运商，按每位转运商的平均损耗率为依据进行选择。再根据各家转运商的排序结果，按照顺序选择相应家数的转运商对原材料制定损耗率最少的转运方案。

与问题二中的方法类似，最后依照配比进行筛选的商家进行订购量的效果分析，对转运商的转运方案效果分析。

2.5 问题四的分析

为了解决企业产能提高量的问题，需要对现有供应商的每周供应量综合分析。该问题没有设置如问题二、问题三中的限定条件，而只是针对 A、B、C 分类的不同供应商在五年内的供应量平均值的最大值进行筛选。将 A、B、C 分类求和后进行以年为单位求平均，分析出每年每类供货商的产能均值，计算每年所有提供原材料供应商的产能总量，最终得到 5 年每年对应的供应量并进行排序挑出其最大值。

三、 模型假设与符号表示

3.1 模型的相关假设

1. 转运商运输原材料过程中不考虑自然环境因素的影响

在转运商转运原材料的途中可能会有不可抗力的因素，以及意料之外的人为因素损耗，这一部分的转运损耗不能够预测是否发生，也不能够预测其发生的概率以及对企业造成的损失，因此不做考虑。

2. 不考虑市场对原材料的偏好行为

可能由于商家口碑、大众传播、媒体传播等因素对大众以及企业对供应商的形象产生影响，从而改变用户偏好，对不同的生产商企业形成具有个人意志思想的偏好行为。

3. 不考虑供应商因为自身厂家出现的问题对原材料的定价产生影响

可能由于自身厂家的生产方面的问题，中断与其他的企业进行合作，从而对企业后续的选择产生影响，也会相应改变生产商对订货量于转运方案的设定。

3.2 模型的符号表示

符号	说明
W	该企业每周接收量所能获得产能
W_0	该企业每周至少应接收原材料量
X_i	A 类供应商，其中 $i \in (1,2,3,\dots,146)$
Y_j	B 类供应商，其中 $j \in (1,2,3,\dots,134)$
Z_k	C 类供应商。其中 $k \in (1,2,3,\dots,122)$
x_i	A 类供应商对应供货量
y_j	B 类供应商对应供货量
z_k	C 类供应商对应供货量
a	单位立方米消耗 A 类原材料量
b	单位立方米消耗 B 类原材料量
c	单位立方米消耗 C 类原材料量
m	周次，取值范围为 $[1,240]$ 且为整数
I	每周排名序列
N	供应商序列
v_i 、 v_j 、 v_k	供应商选取情况（只取 0 或 1）
Δd	每家供应商每周供应量与订购量的差值的绝对值
T_α	转运商，其中 $1 \leq \alpha \leq 8$ 且 $\alpha \in Z$
S_{cm}	每周每家转运商运输损耗率
A_p	近五年供应商数据分组后 A 类供应商
B_p	近五年供应商数据分组后 B 类供应商
C_p	近五年供应商数据分组后 C 类供应商

μ	每周转运效率比值
W_{0p}	近五年数据分组后各组数据对应的产能

四、 问题一的建模与求解

本节主要研究供应商提供原材料产能的大小比较的数学模型和求解算法，并根据题目要求对所有的供应商进行筛选，最终权衡后筛选出 50 个重要合作企业。

基于此，将本部分分为几个小节：

4.1 问题分析，阐述对供应商不同周次所能提供总产能，以及同一周次所有供应商供货量的数据分析。

4.2 数据预处理，剔除掉数据价值较低的厂家，即产能在同周次同类型下较低的供应商。

4.3 模型的建立与求解，此问为题目基础的关键问题，运用贪心算法定量求出最优结果。

4.1 问题分析

在与本企业合作的 402 家供应商中选择 50 家重要合作企业，每一家企业所提供的 A、B、C 材料的种类不同，每一种材料每一周的供应量不同。发现所给的附件中提供的供应商供货量的数据差别较为明显，因此需要先进行数据的预处理，在数据中剔除无价值供应商，并分别将 A、B、C 类分类后，将原材料供应量标准化。

在第一个周筛选出 50 家供应商，使其相比于其他的供应商组合模式，能达到该周数下产能最大的条件，并分别计算每一周产能最大下的 A、B、C 材料配比，即能分别生产 A、B、C 的厂家数。剩余的 239 周用此相同的方法完成后，对总共产生的厂家编号进行筛选，将所有出现的供应商计数处理，利用冒泡排序的方法，在五年时间内共计 240 周，将所有的供应商每一周的产能进行排序筛选。将根据针对供应商所能提供的供应量用同样的排序方式所得出的结果进行对比整合，选择 50 家每周所提供材料较多的供应商作为确定项目。

4.2 数据预处理

其一可以计算出每一周 402 家供应商所能提供产能的总和，考虑企业每一周的需求量最多为 2.82 万立方米，然而其中部分周均低于企业所能承受的最大产能，因此每一周选择 50 家所能提供产能总和最大的供应商，均能符合企业生产条件。其二将每一家供应商在五年中的供应量进行排序，剔除掉产能极低的 17 家总供应量为 28 立方米的供应商。

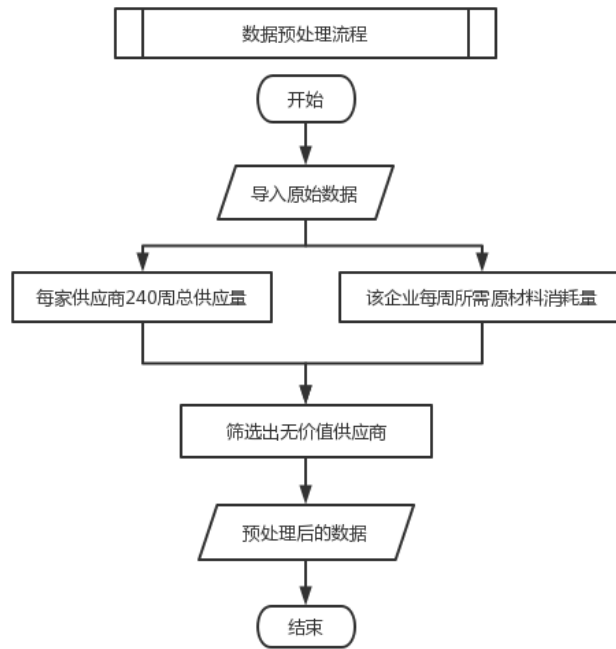


图 1：问题一数据处理流程

根据该企业一周所需消耗的原材料以及每家供应商 240 周内为该企业所提供的总供应量，将附件 1 中的数据预处理，除去少部分远不能够供应该企业产能的供应商。下面列举了部分无法达到供应条件的供应商：

表 1：无法达到条件的部分供应商

供应商	240 周总供应量（单位：立方米）
S014	28
S083	28
S144	28
S162	28
S223	28
S262	28
S382	28
S119	29
S323	29

4.3 模型的建立

通过 EXCEL 筛选，得到供应 A 类材料的供应商有 146 家、供应 B 类材料的供应商有 134 家、供应 C 类材料的供应商有 122 家，对各供应商分类依次编号，令供应 A 类材料的供应商为 x_i ；供应 B 类材料的供应商为 y_j ；供应 C 类材料的供

应商为 z_k 。

表 2：供应商材料分类

材料分类	供应商数量（个）
A	146
B	134
C	122

假设该企业每周可接收原材料总量的生产产能为 W 立方米，根据题目中所给出的单位立方米的消耗原材料信息，计算不同材料单位立方米可产能总和，得：

$$W = \frac{\sum x_i}{a} + \frac{\sum y_j}{b} + \frac{\sum z_k}{c}$$

其中， a 、 b 、 c 分别为每立方米所需要消耗的 A、B、C 类原材料。

为了满足该企业的产能需求，供应商应尽可能供应大量原材料(已经过初步分析可知将所有的供应商每周的供货量累加大多数低于企业的产能上限)，以此根据各家供应商每周对该企业供应的原材料量从大到小依次排序，并以 N_{I_m} 记录下每家供应商的相应序号，其中 m 代表周次，将每周所得的相应序号以 I 记录，计算得出所有供应商在近五年对该企业的供应量排名：

$$I' = \frac{\sum I}{\sum m}$$

利用所得到的排名均值，从小到大进行排序，即可确定排在最前的 50 家供应商为重要的合作对象。

4.4 模型的求解和分析

在问题一中，需在 402 家供应商中，根据供货特征确定 50 家最为重要的供应商，保证该企业的产能每周尽可能达到需求。该企业每周产能为 28200 立方米，所需要 A 类原材料 16920 立方米或 B 类原材料 18612 立方米或 C 类原材料 20304 立方米，如下表所示：

表 3：每种原材料所需的订购量

原材料	每周所需订购量（单位：立方米）
A	16920
B	18612
C	20304

利用上述模型，结合题目要求，通过 EXCEL 筛选，得到供应 A 类材料的供应商有 146 家、供应 B 类材料的供应商有 134 家、供应 C 类材料的供应商有 122 家，现需从 402 家供应商选出最重要的 50 家供应商。因此生产 A、B、C 的

供应商的总和为 50，所求出的产能总和应小于等于企业最大产能，则：

$$\begin{cases} i + j + k = 50 \\ \forall W \leq W_{\max} \end{cases}$$

其中 $i \in (1, 2, 3, \dots, 146)$, $j \in (1, 2, 3, \dots, 134)$, $k \in (1, 2, 3, \dots, 122)$ ，得出最重要的 50 家供应商。

通过数据预处理，得到结论：所有供应商所能供应企业的总和大多数周均无法达到企业产能所消耗的原材料，即会出现供不应求的状态。采用本节上述模型进行求解：

$$\sum m = 240$$

表 4：50 家重点供应商

供应商	供应商	供应商	供应商	供应商
S361	S306	S284	S266	S150
S229	S308	S040	S123	S314
S108	S330	S364	S007	S037
S340	S356	S367	S348	S074
S275	S194	S346	S139	S189
S329	S352	S294	S003	S086
S151	S143	S080	S374	S338
S282	S247	S244	S140	S126
S131	S031	S055	S114	S395
S268	S365	S218	S291	S023

五、 问题二的建模与求解

本节主要在问题一的基础上进行优化和分析，然后为已经得出的供应商制定最优的原材料订购方案和物流转运方案。本节分为三部分进行，第一部分为选定每周最小的供应商数，第二部分为依据选定的供应商制定订购方案，第三部分为基于第二步的订购方案设计合理经济的转运商转运方案。

基于此，本部分分为如下几个小节：

5.1 问题分析，阐述对优化模型的分析 and 建模思路。

5.2 数据的预处理，通过分析发现转运商部分周数不进行转运，在第三部分转运方案的安排中应避免不合理的数据。

5.3 模型的建立，分析得出筛选的约束条件，综合考虑供应商的供货能力，转运商保护原材料的能力，建立分析模型。

5.4 模型的求解，进行定量和定性的分析，并客观评价模型的优势与不足，观察是否能通过进一步的分析优化模型。

5.5 对实施效果的分析，将本题中得出的方案进行效果评价，也是验证其正

确性的方法。

5.1 问题分析

在问题一已经筛选出 402 家重要企业的基础上进一步的分析。计算这所有企业的五年内的供货总量排序，从其中由高到低进行排序，排名越高即供应商提供材料的能力越强。现已知如果需要满足企业产能，需要所有家供应商五年的供货量总共为 240×28200 立方米，设定相应固定的约束条件，则筛选过程中需要剔除的供应商应该尽可能对剩余供应商所能提供的总供货量的影响较小。则将从权重最小的开始进行去除，如果对所提供总产能变化在约束条件之内，则继续上述步骤，如果有较大变化则停止筛选剔除，剩余的即为所求的至少需要选择能够尽可能满足企业产能的供应商 37 家。

关于未来 24 周最经济的订购方案的选择，在上一步进行最少供应商的选择之后，需要对每一家供应商的信用以及供求关系进行判断。对每一家供应商每一周供货量与订货量的差的取绝对值，再对连续两周的绝对值取平均值进行从小到大的排序。其中排除特殊情况，需要剔除掉有所选出的两周之间有重合项的供应商。以每一家供应商为基准选出 12 个排名较为靠前的绝对值之差，即 24 周的选择，该供应商的 24 周最经济的订购方案。

关于制定损耗最少的转运方案，首先要对转运商的数据进行预处理，因为所有的转运商在运输的过程中都会存在一定的损耗率，但是通过附件 2 中给出的有关转运率的信息可知，某一些转运商在该周的转运损耗率为 0，因此可以推断该转运商在该周并没有转运计划，在制定转运方案的时候应事先避免安排这些转运商，在运用数据的时候去除这些“无效数据”。

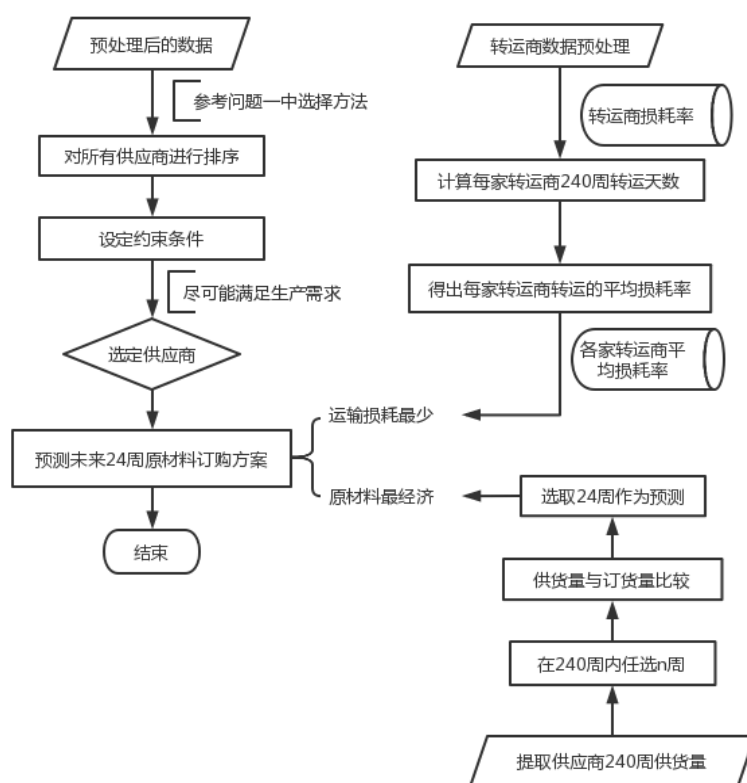


图 2：问题二分析流程

5.2 数据的预处理

通过附件 2 中的数据，发现部分转运商在部分周没有参与到装运原材料中，例如：转运商 T3、T4 在前八周运输损耗率数据为 0，表示这两家转运商在这 8 周没有参与该企业原材料转运方案，此类数据被视为“无效数据”，对统计每家转运商的运输损耗率没有实际意义。因此，需要排除此类数据，统计出有效的运输损耗率数据 $S_{\alpha\beta}$ ，再对每家转运商的运输损耗率进行运行。

5.3 模型的建立

5.3.1 针对供应商的选取模型

针对问题二，先在 402 家供应商中选取满足题目要求的供应商，再在近五年的数据进行对未来 24 周订购方案的模拟。

问题二可以描述为：有 n 家供应商和 1 个企业，对各类产品 A、B、C，其对应单价分别为 a 、 b 、 c ，每家单周供货量为 w_i （单位：立方米），企业一周所能接收的总供货量 W （单位：立方米）。先需选取供应商为企业提供原材料，使企业尽可能满足一周的需求。

对于各个供应商，只有两个选择，即选择其为该企业供应原材料或者不选择，不可多次选择，也不能选择一部分进行运输供应。

假设 v_i 、 v_j 、 v_k 分别表示供应 A、B、C 三类原材料的供应商 n 选取情况，当 $v_i=0$ 或 $v_j=0$ 或 $v_k=0$ 时，表示没有选择该供货商为企业提供原材料；当 $v_i=1$ 时，表示选择该供货商为企业提供原材料。

根据问题描述，设计出如下约束条件和目标函数。

$$\text{约束条件: } \begin{cases} W - \left(\frac{\sum x_i v_i}{a} + \frac{\sum y_j v_j}{b} + \frac{\sum z_k v_k}{c} \right) \leq \eta \\ v_i, v_j, v_k \in \{0,1\} \end{cases}$$

$$\text{目标函数: } \max(\sum w_i v_i + \sum w_j v_j + \sum w_k v_k)$$

其中， $1 \leq i \leq n$ ， η 为约束参数，问题二归结为寻找一个满足约束条件并且是目标函数达到最大的解向量 $V = (v_1, v_2, \dots, v_n)$ 。

5.3.1 针对订购方案和转运方案的选取模型

5.3.1.1 针对供应商供应方案的选取

先计算出每家供应商每周对该企业供应量与该企业每家供应商每周的订购量的差值的绝对值，记作 Δd_{nm} ，单位为立方米， n 为供应商编号， m 为周次，则

$$\Delta d_{nm} = |\text{供货量} - \text{订购量}|$$

取连续 2 周的差值的平均值，即 $\overline{\Delta d_{nm}} = \frac{1}{2}(\Delta d_{nm} + \Delta d_{n(m+1)})$ ，作为选取订购方案的参照标准，在已知订购和供货历史数据中预测未来该企业的订购方案。

5.3.1.2 针对转运商转运方案的选取

令 T_α 为转运商，每周每家转运商的运输损耗率为 $S_{\alpha m}$ ，其中 $1 \leq \alpha \leq 8$ ， $1 \leq m \leq 240$ 。计算每家转运商的平均损耗率：

$$\overline{S_\alpha} = \frac{\sum S_{\alpha m}}{\sum m}$$

根据所得的平均损耗率 $\overline{S_\alpha}$ ，将去作为参考标准，尽可能多的选择 $\overline{S_\alpha}$ 较小的转运商参与未来 24 周每周的转运方案中。

5.4 模型的求解和分析

5.4.1 关于供应商的选择

参考问题一中选取的 402 家最重要的供应商，利用问题一中的排序（按照每周各供应供货量排序，取平均名次），计算出每家供应商近五年为该企业所供应的原材料总和。若该企业每周均有 A 类原材料产能，根据每立方米产品需消耗 A 类原材料 0.6 立方米，得：

$$w_0 = 28200 \times 0.6 = 16920 \quad (\text{单位：立方米})$$

设定约束参数 $\eta = 28200 \times 240 \times 0.3\%$ ，现为了使企业尽可能满足生产需求，

因此企业每周所需要达到的接收量至少为 w_0 。根据约定参数，在 402 家供应商中一一排除，计算出排除各家供应商后，剩余供应商所能提供的供货量与 W 作差，是否小于约束条件。

现规定如下：

（1）若所得差值小于约束条件，则被视为排除该家供应商后，对最后企业产能无影响，即该家供应商每周对企业的供应量不会影响企业产能；

（2）若差值大于约束条件，则被视为重要供应商，作为最后的选择，将其对企业的供应量计入最后的供应量中。

以此为条件约束，至少选取 37 家供应商才能满足该企业原材料的供应去需求，选取结果如下图所示：

表 5：选取的 37 家供应商		
供应商	供应商	供应商
S229	S330	S037
S361	S356	S374
S140	S268	S126
S108	S306	S284
S151	S194	S365
S340	S348	S031
S282	S352	S040
S275	S143	S338
S329	S201	S364
S139	S307	S367
S131	S395	S055
S308	S247	S346
		S080

5.4.2 关于订购方案的设计

针对上述方式所选取的 37 家供应商，调取各家供应商近五年的订购量和供货量，通过计算得到每家供应商连续 2 周相应的 $\overline{\Delta d_{mm}}$ ，为保证该企业所定制的未来 24 周订购方案原材料最经济，需对所得数值由小到大进行排序。经计算排序后得出相应原材料订购方案，结果在附件 A 中给出。

5.4.3 关于转运方案的设计

运用预处理完后的转运商运输损耗率数据，参考问题一中对各供应商的排序方式，对附件 2 中所给定的各家转运商进行排序，结果如下表所示：

表 6：8 家转运商的平均损耗率商

次序	转运商 ID	平均损耗率
1	T3	0.186
2	T6	0.544
3	T2	0.921
4	T8	1.010
5	T4	1.570
6	T1	1.905
7	T7	2.079
8	T5	2.890

根据上述制定的订购方案，结合题目中已知条件：每家转运商每周可转运 6000 立方米原材料。计算出未来 24 周每周所需要的转运商家数，相应结果在下表给出：

表 7：未来 24 周所需的转运商家数

周次	转运商家数	周次	转运商家数	周次	转运商家数
第 01 周	3	第 09 周	2	第 17 周	2
第 02 周	4	第 10 周	3	第 18 周	2
第 03 周	3	第 11 周	3	第 19 周	3
第 04 周	4	第 12 周	3	第 20 周	5
第 05 周	4	第 13 周	2	第 21 周	3
第 06 周	3	第 14 周	3	第 22 周	4
第 07 周	3	第 15 周	3	第 23 周	5
第 08 周	4	第 16 周	6	第 24 周	6

再根据各家转运商的排序结果，按照顺序选择相应家数的转运商对原材料制定损耗率最少的转运方案。转运方案如下表所示：

表 8：针对于不同转运商每周的安排

转运商 ID	周次	转运商 ID	周次	转运商 ID	周次
T3	第 09 周	T3	第 01 周	T3	第 02 周
T6	第 13 周	T6	第 03 周	T6	第 04 周
	第 17 周	T2	第 06 周	T2	第 05 周
	第 18 周		第 07 周	T8	第 08 周
			第 10 周		第 22 周
			第 11 周		
			第 12 周		
			第 14 周		
			第 15 周		
			第 19 周		
			第 21 周		

转运商 ID	周次	转运商 ID	周次
T3	第 20 周	T3	第 16 周
T6	第 23 周	T6	第 24 周
T2		T2	
T8		T8	
T4		T4	
		T1	

5.5 对实施效果的评价分析

5.5.1 订购方案实施效果分析

运用上述模型中的 Δd_{nm} ，结合近五年内 37 家供应商供货量与企业向该供应商订购量的数据，计算 37 家供应商供货量与订购量的差值，将其各差值在折线图中表示。

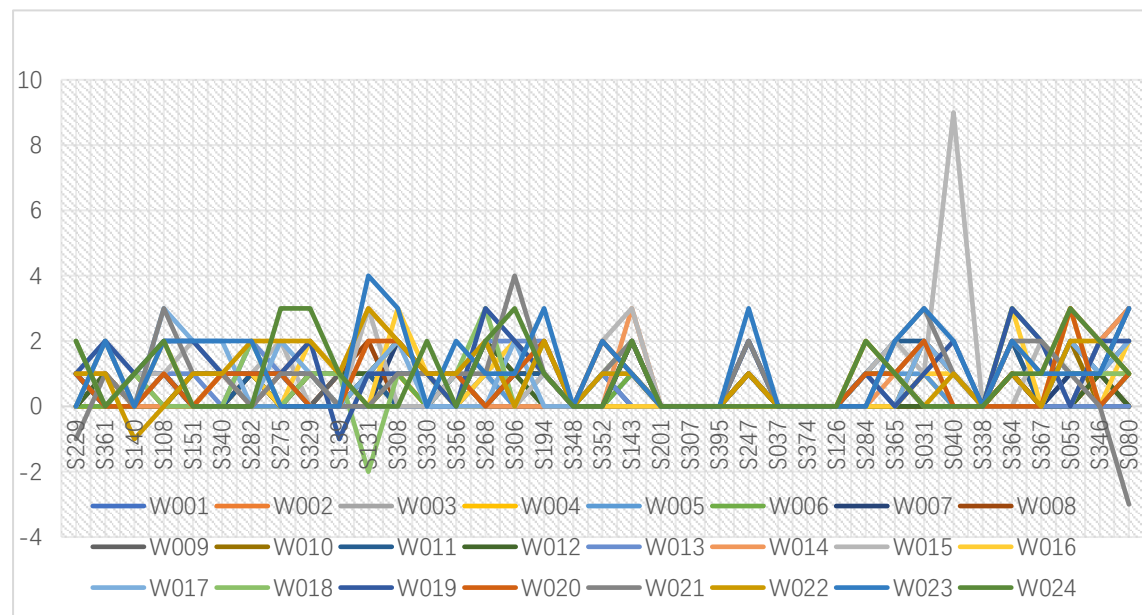


图 3：订购方案插值分析

通过图表，以近五年这 37 家供应商对该企业的供货量与订购量模拟未来 24 周订购方案的实施。有 2.9% 的供应商每周为企业提供的供货量略微多余企业的实际订购量，有 0.2% 的供应商每周的供货量达不到企业订购需求。因为考虑到在原材料运输的过程中，会有部分原材料因为自身材料的特殊性或是外力原因有所损耗，结合附件 2 中各家转运商的平均损耗率以及得出的设计转运方案，可以分析出依据上述方法所得出的预测订购方案合理。

5.5.2 转运方案实施效果分析

结合题目，每家转运商每周的转运能力为 6000 立方米，根据每周不同的订购方案确定每周合适的转运商进行每周的原材料转运方案。

计算未来 24 周每周所有供货商的供货量 $\sum w_i$ ，根据得出的转运方案计算出每周所有转运商可转运原材料的最大数值 W' ，则：

$$W' = \text{每周转运商个数} \times 6000 \text{ 立方米 / 周}$$

利用上述两个数值，作比较，得出每周转运效率比值，记作 μ ，即

$$\mu = \frac{\sum w_i}{W'}$$

根据所得的 μ 值，作出下图：

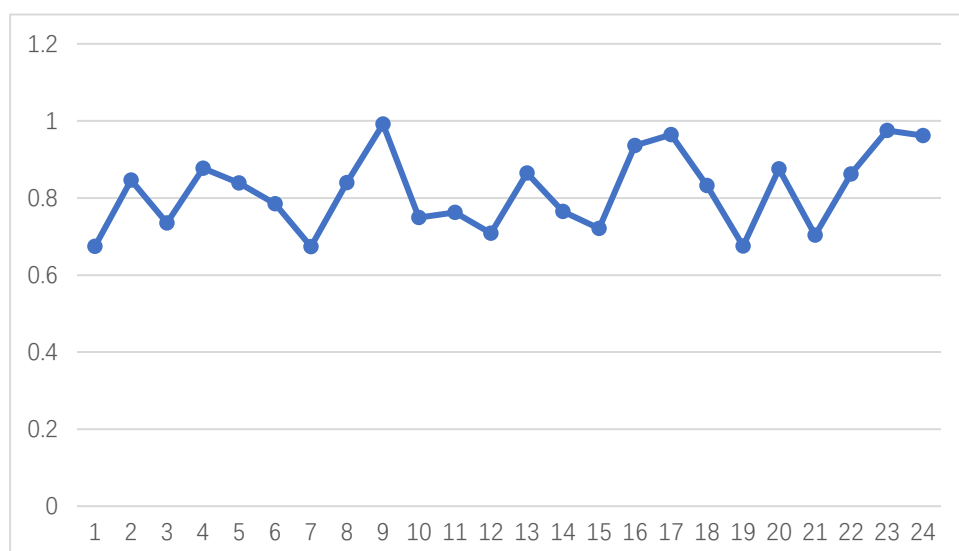


图 4：每周转运效率比值

结论：根据图表可以直观的得出设计的转运方案中，所有周的转运效率都维持在 60% 以上，其中例如第 9 周、第 17 周、第 23 周和第 24 周的转运效率接近 100%，由此说明转运效率可以达到该公司需求的基本要求，上述设计的转运方案合理有效。

六、 问题三的建模与求解

本节的主要问题是针对 A、B、C 订购的配比问题，进行新的模型建立与分析，从而尽可能减少企业仓储和转运成本。

基于此，本部分将分为以下几个小节：

6.1 问题分析，阐述根据 A、B、C 不同配比的选择制定新的订购和转运策略。

- 6.2 模型的建立，在第二问的基础上增加约束条件后进行相同策略下的建模。
- 6.3 模型的求解和分析，参考第二问和第三问建立模型的区别和联系，合理发散思维进行客观评价与改进。
- 6.4 对实施效果的评价分析，这一部分与上文中所用方法与步骤相似。

6.1 问题分析

针对问题三，为了压缩企业的生产成本，尽可能多的选择 A 材料作为生产产品的原材料，即选择供应 A 类产品的供应商。利用问题一中筛选出的重要合作供应商，将生产 A、B、C 的厂家进行聚类分析并按照 A、B、C 三个组类的顺序排列，在计算每个厂家 240 周的总供应量，计算各自供应商每周供应量的均值，并在分别三个组类中按供应量均值从大到小排序。在数据处理完成之后，进行供应商的选择，筛选的指标主要分为两个。其一是秉持着多 A 少 C 的原则挑选；其二是最后选定的总量，即每次从 A 到 B 到 C、从供应大到供应量小的进行筛选的时候应该尽可能接近该企业每周的极限产能，直至筛选至某一家时刚刚超过企业每周总产能即将其退回。

依据数学模型公式为其设置新的订购方式和转运方式所用的方法基本上与第二问一致。

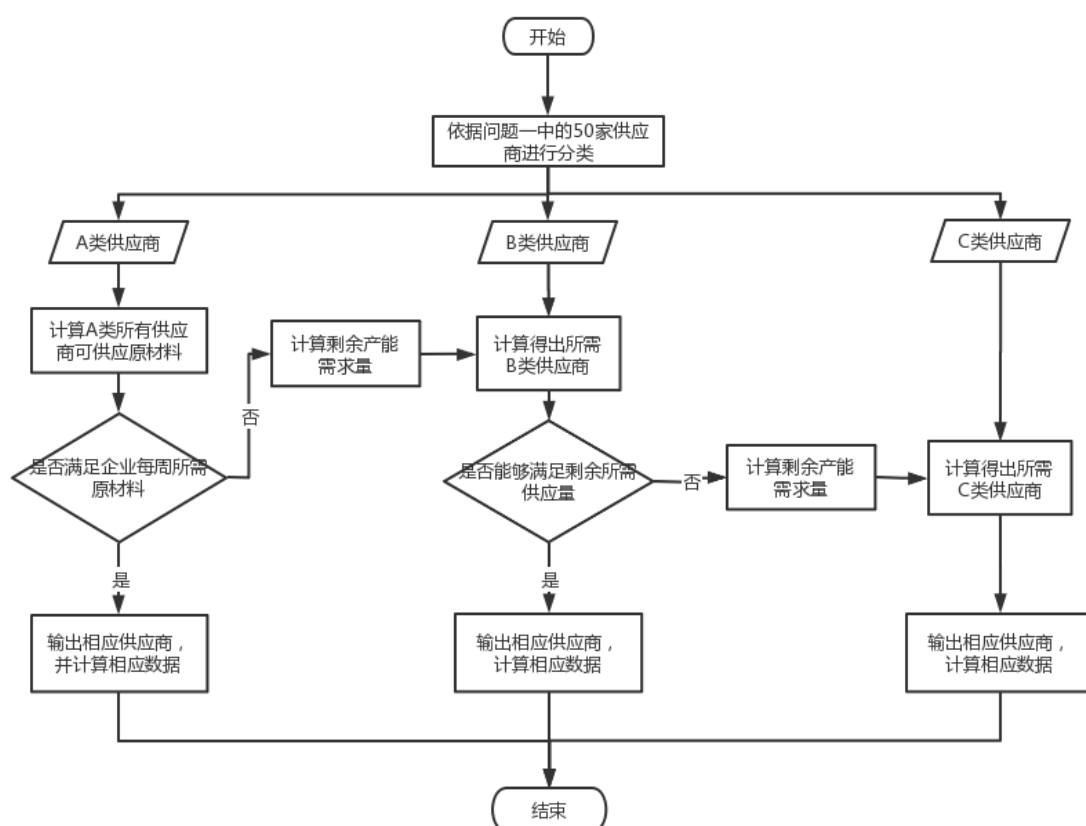


图 5：问题三分析流程

6.2 模型的建立

在问题二的基础之上，为了减少企业生产成本、转运成本、储存成本，尽量多地采购 A 类、尽量少地采购 C 类原材料。根据问题一中的参数：A 类材料的供应商每周为该企业供应的原材料为 x_i （单位：立方米）；供应 B 类材料的供应商每周为该企业供应的原材料为 y_j （单位：立方米）；供应 C 类材料的供应商每周为该企业供应的原材料为 z_k （单位：立方米）。令该企业每周产能为 w_ε ，结合企业每周需要达到的产能综合，得：

$$W' = \frac{\sum x_i}{a} + \frac{\sum y_j}{b} + \frac{\sum z_k}{c}$$
$$\Delta w = \frac{W'}{\text{周数}} - w_\varepsilon$$

由于三类原材料运输和储存的单位费用相同，为了达到减少企业生产成本、转运成本、储存成本的效果，要尽可能求得 i_{\max} 、 k_{\min} 、 Δw_{\min} 且 $\Delta w \geq 0$ 。

6.3 模型的求解和分析

6.3.1 关于供应商的选择

根据题目中的已知条件相应带入数据计算：

Step1: 将 402 家供应商按照供应的原材料类别进行分类，结合问题一的模型，计算出最为重要的 50 家供应商，将这些供应商分为 A、B、C 三类；

Step2: 计算出每家供应商 240 周平均每周对该企业的供应量，并按照分类将其平均每周供应量从大到小排序；

Step3: 先对 A 类供应商平均每周供应量进行叠加，与企业每周所需供应量作对比，计算得出仅 A 类供应商为企业供应原材料无法达到所需，并在此前提下保证了选择更多的 A 类原材料供应商；

Step4: 计算出剩余产能需求，由供应 B 类原材料的供应商供应剩余部分原材料；

Step5: 通过计算求解，选取 B 类供应商供应部分剩余原材料，并判断是否能够满足供应需求，通过计算得出无法满足；

Step6: 计算出 A 类和 B 类供应后剩余产能需求，最后剩余极少数材料供应需求由供应 c 类原材料的供应商供应；

Step7: 通过计算求解，选取 c 类供应商供应剩余原材料，计算出需要的 C 类供应商的家数及相应数据。

以此为供应方案依据，选择相应供应商为企业供应原材料，满足 i_{\max} 、 k_{\min} 、 Δw_{\min} 且 $\Delta w \geq 0$ 的条件。

表 9：选择的 32 家供应商

供应商	供应原材料类别	供应商	供应原材料类别
S229	A	S143	A
S201	A	S352	A
S282	A	S157	A
S348	A	S015	A
S275	A	S096	A
S329	A	S097	A
S395	A	S208	A
S307	A	S140	B

供应商	供应原材料类别	供应商	供应原材料类别
S108	B	S040	B
S308	B	S055	B
S139	B	S364	B
S330	B	S367	B
S340	B	S346	B
S131	B	S361	C
S031	B	S151	C
S338	B	S356	C

6.3.2 关于订购方案的设计

在供应商的选择环节已经筛选处理 32 家供应商，与问题二处理的方法类似。

引入 $\overline{\Delta d_{nm}}$ 表示供应商的供货效果用以为第一个筛选条件。

为保证该企业所定制的未来 24 周订购方案原材料最经济，引入平均每周供货量作为第二个筛选条件。综合已经判断出的两个筛选条件，进行其一运用计算排序后得出相应原材料订购方案，结果在附件 A 中给出。

6.3.3 关于转运方案的设计

在制定完成相应的订购方案的设计后，结合题目中的每周做大能进行转运材料的限定空间大小，计算出前 24 周每周所需要的转运商家数，相应结果在下表给出：

表 10：前 24 周每周所需转运商数

周次	转运商家数	周次	转运商家数	周次	转运商家数
第 01 周	2	第 09 周	2	第 17 周	2
第 02 周	4	第 10 周	3	第 18 周	2
第 03 周	3	第 11 周	3	第 19 周	2
第 04 周	4	第 12 周	3	第 20 周	5
第 05 周	4	第 13 周	2	第 21 周	3
第 06 周	3	第 14 周	3	第 22 周	4
第 07 周	2	第 15 周	3	第 23 周	5
第 08 周	4	第 16 周	6	第 24 周	6

转运方案如下表所示：

表 11：针对于不同转运商每周的安排

转运商 ID	周次	转运商 ID	周次	转运商 ID	周次
T3	第 01 周	T3	第 03 周	T3	第 02 周
T6	第 07 周	T6	第 06 周	T6	第 04 周
	第 09 周	T2	第 10 周	T2	第 05 周
	第 13 周		第 11 周	T8	第 08 周
	第 17 周		第 12 周		第 22 周
	第 18 周		第 14 周		
	第 19 周		第 15 周		
			第 21 周		
T3	第 20 周			T3	第 16 周
T6	第 23 周			T6	第 24 周
T2				T2	
T8				T8	
T4				T4	
				T1	

6.4 对实施效果的评价分析

6.4.1 订购方案实施效果分析

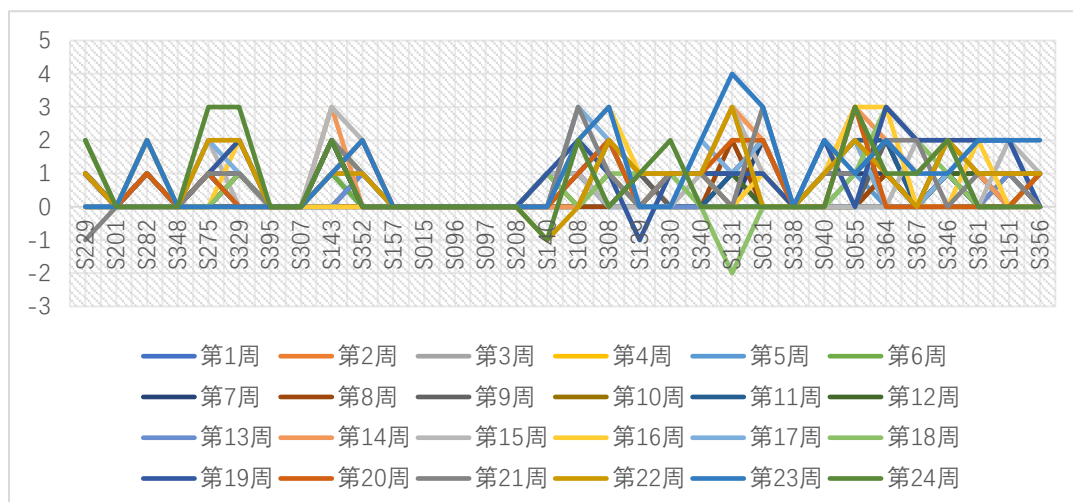


图 6：订购方案插值分析

6.4.2 转运方案实施效果分析

结合题目，每家转运商每周的转运能力为 6000 立方米，根据每周不同的订购方案确定每周合适的转运商进行每周的原材料转运方案。

计算未来 24 周每周所有供货商的供货量 $\sum w_i$ ，根据得出的转运方案计算出每周所有转运商可转运原材料的最大数值 W' ，则：

$$W' = \text{每周转运商个数} \times 6000 \text{ 立方米 / 周}$$

利用上述两个数值，作比较，得出每周转运效率比值，记作 μ ，即

$$\mu = \frac{\sum w_i}{W'}$$

根据所得的 μ 值，作出下图：

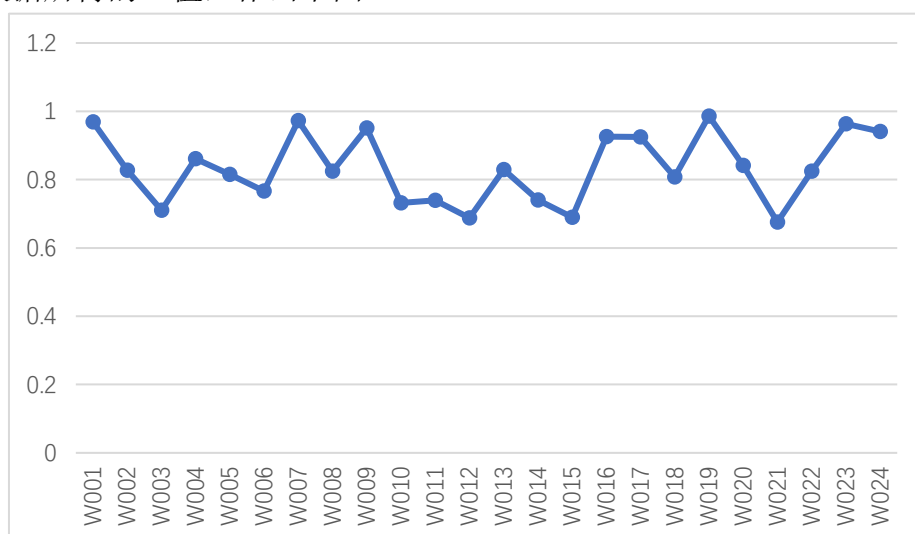


图 7：每周转运效率比值

结论：根据图表可以直观的得出设计的转运方案中，所有周的转运效率都维持在 60%以上，其中例如第 2 周、第 8 周、第 10 周、第 19 周和第 23 周的转运效率接近 100%，由此说明转运效率可以达到该公司需求的基本要求，上述设计的转运方案合理有效。

七、 问题四的建模与求解

本节在新的限定条件下进行新的求解，主要分为四部分，第一部分是选择企业需要依新旧照供应商提高的产能值，第二部分是对比企业的新旧最大产能，计算出产能提高率。第三部分是根据新的产能值设定合理订购方案，第四部分是依照企业的订购方案设置转运方案。

基于此，本部分将分为以下几个小节：

7.1 问题分析，基于前两问的条件步骤，分析出问题四的变化点和需要解决的问题。

7.2 数据预处理，将数据进行分类分组，以便之后步骤的使用。

7.3 模型的建立，以平均值、最大值为判定标准建立模型。

7.4 模型的求解和分析，最终计算出的企业每周的产能能极限提高至某一个值，后续的方案涉及问题与前文的方法相同，没有实质的变化。

7.1 问题分析

为了计算企业每周产能可以提高多少，则需要通过现有的供应商供货量的数据进行求解。经过数据分析可知供应商提供的材料在某些周数的时候可以超过原有的企业最大产能，而有一些周数的时候可以符合现有企业最大产能。企业每周产能的上限提高则可以接受更多供应商所供应的原材料。将 A、B、C 分类求和后进行以年为单位求平均，分析出每年每类供货商的产能均值，计算每年所有提供原材料供应商的产能总量，最终得到 5 年每年对应的供应量并进行排序挑出其最大值。

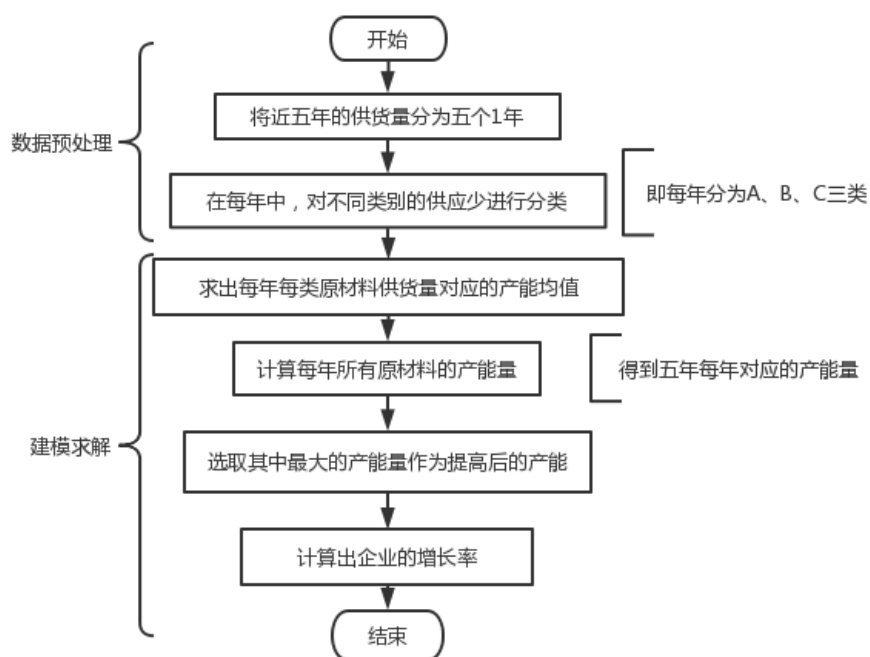


图 8：问题四分析流程

7.2 数据预处理

将近五年供应商的供货量数分为五个部分进行处理，在每一年中针对不同的供应材料种类分类，即每年分为 A、B、C 三类。

7.3 模型的建立和求解

7.3.1 关于最大产能的计算

将近五年的数据按照年份分为 5 组，即 240 周数据每 48 周为一组，并对每组数据按照供货商的供货类别进行分类，分别令为 A_p 、 B_p 、 C_p ，其中 $1 \leq p \leq 5$ 且为整数。根据不同类别，分别计算出每年每类原材料所对应的供货量，再结合单位消耗量，得出对应的产能并求均值，记作 W_{0p} ，单位为立方米，即：

$$W_{0p} = \frac{\sum A_p}{a} + \frac{\sum B_p}{b} + \frac{\sum C_p}{c}$$

观察所得的 W_{0p} 值，挑选出产能最大的值作为标淮，计算该企业产能的提高效率：

$$\varphi = \frac{W_0 - \left(\sum A/a + \sum B/b + \sum C/c \right)}{W_0} \times 100\%$$

代入附件 1 中所提供的数据信息，计算得出：（单位：立方米）

表 12：每种材料每年的平均供应量

	第一组	第二组	第三组	第四组	第五组
$\overline{\sum A_p}$	7192.7	10123.1	9397.8	8664.0	8836.3
$\overline{\sum B_p}$	6671.6	8029.0	7962.7	8631.0	8181.9
$\overline{\sum C_p}$	7062.9	8271.7	8121.3	6607.0	7437.6
W_{0p}	31905.8	40525.5	39007.5	36693.6	37454.0

7.3.2 关于订购方案和转运方案的设计

在问题二、问题三中引入的数据说明： $\overline{\Delta d_{nm}}$ 越小，证明该家供应商对该企业的供货量越符合该企业在该供应商的订购量，供应商的供货效果越好，筛选出供应商。订购方案和转运方案因数据过多，详见附件 A、B。

八、 模型的评价

8.1 模型的评价

8.1.1 问题一

（1）模型优缺点：

首先保证了对数据筛选以减少数据的输入量，使其由浅入深集中性整理，从定性分析再过度到定量分析。最后设计新的筛选方案对结果进行验证更保证进一步确定，更保证其准确性。模型设置的限定条件可能过于单一，只是考虑到产能最大化这一个限定条件，可能涉及到的方面并不是很全面。

（2）模型的改进：

例如在供应量与订购量之间的相关关系之中可以更深层次的考虑，将其余关系也列入分析之中。

8.1.2 问题二

（1）模型优缺点：

本节中采用的模型较多，针对附件 1 和附件 2 中所有的数据建立，逐步筛选、优化、检验和完善使模型具有较高的准确性。在分析题中各个因素的影响时可能忽略了各个因素之间的相互联系，模型在参数的设置方面仍然具有主观的随意性。

(2) 模型的改进：

确定各个参数的最佳值，确定各个因素之间的相关性以及相关性的强弱关系，最理想的状态数给出之间的权重关系。

8.1.3 问题三

(1) 模型优缺点：

进行了模拟仿真，给出了相对精确的结果，对未来的结果进行预测分析。在最后的实施效果分析，且数据没有办法规定在某一区间内，最终确定的每周的转运效率可能仍就较为分散，检验的标准为 60%可能不够精确。

(2) 模型的改进：

进一步挖掘题目中给定的隐藏的约束条件能够进一步准确的分析，降低结果检验成效的模糊度。或许能够将最终对供应商转运方案的选取更加优化，对订购方案的设计进一步降低企业生产成本，对转运方案整体的转运率提高、整体进行搭配后的损耗率降低。

8.1.4 问题四

(1) 模型优缺点：

进行了直观的定量分析，只运用到了一个条件。

(2) 模型的改进：

在分析题目的过程中，另一种与本题中使用的方法相似，可以进行另一种思路的常识。用所有企业每一周的均值来设定为最大的产能，还是在每类供应商中选取最大的均值作为最大产能是一个可以尝试的思路。

参考文献

- [1]朱崇贤.企业物资供应链构建分析[J].中国商论,2021(16):114-116.
- [2]叶红霞.制造企业库存管理问题和对策研究[J].大众投资指南,2021(13):155-156.
- [3]张雨晨. 建筑工程项目群材料运输问题研究[D].沈阳工业大学,2020.
- [4]陶连城. 基于 JSHOP2 的任务动态规划系统研究与实现[D].北京邮电大学,2020.

九、 附录

9.1 附录一：源工程文件 0. py

```
# -*- coding: utf-8 -*-
#二维数组
import numpy
num_list = numpy.zeros((402,240))
for i in range(0,402):
    for j in range(0,240):
        num_list[i][j] = i+1
#excel 表格引入数据
import xlrd
file_location = "C:/Users/95870/Desktop/数学建模代码/a.xlsx"
data = xlrd.open_workbook(file_location)
sheet = data.sheet_by_index(1)
sheet2=data.sheet_by_index(0)
data = [[sheet.cell_value(r,c) for c in range(sheet.ncols)] for r in range(sheet.nrows)]
data1 = [[sheet.cell_value(r,c) for c in range(sheet.ncols)] for r in range(sheet.nrows)]
data_d=[[sheet2.cell_value(r,c) for c in range(sheet2.ncols)] for r in range(sheet2.nrows)]
#排序
#(num_list[0][0],num_list[401][0])=(num_list[401][0],num_list[0][0])
y = numpy.zeros((402,240))
for k in range (240):
    for i in range (0,402):
        for j in range(0,401-i):
            if data[j+1][k+2]<data[j+2][k+2]:
                (data[j+1][k+2],data[j+2][k+2])=(data[j+2][k+2],data[j+1][k+2])
                (num_list[j][k],num_list[j+1][k])=(num_list[j+1][k],num_list[j][k])
#排名相同的同一个排名
for i in range (0,240):
    a=1
    y[0][i]=1
    n=1
    for j in range(0,401):
        if data[j+1][i+2]==data[j+2][i+2]:
```

```

        y[j+1][i]=y[j][i]
        n=n+1
    else :
        y[j+1][i]=a+n
        a=a+n
        n=1
for i in range(0,402):
    print(y[i][0])
#除以了 240 了的
list = numpy.zeros((402,2))
for k in range(1,403):
    sum=0.0
    for i in range(0,402):
        for j in range (0,240):
            if num_list[i][j]==k:
                sum=sum+y[i][j]
    sum=sum/240
    list[k-1][0]=sum
    list[k-1][1]=k
# 排序给出那 50 个（平均下来每周的）
for i in range (0,402):
    for j in range(0,401-i):
        if list[j][0]>list[j+1][0]:
            (list[j][0],list[j+1][0])=(list[j+1][0],list[j][0])
            (list[j][1],list[j+1][1])=(list[j+1][1],list[j][1])
#print(list1)
#5%:240*28200*0.5%=333840(这个不是排名， 是实际值)
list1 = numpy.zeros((402,2))#240 周总共的
for i in range(0,402):
    sum=0
    for j in range(0,240):
        sum=sum+data1[i+1][j+2]
    list1[i][0]=sum
    list1[i][1]=i+1
for i in range (0,402):#排序
    for j in range(0,401-i):
        if list1[j][0]<list1[j+1][0]:
            (list1[j][0],list1[j+1][0])=(list1[j+1][0],list1[j][0])
            (list1[j][1],list1[j+1][1])=(list1[j+1][1],list1[j][1])
for i in range(0,402) :
    print(list1[i][0],list1[i][1])
#第四问
list3 = numpy.zeros((402,2))#累加， 第一个是累加的数值， 第二个是多少个进行了累加
sum=0

```

```

for i in range (0,402):
    sum=sum+list1[i][0]
    list3[i][0]=sum
    list3[i][1]=i+1
for i in range(0,402) :
    print(list3[i][0])
list3 = numpy.zeros((240,3)) #分类看歌类每周的总值
for j in range(0,240):
    sum=0
    for i in range(0,402):
        if data1[i+1][1]=='A':
            sum=sum+data_d[i+1][j+2]
    list3[j][0]=sum
for j in range(0,240):
    sum=0
    for i in range(0,402):
        if data1[i+1][1]=='B':
            sum=sum+data_d[i+1][j+2]
    list3[j][1]=sum
for j in range(0,240):
    sum=0
    for i in range(0,402):
        if data1[i+1][1]=='C':
            sum=sum+data_d[i+1][j+2]
    list3[j][2]=sum
for i in range(1,6):
    sumA=sumB=sumC=0
    print(i)
    for j in range((i-1)*48,i*48) :
        sumA=sumA+list3[j][0]
        sumB=sumB+list3[j][1]
        sumC=sumC+list3[j][2]
    sumA=sumA/48
    sumB=sumB/48
    sumC=sumC/48
    print(sumA)
    print(sumB)
    print(sumC)
    print(sumA/0.6+sumB/0.66+sumC/0.72)
print()
for i in range (0,240):#排序
    for j in range(0,239-i):
        if list3[j][0]<list3[j+1][0]:
            (list3[j][0],list3[j+1][0])=(list3[j+1][0],list3[j][0])

```

```

for i in range(0,240):
    print(list3[i][0])
a=228 #240*0.95
sumA=0
for i in range (0,a):
    sumA =sumA+list3[i][0]
sumA=sumA/a
print(sumA)
sumB=0
for i in range (0,a):
    sumB =sumB+list3[i][1]
sumB=sumB/a
print(sumB)
sumC=0
for i in range (0,a):
    sumC =sumC+list3[i][2]
sumC=sumC/a
print(sumC)
print(sumA/0.6+sumB/0.66+sumC/0.72)
for i in range(0,240):
    print(list3[i][2])
#按 90%计算, 240*0.9=216
for i in range(0,50):
    print(list1[i][1])

for i in range(0,50):
    print(list2[i][0],list2[i][1])
    a=int(list2[i][1])
    print(a)
    print(data_d[a][0],data_d[a][1])

qushu=0
for i in range (0,402):
    if list1[i][0]<20304:
        qushu=i+1
        break

print(qushu)
for i in range(0,qushu):
    print(list1[i][1])

#第三问按

#订货量算总数

```

```

list2 = numpy.zeros((402,2))

for i in range(0,402):
    sum=0
    for j in range(0,240):
        sum=sum+data_d[i+1][j+2]
    list2[i][0]=sum
    list2[i][1]=i+1

#订货量排名
for i in range (0,402):#排序
    for j in range(0,401-i):
        if list2[j][0]<list2[j+1][0]:
            (list2[j][0],list2[j+1][0])=(list2[j+1][0],list2[j][0])
            (list2[j][1],list2[j+1][1])=(list2[j+1][1],list2[j][1])

num=0
sum=0
for i in range(0,50):
    a=int(list2[i][1])
    if data_d[a][1]=='A':
        sum=sum+list2[i][0]/0.6
        num=num+1
        print(list2[i][1])
        if (sum/240) >28200:
            print('$',num+1,list2[i][1])
            break

for i in range(0,50):
    a=int(list2[i][1])
    if data_d[a][1]=='B':
        sum=sum+list2[i][0]/0.66
        num=num+1
        print(list2[i][1])
        if (sum/240) >28200:
            print('$',num+1,list2[i][1])
            break

for i in range(0,50):
    a=int(list2[i][1])
    if data_d[a][1]=='C':
        sum=sum+list2[i][0]/0.72
        num=num+1
        print(list2[i][1])

```

```

        if (sum/240) >28200:
            print('$',num,list2[i][1])
            break

print('结束')

#除了 0.6、0.66、0.72 的
x = numpy.zeros((402,240))
for i in range(0,402):
    if data[i+1][1]=='A':
        for j in range(0,240):
            x[i][j] = data[i+1][j+2]/0.6
    elif data[i+1][1]=='B':
        for i in range(0,240):
            x[i][j] = data[i+1][j+2]/0.66
    else :
        for i in range(0,240):
            x[i][j] = data[i+1][j+2]/0.72

for i in range(0,50):
    print(list[i][1])

```

9.2 附录二：源工程文件 1. py

```

# -*- coding: utf-8 -*-
import numpy
import xlrd
file_location = "C:/Users/95870/Desktop/数学建模代码/b.xlsx"
data = xlrd.open_workbook(file_location)
sheet = data.sheet_by_index(0)
data = [[sheet.cell_value(r,c) for c in range(sheet.ncols)] for r in range(sheet.nrows)]

file = "C:/Users/95870/Desktop/数学建模代码/c.xlsx"
data1 = xlrd.open_workbook(file)
sheet1 = data1.sheet_by_index(0)
data_d = [[sheet1.cell_value(r,c) for c in range(sheet1.ncols)] for r in range(sheet1.nrows)]

sheet2 = data1.sheet_by_index(1)
data_g = [[sheet2.cell_value(r,c) for c in range(sheet2.ncols)] for r in range(sheet2.nrows)]

file_location = "C:/Users/95870/Desktop/数学建模代码/d.xlsx"
data_y = xlrd.open_workbook(file_location)
sheet = data_y.sheet_by_index(0)
data_y = [[sheet.cell_value(r,c) for c in range(sheet.ncols)] for r in range(sheet.nrows)]

```

```

file = "C:/Users/95870/Desktop/数学建模代码/e.xlsx"
data1 = xlrd.open_workbook(file)
sheet1 = data1.sheet_by_index(0)
data3_d = [[sheet1.cell_value(r,c) for c in range(sheet1.ncols)] for r in range(sheet1.nrows)]

```

```

sheet2 = data1.sheet_by_index(1)
data3_g = [[sheet2.cell_value(r,c) for c in range(sheet2.ncols)] for r in range(sheet2.nrows)]

```

```

file_location = "C:/Users/95870/Desktop/数学建模代码/f.xlsx"
data3_y = xlrd.open_workbook(file_location)
sheet = data3_y.sheet_by_index(0)
data3_y = [[sheet.cell_value(r,c) for c in range(sheet.ncols)] for r in range(sheet.nrows)]

```

```

file_location = "C:/Users/95870/Desktop/数学建模代码/a.xlsx"
data4 = xlrd.open_workbook(file_location)
sheet = data4.sheet_by_index(1)
sheet2=data4.sheet_by_index(0)
data4_g = [[sheet.cell_value(r,c) for c in range(sheet.ncols)] for r in range(sheet.nrows)]
data4_d=[[sheet2.cell_value(r,c) for c in range(sheet2.ncols)] for r in range(sheet2.nrows)]

```

```

file_location = "C:/Users/95870/Desktop/数学建模代码/g.xlsx"
data4_y = xlrd.open_workbook(file_location)
sheet = data4_y.sheet_by_index(0)
data4_y = [[sheet.cell_value(r,c) for c in range(sheet.ncols)] for r in range(sheet.nrows)]

```

```

#sum0 是零的个数
list = numpy.zeros((8,2))
for i in range(0,8):
    sum0=0
    sum=0
    for j in range (0,240):
        if data[i+1][j+1]==0 :
            sum0=sum0+1
        else :
            sum=sum+data[i+1][j+1]
    sum=sum/(240-sum0)
    list[i][0]=sum
    list[i][1]=i+1
for i in range (0,8):
    for j in range(0,7-i):
        if list[j][0]>list[j+1][0]:
            (list[j][0],list[j+1][0])=(list[j+1][0],list[j][0])

```



```

        (list[j][1],list[j+1][1])=(list[j+1][1],list[j][1])

...

for i in range(0,8):
    print(list[i][0],list[i][1])
...

#给预计的订货量
#第二问的 37 家      1
list1 = numpy.zeros((37,239,4))#第三和第四是每个人的差值（有正负那种）
a=b=c=0
for i in range(0,37):
    for j in range(0,239):
        a=data_g[i+1][j+2]-data_d[i+1][j+2]
        b=data_g[i+1][j+3]-data_d[i+1][j+3]
        list1[i][j][2]=a
        list1[i][j][3]=b
        if a>=0:
            a=a
        else:
            a=-a
        if b>=0:
            b=b
        else:
            b=-b
        c=(a+b)/2
        list1[i][j][0]=c #差值（除法）
        list1[i][j][1]=j+1#周数

for k in range(0,37):
    for i in range (0,239):
        for j in range(0,238-i):
            if list1[k][j][0]>list1[k][j+1][0]:
                (list1[k][j][0],list1[k][j+1][0])=(list1[k][j+1][0],list1[k][j][0])
                (list1[k][j][1],list1[k][j+1][1])=(list1[k][j+1][1],list1[k][j][1])
                (list1[k][j][2],list1[k][j+1][2])=(list1[k][j+1][2],list1[k][j][2])
                (list1[k][j][3],list1[k][j+1][3])=(list1[k][j+1][3],list1[k][j][3])

#第三问的 32 家      1
list_1 = numpy.zeros((32,239,4))#第三和第四是每个人的差值（有正负那种）
a=b=c=0
for i in range(0,32):
    for j in range(0,239):
        a=data3_g[i+1][j+2]-data3_d[i+1][j+2]
        b=data3_g[i+1][j+3]-data3_d[i+1][j+3]

```

```

list_1[i][j][2]=a
list_1[i][j][3]=b
if a>=0:
    a=a
else:
    a=-a
if b>=0:
    b=b
else:
    b=-b
c=(a+b)/2
list_1[i][j][0]=c #差值 (除法)
list_1[i][j][1]=j+1#周数

for k in range(0,32):
    for i in range (0,239):
        for j in range(0,238-i):
            if list_1[k][j][0]>list_1[k][j+1][0]:
                (list_1[k][j][0],list_1[k][j+1][0])=(list_1[k][j+1][0],list_1[k][j][0])
                (list_1[k][j][1],list_1[k][j+1][1])=(list_1[k][j+1][1],list_1[k][j][1])
                (list_1[k][j][2],list_1[k][j+1][2])=(list_1[k][j+1][2],list_1[k][j][2])
                (list_1[k][j][3],list_1[k][j+1][3])=(list_1[k][j+1][3],list_1[k][j][3])

#第四问的 402 家      1
list4_1 = numpy.zeros((402,239,4))#第三和第四是各个的差值 (有正负那种)
a=b=c=0
for i in range(0,402):
    for j in range(0,239):
        a=data4_g[i+1][j+2]-data4_d[i+1][j+2]
        b=data4_g[i+1][j+3]-data4_d[i+1][j+3]
        list4_1[i][j][2]=a
        list4_1[i][j][3]=b
        if a>=0:
            a=a
        else:
            a=-a
        if b>=0:
            b=b
        else:
            b=-b
        c=(a+b)/2
        list4_1[i][j][0]=c #差值 (除法)
        list4_1[i][j][1]=j+1#周数

```

```

for k in range(0,402):
    for i in range (0,239):
        for j in range(0,238-i):
            if list4_1[k][j][0]>list4_1[k][j+1][0]:
                (list4_1[k][j][0],list4_1[k][j+1][0])=(list4_1[k][j+1][0],list4_1[k][j][0])
                (list4_1[k][j][1],list4_1[k][j+1][1])=(list4_1[k][j+1][1],list4_1[k][j][1])
                (list4_1[k][j][2],list4_1[k][j+1][2])=(list4_1[k][j+1][2],list4_1[k][j][2])
                (list4_1[k][j][3],list4_1[k][j+1][3])=(list4_1[k][j+1][3],list4_1[k][j][3])
#print(list1)
'''
for i in range(0,37):
    print(i+1)
    for j in range(0,20):
        print(list1[i][j][0],list1[i][j][1],list1[i][j][2],list1[i][j][3])
    print()
'''

#24 周的信息（供应商家数，周数，从哪儿挑出来的）第二问的      2
list2 = numpy.zeros((37,12,3))
for i in range(0,37):
    for j in range(0,12):
        list2[i][j][0]=-2
for i in range(0,37):
    #k=12
    k=0
    for flag in range(0,50):
        #for k in range(0,12):
        b=0
        for j in range (0,12):
            #if list1[i][12-k][1]==list2[i][j][0]+1:
            if list1[i][flag][1]==list2[i][j][0]+1 or list1[i][flag][1]==list2[i][j][0]-1:
                b=1
                break
        if b==0:
            list2[i][k][0]=list1[i][flag][1]
            list2[i][k][1]=list1[i][flag][2]
            list2[i][k][2]=list1[i][flag][3]
            #list2[i][12-k][0]=list1[i][12-k][1]
            k=k+1
        if k==12:
            break

#第三问的      2
list_2 = numpy.zeros((32,12,3))

```

```

for i in range(0,32):
    for j in range(0,12):
        list_2[i][j][0]=-2
for i in range(0,32):
    #k=12
    k=0
    for flag in range(0,50):
        #for k in range(0,12):
            b=0
            for j in range (0,12):
                #if list1[i][12-k][1]==list2[i][j][0]+1:
                if list_1[i][flag][1]==list_2[i][j][0]+1 or list_1[i][flag][1]==list_2[i][j][0]-1:
                    b=1
                    break
            if b==0:
                list_2[i][k][0]=list_1[i][flag][1]
                list_2[i][k][1]=list_1[i][flag][2]
                list_2[i][k][2]=list_1[i][flag][3]
                #list2[i][12-k][0]=list1[i][12-k][1]
                k=k+1
            if k==12:
                break
    ...

for i in range(15,32):
    print(i+1)
    for j in range(0,12):
        print(list_2[i][j][0],list_2[i][j][1],list_2[i][j][2])
    print()
    ...

#第四问的      2
list4_2 = numpy.zeros((402,12,3))
for i in range(0,402):
    for j in range(0,12):
        list4_2[i][j][0]=-2
for i in range(0,402):
    #k=12
    k=0
    for flag in range(0,50):
        #for k in range(0,12):
            b=0
            for j in range (0,12):
                #if list1[i][12-k][1]==list2[i][j][0]+1:
                if list4_1[i][flag][1]==list4_2[i][j][0]+1 or list4_1[i][flag][1]==list4_2[i][j][0]-1:
                    b=1

```

```

        break
    if b==0:
        list4_2[i][k][0]=list4_1[i][flag][1]
        list4_2[i][k][1]=list4_1[i][flag][2]
        list4_2[i][k][2]=list4_1[i][flag][3]
        #print(list4_2[i][k][0])

        #list2[i][12-k][0]=list1[i][12-k][1]
        k=k+1
    if k==12:
        break

for i in range(0,402):
    for j in range(0,12):
        print(list4_2[i][j][1],list4_2[i][j][2],end=' ')
    print()
#第二问的          3
list3 = numpy.zeros((37,24))

for i in range(0,37):
    a=b=c=0
    for j in range (0,12):
        a=list2[i][j][0]
        b=a+1
        c=int(a+2)
        list3[i][j*2]=data_d[i+1][c]
        k=k+1
        c=int(b+2)
        list3[i][j*2+1]=data_d[i+1][c]
    ...
print('@')
for i in range (0,37):
    for j in range(0,24):
        print(list3[i][j])
    print()
print("@")
'''
#第三问的          3
list_3 = numpy.zeros((32,24))

for i in range(0,32):
    a=b=c=0
    for j in range (0,12):
        a=list_2[i][j][0]

```

```

        b=a+1
        c=int(a+2)
        list_3[i][j*2]=data3_d[i+1][c]
        k=k+1
        c=int(b+2)
        list_3[i][j*2+1]=data3_d[i+1][c]
    ...

for i in range (0,32):
    print(i+1)
    for j in range(0,24):
        print(list3[i][j])
    print()
...

```

#第四问的 3

list4_3 = numpy.zeros((402,24))

```

for i in range(0,402):
    a=b=c=0
    for j in range (0,12):
        a=list4_2[i][j][0]
        b=a+1
        c=int(a+2)
        #print(c)
        list4_3[i][j*2]=data4_d[i+1][c]
        k=k+1
        c=int(b+2)
        #print(c)
        list4_3[i][j*2+1]=data4_d[i+1][c]
    ...

print('@')
for i in range (32,50):
    print(i+1)
    for j in range(0,24):
        print(list4_3[i][j])
    print()
print("@")
...

```

```

...

for j in range(0,37):
    print(j+1)
    for i in range(0,24):
        print(list3[j][i])

```

```

        print()
    ...
#第二问的          4
a=0
list4= numpy.zeros(24)
for i in range(0,24):#列
    sum=0
    for j in range(0,37):#行
        a = data_y[j+1][i+2]
        sum=sum+a
    list4[i]=sum
#print(list4)

#第三问的          4
a=0
list_4= numpy.zeros(24)
for i in range(0,24):#列
    sum=0
    for j in range(0,32):#行
        a = data3_y[j+1][i+2]
        sum=sum+a
    list_4[i]=sum

#第四问的          4
a=0
list4_4= numpy.zeros(24)
for i in range(0,24):#列
    sum=0
    for j in range(0,402):#行
        a = data4_y[j+1][i+2]
        sum=sum+a
    list4_4[i]=sum

#第二问的          5

for i in range(0,24):
    b=list4[i]/6000
    if list4[i]%6000 != 0:
        b=b+1
        #print(b)
    #print(list4[i]/(6000*b))

#第三问的          5
for i in range(0,24):

```

```

        b=list_4[i]//6000
        if list_4[i]%6000 != 0:
            b=b+1
            #print(b)
        #print(list_4[i]/(6000*b))

#第四问的的          5
'''
for i in range(0,24):
    b=list4_4[i]//6000
    if list4_4[i]%6000 != 0:
        b=b+1
        print(b,end=" ")
    print(list_4[i]/(6000*b))
'''

'''
list5= numpy.zeros((402,192))
for i in range (0,24):
    sum1=sum2=0
    flag=0
    a=list[flag][1]
    for j in range(0,402):
        sum2=sum1+data4_y[j+1][i+2]
        b=int(8*i+a-1)
        if sum2>6000:
            list5[j][b]=6000-sum1
            #print('@',b,list5[j][b],j)
            flag=flag+1
            a=list[flag][1]
            b=int(8*i+a-1)
            list5[j][b]=sum2-6000
            #print('!',b,list5[j][b],j)
            sum1=sum2=0
        else:
            list5[j][b]=data4_y[j+1][i+2]
            #print(b,list5[j][b],j)
            sum1=sum1+data4_y[j+1][i+2]

for i in range(0,402):
    for j in range(150,192):
        print(list5[i][j],end=" ")
    print()
'''

```



```

list6= numpy.zeros((32,192))
for i in range (0,24):
    sum1=sum2=0
    flag=0
    a=list[flag][1]
    for j in range(0,32):
        sum2=sum1+data3_y[j+1][i+2]
        b=int(8*i+a-1)
        if sum2>6000:
            list6[j][b]=6000-sum1
            #print('@',b,list5[j][b],j)
            flag=flag+1
            a=list[flag][1]
            b=int(8*i+a-1)
            list6[j][b]=sum2-6000
            #print('!',b,list5[j][b],j)
            sum1=sum2=0
        else:
            list6[j][b]=data3_y[j+1][i+2]
            #print(b,list5[j][b],j)
            sum1=sum1+data3_y[j+1][i+2]
    ...
for i in range(0,32):
    for j in range(150,192):
        print(list6[i][j],end=" ")
    print()
...

```