

基于二进制整数规划的生产决策模型优化

摘要

本文讨论了工厂生产中对检测步骤的决策方案，基于重要因素优先，以生产利润为目标函数建立了二进制整数规划优化模型，为给定的具体生产流程寻找最优决策方案，同时分析考察了对应方案下的市场次品率，最终研究了更一般情况的情况，并提出了计算方法。

针对问题一，判断次品个数服从二项分布，采用**精确二项检验**方法进行了**单侧假设检验**。为尽可能节省工厂的检测花销，同时保证检测结果的稳定和准确，对二项概率分布进行了**正态近似**，利用区间估计方法推导出了在可接受误差 E 范围内的**最小样本公式**。对于问题给出的两种具体情况给出了详细的抽样检测方案，根据不同零件之间具有独立性，确定多次重复的**简单随机抽样**为抽样方法；对于情况一，属于右侧检验，设置**推荐的标准误差 E 值为 0.05**，计算得**最小抽样次数为 97 次**，能够允许出现的**样本次品率上限为 0.1563**，最多可以容忍的次品数量为 15 个；对于情况二，属于左侧检验，设置推荐的标准误差 E 值为 0.05，计算得**最小抽样次数为 59 次**，能够允许出现的**样本次品率上限为 0.0543**，最多可以容忍的次品数量为 3 个。

针对问题二，考虑**企业盈利**为最重要的**决策指标**，假设生产初始状况下两种零件数量相等，建立**二进制非线性整数规划模型**，将企业收益设置为**目标函数**，希望最大化；将每一步决策设计为**二进制变量**，取值为 0 或 1 表示是否执行该步骤；将检测费用、拆解费用和零件单价等设置为**决策参数**。在一轮决策中，由于变量组合较少，可以通过**枚举演绎法**获得各组合精确的利润值，比较后得出最后决策方案。考虑将拆解回收的零件再次投入生产线继续获利，由于回收的零件来自检测不合格的成品，因此其次品率将发生改变，利用**贝叶斯公式**求出其后**验概率**，作为条件可再次使用与第一轮类似的枚举演绎法重置决策方案，不断**循环利用**，直到回收的零件次品率过高，不能为企业带来收益。得到在初始各 1000 个零件的条件下，**情况一的最大利润为 13491 元；情况二的最大利润为 6640 元；情况三的最大利润为 12117 元；情况四的最大利润为 11815 元；情况五的最大利润为 8065 元；情况六的最大利润为 18562 元；**

针对问题三，仍然延用问题二设置**决策变量**和**目标函数**的方式，由于零件和工序的增多，引进**遗传进化算法**求解最优的决策方案。计算时为考虑回收品的利用，规定从零件加工直到第一次成品拆解为**一轮完整的决策**。仍然引入**贝叶斯公式**建立相邻两轮的次品率推导关系，则可通过题目给定的初始次品率，不断得到下一轮的决策条件。对于题目给出的 8 个零件，两道工序的具体情况，求得了最优方案，在**初始各零件数量均为 1000**的情况下，企业将**盈利 52490 元**，在生产过程中一共得到了**996 件产品**，其中**897 件为正品**，**市场次品率为 0.0994**。随后，思考并探究了在 m 个零件， n 道工序的情况下，建立决策的模型仍然可以沿用问题三的思路，即**二进制整数规划结合贝叶斯公式**。

针对问题四，使用问题一中推导得出的**最小样本公式**确定抽样个数，通过计算机的随机生成相应个数的**随机数**，根据原题目信息，确定对应比例的数为次品，记录与问题二、三相同情况下的**样本次品率**，代入重复上述算法。重新计算得**问题二各情况的最大收益为 13793 元，4403 元，10313 元，11862 元，13479 元，20674 元；重新计算得问题三的最大收益为 55318 元，共生产 993 件产品，其中 900 件正品，次品率为 0.0937。**

关键词： 精确二项检验 二进制整数规划 枚举演绎法 贝叶斯公式 遗传算法

一、 问题重述

1.1 问题背景

某企业在生产一种畅销电子产品时，需要购买零配件一和零配件二组装成成品。两种零件均存在不合格品；对于组装得到的成品，只要其中一个零件不合格，则成品一定不合格，两个零件都合格的情况下，成品也可能不合格。因此企业需要花费检测费用，在过程中选择检测两种零件以及最终成品的合格性。此外，对于确定不合格的成品，企业可以选择报废丢弃，或花费拆解费用对其中的零件进行回收。

1.2 问题要求

问题一：企业决定自行付费，对一批零配件的次品率进行抽样检测，以判断该批次零配件的次品率是否不超过供应商声称的标称值，从而决定是否买入这批零配件。为尽可能减少企业的检测费用，设计检测次数尽可能少的抽样检测方案。

问题二：建立优化模型，为企业生产过程的关键步骤做出决策，决策步骤如下。

零配件检测：决定是否对零配件 1 和/或零配件 2 进行检测。若不检测，所有零配件将直接用于装配；若检测，则不合格品将被丢弃。

成品检测：对于装配完成的成品，决定是否进行检测。若不检测，成品将直接进入市场；若检测，则只有合格的成品进入市场。

不合格成品处理：对于检测出的不合格成品，决定是否拆解。若不拆解，成品将被丢弃；若拆解，则拆解后的零配件将重新进入生产流程。

退货处理：对于市场上售出的不合格品，企业无条件更换，并承担相应的调换成本。退回的不合格成品根据上述决策处理。

为帮助企业更好应对不同情况，问题给出六种企业遇到的具体情况，要求设计对应的具体决策方案，并阐释决策原因，量化决策指标，更好地帮助企业建构合适的生产体系。

问题三：对 m 道工序， n 个零配件的情况，已知各种零件、半成品和成品的次品率，要求建立生产过程决策模型，提出决策方案设计方法。题中给出 8 个零件、2 道工序情况的具体生产参数，利用这些参数计算该情况时理想的决策方案，阐明决策原因，量化决策指标。

问题四：利用适合的抽样方法（例如，问题一中的抽样方法）重新计算问题二和问题三中零件、半成品和成品的次品率，并重新考虑问题二和问题三。

二、 问题分析

2.1 问题一

已知供应商给出的零件次品率，设计适当的抽样检测方法，采用单侧假设检验的方法通过样本估计总体。以频率估计概率，10%可认为是每个零件不合格的概率，进而单个零件是否合格服从 0-1 分布，全体样本中的不合格品服从二项分布。通过选择合适的统计量，并利用相应的置信度计算接受域和拒绝域，可判断声称的不合格率是否真实。此外，要求尽量减少被检测零件数量，意味着检测样本数量不能过大，但同时还需要保障检测结果的稳定性和准确性，不能一味减小数量。在此过程中，可对零件服从的分布进行正态近似，设定可接受的最小误差，确定最小的抽样数量。

2.2 问题二

已知在生产过程中零件和成品的购买销售价格、次品率、各项工序成本损耗和售后损失，要求对生产过程作四步合理的决策，分别是检测零件一、检测零件二，检测成品和拆解不合格成品。首先基于两种零件一比一装配的实际情况，假设两种零件初始提供的数量相同；在不同的决策条件下，将生产利润和流入市场的次品率设置为被优化的目标函数，希望获得的收益尽量高，且市场次品率尽量低；将设置决策为 0/1 变量表示是否进行此步骤。若有零件被回收，这部分零件的次品率将由于前面检测步骤的选择而发生改变，还应该考虑对其进行新一轮生产。在对回收品的新一轮生产中，决策方法应该根据实际情况的变化重新选择。最后对回收零件的重生产循环设置恰当的终止准则。

2.3 问题三

在问题二的基础上，问题三将问题拓展为 m 道工序、 n 个零件，优化关键步骤获得最佳方案，决策步骤更多，方案组合更复杂，需要获得一般性的原则和方法指导决策。要求对 8 个零件 2 道工序的生产流程确定最优执行方案，为量化比较各种方案，仍然假设生产初始状态 8 种零件数量相同，随后设置企业收益为决策指标，设置 0/1 作为决策变量，表示是否执行该步骤，优化后分析市场次品率，验证决策方案的正确性。这一问仍然需要回收零件和半成品再次利用，以尽可能减少成本损失，同时回收的零件和半成品的次品率将相较于上一轮循环初始而发生改变，需要计算更新，并重新选择决策方案。最后，设置合理的准则，停止对回收零件的无利润利用。

2.4 问题四

利用计算机随机数或其他合理的仿真方法，参考问题二和问题三中给定的零件、半成品和成品次品率为总体次品率，选择适当的样本数量，模拟抽样检测过程，并识别次品，可计算得到样本次品率。考虑在产品检测时应存在相应花费，则需要在保证样本代表性的情况下尽可能减少样本数量。同时为保证抽样检测的稳定性，应多次重复抽样过程取平均，将平均值作为各部件次品率，代入之前的算法中，可计算得到相应结果。

三、 模型假设

为了对模型进行量化计算与合理简化，我们建立了以下模型假设：

（1）在问题二的分析中，基于一比一装配的实际情况，我们假设生产过程中初始提供的零件一和零件二数量相等。

（2）问题二和问题三中，若进行检测后，部分次品被剔除造成了零件或半成品数量不相等，我们认为按其中数量较小者进行装配成下一阶段产品。当半成品存在剩余时，我们将全部回收，考虑拆解出其中可能的合格零件；而对于零件，只在第一轮生产时将多余零件回收仓库，且不算入成本，处于其他轮数时则将多余的部分丢弃，防止收入仓库影响整体质量。

（3）问题二和问题三中，假设工厂在对零件进行多次回收利用后，若利用某一次回收的零件和半成品已不能使工厂获利，则认为这部分零件的品质过差，将其丢弃，并计算相应的损失，从总收益中扣除。

（4）问题四中，假设待抽样检测的半成品和成品都是由合格的零配件、半成品装配而成，零配件次品率不影响半成品、成品的次品率，半成品次品率不影响成品次品率，因此抽样检测方法得到的半成品和成品次品率就是装配造成的次品率。

四、符号表示

表 1 符号说明

符号	含义	单位
n	初始设的零件数量	个
S_{ij} 、 S_k	选择是否检测、拆解 (0/1)	决策变量
p_i 、 p'_i 、 p_{ij} 、 p'_{ij}	原零件次品率和 拆解回收的零件次品率	百分比%
b_i 、 b_{ij}	零件购买成本	元
t_i 、 t_{ij}	产品检测成本	元
C_{ij} 、 C_k	参与生产的零件数量	个
R_i	零件剩余数量	个
C_{zi} 、 C_{ci}	第 i 环节的正品数量、次品数量	个
n_{ij}	再决策时初始零件数量	个
a_i	某阶段节省的成本	元
c_i	某阶段中花费的成本	元
W	总利润	元
f_i	某阶段的总成本	元
A_{ij}	第 i 个环节第 j 个零件是次品	事件
B_i	第 i 个半成品是次品	事件
p_s	样本次品率	百分比%
p_{sk}	第 k 次实验的样本次品率	百分比%

注：其他符号将在下文中具体说明

五、模型建立与求解

5.1 问题一模型建立与求解

5.1.1 模型建立

由题目条件可知，供应商声称零件次品率的标称值为 10%，则对于抽样后的全体样本，认为其中次品数量应该服从参数为 $p = 0.1$ 的二项分布。为检验标称值是否与真实情况符合，对全体产品进行抽样，并采用单侧假设检验的方法来判别。

(1) 使用正态近似的最小样本公式确定抽样数量

在假设检验的过程中，为保证检测结果的精确性和稳定性，收取的样本数量不能过少；而另一方面，当样本独立抽取，并且数量适宜时，可以对服从二项分布的随机变量进行正态近似。因此在正态近似的前提下推导出最小样本公式，最后用正态近似的条件检验结果，以验证对二项分布近似正态分布的操作理论上可行。

最小样本公式推导如下：

在抽样检测时，我们关注抽出的所有零件中次品所占比例：

$$\hat{P} = \frac{\text{事件发生的次数}}{\text{样本总数}} \quad (1)$$

计算次品比例的样本误差为：

$$SE_{\hat{P}} = \sqrt{\frac{P_0(1 - P_0)}{n}} \quad (2)$$

当样本总量 n 值合适时，取二项分布为正态分布近似，可得到在信度为 α 下的置信区间为： $\hat{P} \pm Z_{\alpha} \cdot SE_{\hat{P}}$

再自主取定误差范围 E ，该误差范围表示样本检测时，预计可接受的与给定检测参数的差值；根据专家经验， E 一般可取 0.01 至 0.05，使得：

$$Z_{\alpha} \sqrt{\frac{P_0(1 - P_0)}{n}} \leq E \quad (3)$$

两边平方后得到样本量计算公式：

$$n \geq \frac{Z_{\alpha}^2 P_0(1 - P_0)}{E^2} \quad (4)$$

则可得到检验所需要的最小样本量为：

$$n = \frac{Z_{\alpha}^2 P_0(1 - P_0)}{E^2} \quad (5)$$

(2) 合理的抽样方法

厂家在确定合适的抽样数量后，需要采用适当的抽样方法获得相应数量的样本。恰当的抽样方法是保证检测高效且准确的必要环节。对于题目中的情况，由于所有零件都是独立的，且无其他因素影响，零件被抽到的概率相同，一般常用的抽样方法有：简单随机抽样和系统抽样。

简单随机抽样：为一批待抽样的零件每一个都分配一个编号，使用随机数表或计算机生成的随机数来选取样本。

系统抽样：按照固定的间隔，对编号的零件每隔一定数量选取零件。

(3) 利用精确二项分布进行假设检验

抽取相应数量的样本并检验其合格率，检验方法为假设检验。首先确定原假设和备择假设，再采用精确二项检验方法，这种方法基于二项分布离散的特点，无需正态拟合，比近似拟合的效果更加精确。

对于一次具体的抽样，确定抽样总数 n 和含有的次品数量 x ，再确定置信水平 $1 - \alpha$ ，随后利用二项分布的累积函数(CDF)确定相应置信度下精确的置信区间，区间的上界和下界分别为：

$$P(X \leq x) = \sum_{k=0}^x \binom{n}{k} p^k (1-p)^{n-k} \geq 1 - \alpha \quad (6)$$

$$P(X \geq x) = \sum_{k=x}^n \binom{n}{k} p^k (1-p)^{n-k} \geq 1 - \alpha \quad (7)$$

比较厂商提供的比例参数值 p_0 和置信区间，若 p_0 落在置信区间内，则不能拒绝原假设；若 p_0 不在置信区间内，则拒绝原假设。

5.1.2 模型求解

(1) 在 95%的信度下认定零配件次品率超过标称值，则拒收这批零配件。

由题目，可得到该检验的原假设和备择假设分别为：

$$H_0 : p = 10\%$$

$$H_1 : p > 10\%$$

置信水平 $1 - \alpha$ 为 95%，则 α 为 0.05，对应的 Z_α 为 1.645。分别取误差范围 E 为 0.01, 0.02, 0.03, 0.04, 0.05，计算对应的样本数量 n 值。

根据所需的样本数量，采取简单随机抽样或系统抽样进行样本抽取，并检测其次品比例。由于该检验只考虑超过标称值的情况，因此为右单侧假设检验。编程计算基于 matlab 软件进行，调用 Statistics and Mechine Learning Toolbox 中的 binofit 函数和 binoinv 函数可直接计算精确二项检验的相应 α 下的双侧边界；考虑到问题属于单侧检验，因此需要将 α 翻倍为 0.1，并且取上界作为分界点。

若样本次品占比大于上边界，则拒绝原假设，即拒绝这批零件；若样本次品率小于上边界，则不拒绝原假设，即接收这批零件。

根据最小样本公式以及误差范围，计算出一组抽样个数 n 如下表 2 所示。然后用精确二项检验方法观察各误差范围的敏感度，选择较合适的方案。

模拟实验中样本次品数量为抽样个数的 10%（取整），代入公式（6）可计算得到 95%的信度下次品比例 p 值的上限和最大的次品数量。不同误差范围内样本次品率上限和最大次品数如表 2 所示。

表 2 95%信度不同误差范围内样品次品率上限和最大次品数

误差范围 E	抽样个数 n	p 上限	最大次品数 x
0.01	2435	0.1104	268
0.02	609	0.1207	73
0.03	271	0.1348	35
0.04	152	0.1479	21
0.05	97	0.1563	15

(2) 在 90%的信度下认定零配件次品率不超过标称值，则接收这批零配件。
由题目，可得到该检验的原假设和备择假设分别为：

$$H_0 : p = 10\%$$

$$H_1 : p < 10\%$$

置信水平 $1 - \alpha$ 为 90%，则 α 为 0.1，对应的 Z_α 为 1.282。分别取误差范围 E 为 0.01, 0.02, 0.03, 0.04, 0.05，计算对应的样本数量 n 值如表 3 所示。

同上，根据所需的样本数量，采取简单随机抽样或系统抽样进行零件抽取，并检测次品比例。由于该检验只考虑不超过标称值的情况，因此为左单侧假设检验。调用 `binofit` 函数计算置信度为 2α 下的双侧边界；考虑到问题属于单侧检验，取下界作为分界点。

若样本次品占比大于下边界，则接受原假设，即拒绝接收这批零件；若样本次品率小于下边界，则接受备择假设，即接收这批零件。

根据最小样本公式以及误差范围，计算出一组抽样个数 n 如下表 3 所示。然后用精确二项检验方法观察各误差范围的敏感度，选择较合适的方案。

模拟检验得到的样本次品数量为抽样个数的 10%（取整），代入公式（7）可计算得到 90%信度下次品比例 p 值的下限和最小次品数。不同误差范围内样本次品比例下限和最小次品数如表 3 所示。

表 3 90%信度不同误差范围内样品次品率上限和最大次品数

误差范围 E	抽样个数 n	p 上限	最小次品数 x
0.01	1479	0.0901	133
0.02	369	0.0805	30
0.03	164	0.0687	12
0.04	92	0.0599	6
0.05	59	0.0543	3

5.1.3 结果分析

在计算相应误差范围的最小抽样数量时，我们对二项分布使用了正态近似来推导公式，则此时需检验在具体问题下近似的合理性。根据正态近似的要求，二项分布 $B(n, p)$ 需要同时满足如下条件：

$$np \geq 5 \quad (8)$$

$$n(1 - p) \geq 5 \quad (9)$$

对表 1 和表 2 中的数据进行验证, 结果表明在各种误差范围下, 算得的抽样个数都满足正态近似条件, 即本题适合使用正态近似计算最小样本数量。考虑到厂商希望尽量减少成本, 则推荐使用 E 值为 0.05 的情况, 对应在情况一 95% 的信度下抽样 97 个, 在情况二 90% 的情况下抽样 59 个。

5.2 问题二模型建立与求解

5.2.1 模型建立

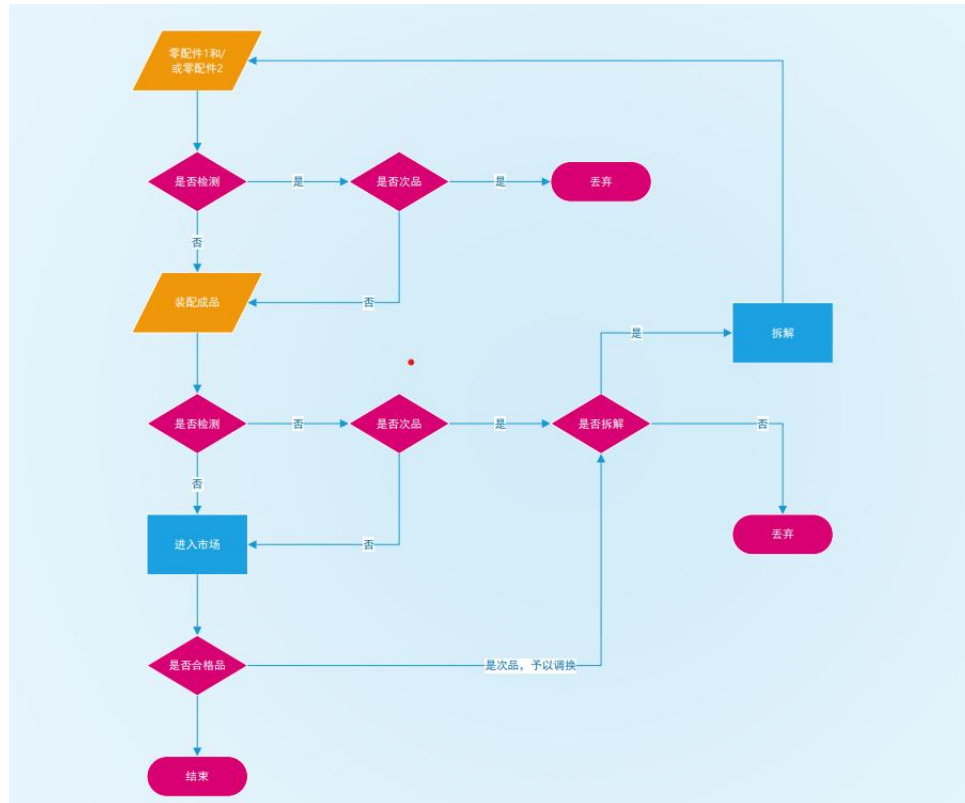


图 1 企业决策流程图

由题目条件, 已知生产周期中的花销包含: 零件购买单价、成品装配成本、零件和产品检测费、产品拆解费以及次品调换费; 而合格的成品出售后将为企业获利。为量化处理并进行有效比较, 并参考零配件一比一装配的实际情况, 引入零配件初始数量相同的条件给出决策方案。由于认为对企业最有利的策略能实现尽可能大的产品净收益, 同时尽可能减小流入市场产品的次品率 (市场次品率) 以提升企业形象, 因此将企业利润设为目标函数, 采用二进制非线性整数规划方法求得最优解, 得到 0/1 序列表示是否执行某一步骤的决策方案, 最后对市场次品率进行考察。

(1) 成品的第一轮生产与决策方案

初始情况下, 有数量相同的零件一和零件二各 n 件, 设定决策变量 S_{11} , S_{12} , S_2 和 S_3 分别表示以下选择: 零件一是否检测, 零件二是否检测, 成品是否检测, 不合格成品是否拆解回收。各决策变量的取值为 0 或 1。下面建立某一决策下产品利润的表达式, p_1 、 p_2 和 p_3 分别表示两种零件的次品率和成品次品率。

根据决策, 计算检测后零件数量的变化。

零件一数量: (若不检测则 C_1 数量等于 n , 否则等于 $(1 - p_1) n$)

$$C_1 = S_{11} \times (1 - p_1) \times n + (1 - S_{11}) \times n \quad (10)$$

零件一的剩余数量：

$$R_1 = n - C_1 \quad (11)$$

零件二数量：

$$C_2 = S_{12} \times (1 - p_2) \times n + (1 - S_{12}) \times n \quad (12)$$

零件二的剩余数量：

$$R_2 = n - C_2 \quad (13)$$

由于装配成品分别需要一个零件一和一个零件二，最终装配的成品数量应与 C_1 和 C_2 中的数量较小者一致，可得成品数量：

$$C_3 = \min(C_1, C_2) \quad (14)$$

对于组装过程中剩余的零件，将返回仓库，不计入生产过程，因此零件购买成本中需要减去剩余零件的购买成本，减小量为：

$$a_1 = R_1 \times b_1 + R_2 \times b_2 \quad (15)$$

设零件一和零件二的购买成本分别为 b_1 和 b_2 ，两种零件的检测成本分别为 t_1 和 t_2 ，将一对零件装配为成品的费用为 z ，则零件购买、检测以及装配成品共花费：

$$c_1 = b_1 \times n + b_2 \times n - a_1 + t_1 \times S_{11} \times n + t_2 \times S_{12} \times n + z \times C_3 \quad (16)$$

设单个成品的检测费用为 t_3 ，则全部成品的检测费用为：

$$c_2 = S_2 \times t_3 \times C_3 \quad (17)$$

利用表格中提供的次品率 p_1 、 p_2 和 p_3 可分别计算出零配件的合格率。

零件一：

$$r_1 = S_{11} + (1 - S_{11}) \times (1 - p_1) \quad (18)$$

零件二：

$$r_2 = S_{12} + (1 - S_{12}) \times (1 - p_2) \quad (19)$$

成品中正品概率为两个零件和装配过程都合格的概率，可得到成品中正品和次品数量分别为：

$$Cz_3 = r_1 \times r_2 \times (1 - p_3) \times C_3 \quad (20)$$

$$Cc_3 = C_3 - Cz_3 \quad (21)$$

售卖正品可获得利润，成品售价为 m ，则企业通过销售正品可获得的利润为：

$$w = m \times Cz_3 \quad (22)$$

对于检测出的次品，若选择拆解回收，则每个次品花费拆解费用 j ，被回收的零件没有丢弃而留作下一轮使用，因此在本轮生产中这批零件节省了成本。

拆解成本：

$$c_3 = S_3 \times Cc_3 \times j \quad (23)$$

回收节省成本：

$$a_2 = S_3 \times Cc_3 \times (b_1 + b_2) \quad (24)$$

若在步骤二未决定检测成品，则将有次品进入市场，并带来调换成本（如物流成本、企业信誉等），设每个产品的调换成本为 t_4 ，则调换产生的总损耗为：

$$c_4 = t_4 \times (1 - S_2) \times Cc_3 \quad (25)$$

综合上述各项盈利与花销，可得到总利润为 W ：

$$W = w + a_2 - c_1 - c_2 - c_3 - c_4 \quad (26)$$

最后以总利润 W 为目标函数， S_{11} 、 S_{12} 、 S_2 、 S_3 为优化变量，应用二进制非线性整数规划模型求解 16 种方案在第一轮生产过程中的总利润，若本轮回收的零件没有利润可得，则选取获得利润最高的方案考察其市场次品率、生产正品率等指标，否则进入下一轮对回收零件的再生产销售过程，完善决策方案。

（2）对回收零件再利用的多轮生产规划建模

在决策过程中，当我们在步骤三选择对不合格成品拆解并回收利用时，需要对回收零件的去向进行明确规划，考虑是否检测零件合格性，并再次装配为成品，再销售后为企业获利。因此，我们需要建立多轮生产模型，使尽可能提高零件的利用率，减少丢弃。

建模时，由于生产时零件时一比一装配的，回收的零件中两种零件数量也将相等，情况与上述第一轮装配时类似；但对于从已经检测为不合格品的成品中回收的零件，其被确认为次品的概率，将根据前一轮两种零件的检测情况，相较于仓库中未经处理的零件，偏高或偏低。我们可以根据已知的初始零件次品率 p_1, p_2 ，结合在两零件皆为正品时成品不合格的概率 p_3 ，通过贝叶斯公式，计算回收零件中的次品率，以此为回收零件的生产决策建模提供前提条件。

首先，结合前一轮生产的零件检测情况，建立相邻两轮生产的零件次品率关系式。设前一轮生产时，两种零件的次品率分别为 p_1, p_2 ，正品装配的成品不合格率为 p_3 ，贝叶斯公式如下：

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \quad (27)$$

1)前一轮两种产品都未检验，成品为次品的概率为：

$$P(B) = 1 - (1 - P(A_1))(1 - P(A_2)) + P(A_3)(1 - P(A_1))(1 - P(A_2))$$

其中事件 A_1 表示零件一为次品，事件 A_2 表示零件二为次品，事件 A_3 表示成品装配为次品，事件 B 表示成品为次品；当 $A_i (i = 1, 2)$ 发生时，事件 B 一定发生，则 $P(A|B)$ 恒等于 1。据此，得到下一轮零件一和零件二次品率的表达式为：

$$p_1' = \frac{p_1}{1 - (1 - p_1)(1 - p_2) + p_3(1 - p_1)(1 - p_2)} \quad (28)$$

$$p_2' = \frac{p_2}{1 - (1 - p_1)(1 - p_2) + p_3(1 - p_1)(1 - p_2)} \quad (29)$$

2)前一轮零件一检测而零件二未检测，成品为次品的概率为：

$$P(B) = P(A_2) + P(A_3)(1 - P(A_2)) \quad (30)$$

由于零件一在上一轮经过了检验，因此回收得到的都是正品，次品率为 0，零件二次品率的则代入公式得：

$$\begin{aligned} p_1' &= 0 \\ p_2' &= \frac{p_2}{p_2 + p_3(1 - p_2)} \end{aligned} \quad (31)$$

3)前一轮零件一未检测而零件二检测，成品为次品的概率为：

$$P(B) = P(A_1) + P(A_3)(1 - P(A_1)) \quad (32)$$

由于零件二在上一轮经过了检验，因此回收得到的都是正品，次品率为 0，零件一次品率的则代入公式得：

$$\begin{aligned} p_1' &= \frac{p_1}{p_1 + p_3(1 - p_1)} \\ p_2' &= 0 \end{aligned} \quad (33)$$

4)前一轮两种零件都经过检测，则回收的零件都是通过检测的正品，次品率都为 0。

综上所述，将四种情况下相邻两次的次品率关系整理在表四中。

表 4 回收零件的零件一和零件二次品率

S_{11}	S_{12}	p_1'	p_2'	p_3
0	0	$\frac{p_1}{1 - (1 - p_1)(1 - p_2) + p_3(1 - p_1)(1 - p_2)}$	$\frac{p_2}{1 - (1 - p_1)(1 - p_2) + p_3(1 - p_1)(1 - p_2)}$	10%
1	0	0	$\frac{p_2}{p_2 + p_3(1 - p_2)}$	10%
0	1	$\frac{p_1}{p_1 + p_3(1 - p_1)}$	0	10%
1	1	0	0	10%

求得回收零件下一轮生产的次品率后，重复运用二进制非线性整数规划寻找最优决策方法，将企业收益设定为目标函数，同时严格考察市场次品率作为重要指标。具体的建模步骤与(1)描述的第一轮一致。

经过多轮循环后，预计将在某一轮，回收次品率过高，将使得在加工出的成品将使工厂亏本，甚至不再继续产出新的正品，此时结束多轮生产。考虑到最后的零件质量不足，已不能有效成产成品，因此选择将这部分产品丢弃，并记录亏损。

5.2.2 模型求解

输入：初始化两个零件的数量

已知参数：两个零件的购买价格、次品率以及检测费用，成品的装配次品率、检测成本、市场售价以及调换费、拆解费。

输出：最优的决策方案以及对应的方案指标，如市场次品率等。

优化变量：四个 0/1 决策变量，分别是是否检测零件一、是否检测零件二、是否检测成品、是否回收。

求解方法：首先用决策变量建立最终利润的优化函数，即所有决策后产生的费用由决策变量的 0/1 表示是否支出。则利润函数建立思路为：

1、计算进入生产过程的实际零配件数量。由于是否检测零配件的决策不同，被检测的合格零配件数量将少于初始零配件数量，根据一个零配件 1 和一个零配件 2 装配的生产原理，将满足能进入生产过程的数量少的零配件的需求，多余的零配件将返回仓库，不作使用。

2、计算零配件花费的成本。零配件花费的成本包括购买成本、检测成本和装配成本，已知这三个成本的单价，乘以实际进入生产过程的零配件数量，得到零配件花费的成本总和。

3、计算成品的合格率和次品率。由于产生次品的原因有：零配件一是次品、零配件二是次品、装配出次品，故成品次品率为三者概率的并集，合格率为次品率的补集。

4、计算成品花费的成本。成品包括合格品和次品两种。对于未检测的成品需要花费的成本为检测成本，乘以所有成品数量得到检测费用；对于经过检测或退货的不合格品需要花费的成本包括调换费用和拆解费用，乘以不合格品数量得到不合格品花费的成本总和。两者相加为成品花费的成本总和。

5、计算成品的收益。由于售出的不合格品需无条件调换成合格品，故可以认为成品中只有合格品部分成功售出并盈利，故成品收益为成品*合格率*售价。

得到利润函数后，由于当前决策变量较少且为 0/1 的整数规划问题，可以通过排列组合算出 16 种情况对应的上述成本与盈利，得到 16 个利润总和的结果，取其中利润最大值为一轮的最优方案。

经过第一轮决策后，选择拆解的方案会剩余一部分回收零部件，将对于这部分回收零部件重复上述操作，区别在于：

1、零配件数量变化：零部件数量等于拆解的不合格品数量。

2、零配件次品率变化：由于这一批零配件是由不合格品拆解而来，如果上一轮决策没有检测零配件，则零配件的次品率将上升，此时根据贝叶斯公式可以计算得到已知成品是不合格品的条件下零配件是次品的概率，次品率变为零件原次品概率与成品不合格品概率的比值。

3、引入终止条件：对于不断产生回收零件的决策方案，考虑到企业盈利的目的，设置只有当这部分零部件至少能生产一个成品且利润为正数的情况下继续加工，否则终止加工。对于终止加工后仍剩余的零件作丢弃处理，花费它们的购买成本但没有收益。

修改以上区别后，更新输入为新的零配件数量、新的零配件次品率，代入利润函数，引入终止条件即可循环求得第一轮所有方案的回收零部件的最佳处理办法，即最佳决策方案和最大利润。，求解方法与上述一致。

将第一轮每个方案所得利润与其对应的回收零件后续最佳处理办法相结合可得 16 种方案的总利润，取总利润最大值的决策方案为推荐决策方案。

过程衡量指标说明：在优化决策变量求解最大利润的目标函数中，包含市场售出成品数量、市场售出次品数量、回收节约成本等表达式，将求解得到的最优决策方案代入表达式中可获得一些体现决策方案性能的指标，可以用于观察该最优方案的实施效果，供企业了解。

取每个零件的初始数量为 1000，得到如表 5 的推荐决策方案结果：

表 5 初始零件数量为 1000 的六种情况推荐决策方案结果

情况	最大利润	推荐决策方案	合格品数量	市场次品率	理论 市场次品率
1	13491	$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$	851	21.32%	27.1%
2	6640	$\begin{bmatrix} 1 & 0 & 0 & 1 \end{bmatrix}$	512	36.00%	48.8%
3	12117	$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$	851	0.00%	27.1%
4	11815	$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$	800	0.00%	48.8%
5	8065	$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$	781	0.00%	35.2%
6	18562	$\begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}$	857	14.30%	14.3%

5.2.3 结果分析

分析表 5 得，由于以企业盈利性质为优先考虑因素，因此本次优化决策变量选择利润最大的决策方案。同时，为了满足最大化利润，决策方案要能根据具体情况调节，尽可能提高零配件利用率和生产成品数量增加销售量，尽可能降低销售的次品率减少产品售后的损失。接下来分析每种情况的推荐方案的合理性和优越性。

方案一中由于零配件和装配的次品率较低，且产品售后费用较低，故决定不检测零配件，不检测成品；而由于被退回的不合格品仍有剩余价值，其再次装配成成品能产生比当前损失高的利润，故决定拆解不合格品。由合格品数量可知该决策零配件利用率高，但由于前期没有检测，市场次品率优化不明显。

方案二中由于零配件和装配的次品率都较高，综合考量总利润的影响，故决定检测零配件一，不检测零配件二和成品；由于被退回的不合格品仍有剩余价值，能再产生比当前损失高的利润，故决定拆解不合格品。由合格品数量可知该情况下零配件利用率也比较理想，且由于有零配件的检测环境，市场次品率减少 25%左右。

方案三中由于不合格品的调换费用高，故决定不检测零配件，检测成品；由于检测出不合格品再利用仍能带来正利润，故决定拆解不合格品。由合格品数量可知该决策零配件利用率高，且市场次品率得到严格保障。

方案四中由于零配件和装配的次品率都较高，且不合格品的调换费用高，根据计算结果决定检测零配件一、零配件二，检测成品。该方案下所有不合格品的零配件都是正常的，再装配成功率高，再利用能带来正利润，故决定拆解不合格品。由合格品数量可知该决策零配件利用率高，且市场次品率得到严格保障。

方案五中零配件二次品率较高，综合考量总利润的影响，故决定不检测零配件，检测成品；检测得到的不合格品仍能再利用带来正利润，故决定拆解不合格品。由合格品数量可知该决策零配件利用率高，且市场次品率得到严格保障。

方案六中由于零配件和装配的次品率都低，考虑检测成本与实际效果决定不检测零配件，不检测成品；被退回的不合格品拆解费高，不能带来正利润，故决定不拆解不合格品。由合格品数量可知该决策零配件利用率高，市场次品率虽没有改善，但能保持初始较低水平。

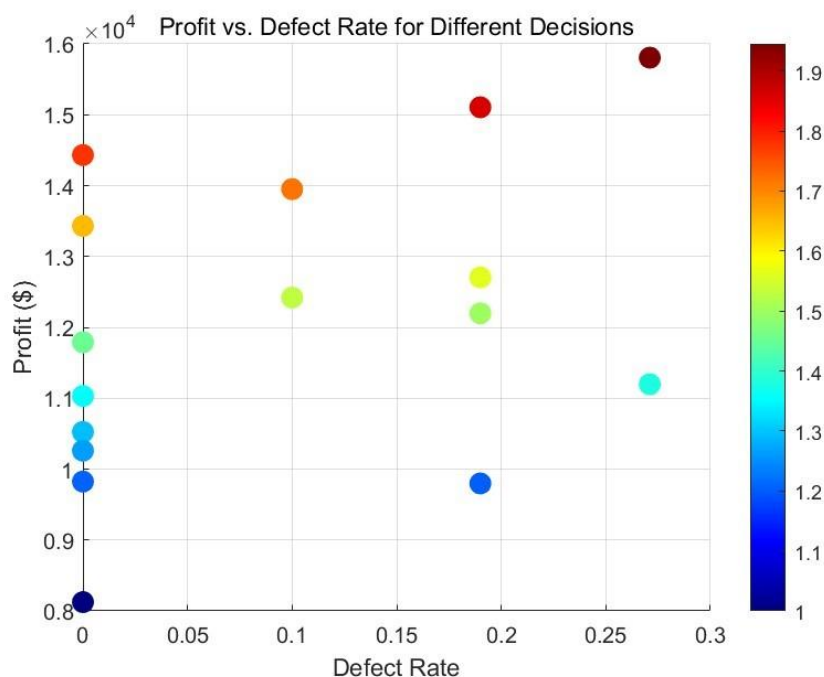


图 2 第一种情况下 16 种决策方案对比的热力散点图

由图 2 可以比较第一种情况下 16 种决策方案的利润以及市场次品率的分布，图中纵坐标为方案的总利润，横坐标为市场次品率。由图可以得到在第一种情况下，利润和市场次品率存在一定矛盾关系，随着市场次品率的减小，对之监管的检测费用将增加，导致总利润受到一定亏损。本次建模以企业盈利性质为最终目的，优先考虑最大化企业利润。

5.3 问题三模型建立与求解

5.3.1 模型建立

根据题目要求，在问题二的基础上进行扩展，当生产流程含有 m 个零件， n 个步骤时，对生产流程的最佳决策方案进行建模求解。

针对本题考察的具体情况，初始有 8 个零配件，通过第一道工序生产得 3 种半成品，再经第二道工序加工出成品。分析可得生产过程中能够进行决策的阶段如下：

1) 对零配件（零配件 1 至零配件 8）是否进行检测，未选择检测的零配件将直接进入第一阶段的装配环节；否则将检测出的不合格零配件丢弃；

2) 对装配好的半成品（半成品 1 至半成品 3）是否进行检测，如果不检测，半成品直接进入第二轮装配；否则只有检测合格的半成品进入到第二轮装配；

3) 对检测出的不合格半成品是否进行拆解，如果不拆解，直接将不合格半成品丢弃；否则对拆解后的零配件，重复步骤(1)和步骤(2)；

4) 对装配好的成品是否进行检测，如果不检测，成品将直接进入市场；否则只有检测合格的成品进入到市场；

5) 对检测出的不合格成品是否进行拆解，如果不拆解，直接将不合格成品丢弃；否则对拆解后的半成品，重复步骤(2)，步骤(3)和步骤(4)；

由于存在对回收零件的再次利用以及重新决策，首先对一轮生产的起始和终止做出明确规定。认为生产从 8 种初始零件开始，先决定是否对其进行检测，随后将剩余零件装配为三种半成品；决定是否对半成品检测和拆解，回收相应的零件，将剩余的半成品投入成品的生产；决定成品是否检测并拆解，若拆解则回收得到的半成品，一轮生产到此终止。

(1) 成品的第一轮生产和决策方案

根据题目特点，采用二进制非线性整数规划方法建模评估各决策方案，针对每一步决策，为其设计一个取值限定在 0 或 1 的决策变量，随后可表示出给定初始零件数量下，通过一轮生产所获得的收益。下面具体推导生产利润的计算表达式。

首先设置决策变量 $S_{1i}(i = 1, 2, 3, 4, 5, 6, 7, 8)$ 表示第 i 个零件是否进行拆解，假设初始提供的零件数量都为 n ，则可得到检测步骤后的零件数量如下。

$$C_{1i} = S_{1i}(1 - p_{1i})n + (1 - S_{1i})n$$

其中 C_{1i} 表示第 i 种零件剩余的数量， p_{1i} 表示第 i 种零件的次品率。

决策一后，组装的三种半成品数量分别：

$$C_{21} = \min(C_{11}, C_{12}, C_{13}) \quad (34)$$

$$C_{22} = \min(C_{14}, C_{15}, C_{16}) \quad (35)$$

$$C_{23} = \min(C_{17}, C_{18}) \quad (36)$$

其中 $C_{2j}(j = 1, 2, 3)$ 分别为第 j 种半成品的数量。

此时，可计算现阶段的成本，包含零件的购买成本，检测成本和组装成本，计为 f_1 ，设各零件的检测费用和购买费用分别为 t_{1i} 和 b_{1i} ，第 j 种半成品的装配费用为 z_{1j} 。

$$f_1 = \sum_1^8 (t_{1i} n S_{1i}) + \sum_1^8 (b_{1i} n) + \sum_1^3 (z_{1j} C_{2j}) \quad (37)$$

而由于检测可能导致零件数量不同，将多余未使用的零件回收入库，可节省成本为 h_1 。

$$h_1 = \sum_1^8 (n - C_{1i}) b_i \quad (38)$$

随后计算半成品 j 组装后的正品率 r_{2j} ，设 p_{2j} 为半成品在组装时产生的次品率。

$$r_{2j} = \bigcap_i (S_{1i} + (1 - S_{1i})(1 - p_{1i})) \times (1 - p_{2j}) \quad (39)$$

则半成品的正品数量，次品数量和次品率分别为：

$$C_{z_{2j}} = r_{2j} C_{2j} \quad (40)$$

$$C_{c_{2j}} = C_{2j} - C_{z_{2j}} \quad (41)$$

$$p_j = \frac{C_{c_{2j}}}{C_{2j}} \quad (42)$$

进行决策 2 利用 S_{2j} 表示，代表是否对半成品进行检测，决策后半成品的数量改变为 C'_{2j} 。

$$C'_{2j} = S_{2j} C_{z_{2j}} + (1 - S_{2j}) C_{2j} \quad (44)$$

此时可计算半成品的检测费用 f_2 ：

$$f_2 = \sum_1^3 (S_{2j} t_{2j} C_{2j}) \quad (45)$$

下面进行决策 3，利用 S_{3j} 表示，代表是否对检测不合格的半成品进行拆解，随后可计算拆解费用 f_3 ：

$$f_3 = \sum_1^3 (S_{3j} q_j S_{2j} C_{c_{2j}}) \quad (46)$$

计算收益 h_2 ：

$$\sum_1^3 (S_{3j} S_{2j} C_{c_{2j}} \times \text{装配零件总单价}) \quad (47)$$

接下来进行成品组装，并计算成品数量 C_3 ，为各类半成品的最小值：

$$C_3 = \min(C'_{21}, C'_{22}, C'_{23}) \quad (48)$$

随后计算成品组装成本 f_4 ：

$$f_4 = z_2 C_3 \quad (49)$$

组装成品后，半成品将因为数量不同而产生剩余，我们将对这部分扮成平计算数量并进行回收，设半成品的剩余数量为 C''_{2j} ：

$$C_{2j}'' = C_{2j} - C_{2j}' \quad (50)$$

计算成品的正品数量 Cz_3 和次品数量 Cc_3 , p_{31} 为组装成品时的次品率:

$$Cz_3 = r_{21}r_{22}r_{23}(1 - p_{31})C_3 \quad (51)$$

$$Cc_3 = C_3 - Cz_3 \quad (52)$$

决策 4 用 S_4 表示, 代表是否对成品进行检测, 随后可计算成品的检测费用 f_5 :

$$f_5 = S_4 t_3 C_3 \quad (53)$$

决策 5 用 S_5 表示, 代表是否对检测出或退回的不合格成品将进行拆解, 产生拆解费用 f_6 :

$$f_6 = S_4 q_2 Cc_3 \quad (54)$$

卖出合格成品可以为企业带来收入 h_3 , 单个成品的售价为 m :

$$h_3 = Cz_3 m \quad (55)$$

而卖出未检测成品将使得企业产生调换损失 f_7 , 单个调换损失为 d :

$$f_7 = (1 - S_4) Cc_3 d \quad (56)$$

最后, 计算拆解后回收得到的半成品数量 C_{2j}''' :

$$C_{2j}''' = S_5 Cc_3 \quad (57)$$

需要注意的是, 由于我们认为一轮生产将在成品被拆解为半成品后结束, 同时由于半成品数通过零件组装而来的, 认为在第二个生产环节半成品的单价为 0。因此, 为将回收的半成品纳入考虑, 需要根据计算得到的半成品数量求出生产的隐形拆解收益 h_4 :

$$h_4 = \sum_1^3 [(C_{2j}'' + C_{2j}''') \times \text{零件单价}] \quad (58)$$

最终得到生产方案的利润目标函数, 可作为比较指标来对方案进行选择。

(2) 对回收零件再利用的多轮生产规划建模

初始状况下, 生产线上各零件数量相同且为 n , 次品率如题目中表 2 所提供, 为 10%, 各类半成品数量为 0, 次品率未显示表示, 与 8 种零件是否检测的选择有关。通过二进制整数非线性规划, 可寻找到一组最优的决策方案, 并执行第一轮生产。在第一轮结束后, 若方案对投入生产的各类零件和半成品进行了回收, 则对于被回收的部件, 我们认为同样需要考虑其再次被投入新一轮生产的决策情况。与第二问相同是, 某些回收的部件本身已经过了上一轮的检测, 且来自检测后被判定为不合格的成品或半成品, 因此对应的次品率将发生相应改变, 可以通过贝叶斯公式计算求得。

下面进行推导, 通过上一轮零件和半成品的次品率, 求出相邻第二轮生产时的各部分次品率。设上一轮时, 8 种零件的次品率分别为 $p_{11}, p_{12}, p_{13}, p_{14}, p_{15}, p_{16}, p_{17}, p_{18}$, 并分别新加工出了 n_{11}, n_{12}, n_{13} 个三类半成品; 而设加工开始时已有的三种半成品的初始次品率为 p_{21}, p_{22}, p_{23} , 初始数量分别为 n_{21}, n_{22}, n_{23} 。进行一轮加工后, 以与半成品 1 有关的生产线为例, 再次回收得到的零件一, 零件二和零件三的次品率 $p_{11}', p_{12}', p_{13}'$; 设

再次回收得到的半成品 1 的次品率为 P'_{21} 。本题中，各环节在装配时由全正品部件组成成品或半成品的概率都为 10%。

则首先可以计算第一个环节中，由零件一、二和三组装为半成品时出现次品的概率，使用 $A_{ij}(i = 1; j = 1, 2, 3)$ 表示三个零件为次品， B_1 表示第一个半成品为次品。根据上一轮三种零件的检测选择，可以计算出 B_1 发生的概率，分为以下 8 种情况。

三种零件都检测：

$$P(B) = 0; \quad (59)$$

只有零件一不检测：

$$P(B) = P(A_{11}) + 0.1(1 - P(A_{11})) \quad (60)$$

只有零件二不检测：

$$P(B) = P(A_{12}) + 0.1(1 - P(A_{12})) \quad (61)$$

只有零件三不检测：

$$P(B) = P(A_{13}) + 0.1(1 - P(A_{13})) \quad (62)$$

只有零件一检测：

$$1 - (1 - P(A_{12}))(1 - P(A_{13})) + 0.1(1 - P(A_{12}))(1 - P(A_{13})) \quad (63)$$

只有零件二检测：

$$1 - (1 - P(A_{11}))(1 - P(A_{13})) + 0.1(1 - P(A_{11}))(1 - P(A_{13})) \quad (64)$$

只有零件三检测：

$$1 - (1 - P(A_{11}))(1 - P(A_{12})) + 0.1(1 - P(A_{11}))(1 - P(A_{12})) \quad (65)$$

三种零件都不检测：

$$1 - (1 - P(A_{11}))(1 - P(A_{12}))(1 - P(A_{13})) + 0.1(1 - P(A_{11}))(1 - P(A_{12}))(1 - P(A_{13})) \quad (66)$$

则当 $P(B)$ 不等于 0 时，代入贝叶斯公式（公式编号），可得到回收的第 i 种零件新一轮的次品率，而 $P(B)$ 为 0 则表示三类零件在上一轮中都经过了检测，回收的零件次品率全部为 0。各情况下新一轮零件的次品率如表六所示。

表六 回收零件的次品率变化情况

是否检测零配件			拆解回收的零件次品率		
零件一	零件二	零件三	p'_{11}	p'_{12}	p'_{13}
1	1	1	0	0	0
0	1	1	$\frac{p_{11}}{1 - 0.9(1 - p_{11})}$	0	0
1	0	1	0	$\frac{p_{12}}{1 - 0.9(1 - p_{12})}$	0
1	1	0	0	0	$\frac{p_{13}}{1 - 0.9(1 - p_{13})}$
1	0	0	0	$\frac{p_{12}}{1 - 0.9(1 - p_{12})(1 - p_{13})}$	$\frac{p_{13}}{1 - 0.9(1 - p_{12})(1 - p_{13})}$
0	1	0	$\frac{p_{11}}{1 - 0.9(1 - p_{11})(1 - p_{13})}$	0	$\frac{p_{13}}{1 - 0.9(1 - p_{11})(1 - p_{13})}$
0	0	1	$\frac{p_{11}}{1 - 0.9(1 - p_{11})(1 - p_{12})}$	$\frac{p_{12}}{1 - 0.9(1 - p_{11})(1 - p_{12})}$	0
0	0	0	$\frac{p_{11}}{1 - 0.9(1 - p_{11})(1 - p_{12})(1 - p_{13})}$	$\frac{p_{12}}{1 - 0.9(1 - p_{11})(1 - p_{12})(1 - p_{13})}$	$\frac{p_{13}}{1 - 0.9(1 - p_{11})(1 - p_{12})(1 - p_{13})}$

在推算了零件次品率的更新方法后，下面推导半成品相邻两轮初始次品率的关系。由于每一轮生产从零件开始合成半成品，因此除了生产初始的半成品，还需要考虑在本轮中由零件新生产的半成品，分别为 n_{11}, n_{12}, n_{13} 个。假设根据选定的方案，新加工出的三类半成品为次品的概率分别为 $p''_{21}, p''_{22}, p''_{23}$ ，则可以算得在开始进行第二步加工环节的决策时，半成品的真实次品率。

半成品一：

$$p'''_{21} = \frac{p_{21}n_{21} + p''_{21}n_{11}}{n_{11} + n_{21}} \quad (67)$$

半成品二：

$$p'''_{22} = \frac{p_{22}n_{22} + p''_{22}n_{12}}{n_{12} + n_{22}} \quad (68)$$

半成品三：

$$p'''_{23} = \frac{p_{22}n_{23} + p''_{22}n_{13}}{n_{23} + n_{13}} \quad (69)$$

接下来, 类似于回收零件的次品率计算, 首先可求出 8 种情况下成品为次品的概率, 利用事件 B_2 来表示, 利用贝叶斯公式可得到更新后的半成品次品率, 用于下一轮计算。新一轮半成品的次品率如表 7 所示。

表 7 剩余半成品与拆解回收半成品的加成次品率变化情况

是否检测半成品			半成品的加成次品率		
半成 品一	半成 品二	半成 品三	p'_{21}	p'_{22}	p'_{23}
1	1	1	0	0	0
0	1	1	$\frac{p'''_{21}}{1 - 0.9(1 - p'''_{21})}$	0	0
1	0	1	0	$\frac{p'''_{22}}{1 - 0.9(1 - p'''_{22})}$	0
1	1	0	0	0	$\frac{p'''_{23}}{1 - 0.9(1 - p'''_{23})}$
1	0	0	0	$\frac{p'''_{22}}{1 - 0.9(1 - p'''_{22})(1 - p'''_{23})}$	$\frac{p'''_{23}}{1 - 0.9(1 - p'''_{22})(1 - p'''_{23})}$
0	1	0	$\frac{p'''_{21}}{1 - 0.9(1 - p'''_{21})(1 - p'''_{23})}$	0	$\frac{p'''_{23}}{1 - 0.9(1 - p'''_{21})(1 - p'''_{23})}$
0	0	1	$\frac{p'''_{21}}{1 - 0.9(1 - p'''_{21})(1 - p'''_{22})}$	$\frac{p'''_{22}}{1 - 0.9(1 - p'''_{21})(1 - p'''_{22})}$	0
0	0	0	$\frac{p'''_{21}}{1 - 0.9(1 - p'''_{21})(1 - p'''_{22})(1 - p'''_{23})}$	$\frac{p'''_{22}}{1 - 0.9(1 - p'''_{21})(1 - p'''_{22})(1 - p'''_{23})}$	$\frac{p'''_{23}}{1 - 0.9(1 - p'''_{21})(1 - p'''_{22})(1 - p'''_{23})}$

同时, 考虑在上一轮生成中由于版承平数量不一定一直, 可能就会有半成品剩余, 我们考虑将这部分剩余的半成品一起划归入下一轮生产进行考虑, 这部分半成品的次品率是不变的, 即为上一轮上产中的 $p'''_{21}, p'''_{22}, p'''_{23}$, 将回收得到的半成品由贝叶斯公式计算得到的后验概率根据数量权重与上一轮剩下的半成品进行加权, 可最终确定本轮生产开始的产成品次品率。然而, 对于除第一轮以外的其他时间, 若生产同一个半成品的零件数量不相等, 则直接将多余部分丢弃, 防止回收入库后对仓库的零件质量造成影响。

综上, 通过对上一轮生产过程关键参数的记录, 可为下一轮决策提供必要全面的考量指标, 进而可反复对回收的零件和半成品进行生产决策, 直到发现回收产品无法为工厂获利, 甚至无法有效生产出一个合格成品, 则停止会输利用, 并认为回收的零件品质过差, 将其丢弃, 同时从收益中减去损耗。

(3) 建立 m 个零件, n 个步骤时的通用决策方案

当存在多个零件和生产步骤时, 仍然可以用类似的方法对生产流程进行决策。首先仍需要列举流程中可能存在的各项决策步骤, 并为每个步骤设定取值为 0 或 1 的决策变量, 设置希望优化的目标结果为净利润最大化, 同时严格考察流入市场的次品率。由于

存在对不合格半成品和成品拆解回收的考虑，我们仍需要定义一轮生产的开始和结束，并设置合理的计算方法，求出每次对回收品再加工时新的次品率。

具体来说，当生产流程存在 n 道工序，将会产生从零配件到成品的 $n + 1$ 个检测关卡。一轮生产定义为从零配件开始，考虑是否检测，并组装为第一组半成品；随后考虑是否对这部分半成品进行检测和拆解，剩余的半成品继续流向下一道工序，组装为第二组半成品；后面的生产以此类推，直到最后一组半成品组装为成品，对成品考察是否检测拆解，回收相应的半成品，认为一轮生产到此结束。

若零件和各半成品的回收数量不等于 0，则将进入下一论生产，并重新决策。需要注意的是，我们应及时利用贝叶斯公式回收零配件开始，逐步更新次品率。除此之外，若在生产的同一步，各组分数量不同，将按其中的最小数量装配下一步产品，在这一过程中剩余的半成品需要全部回收，而对于零件，只在第一轮生产时回收多余的零件，处于其他轮数时则将多余的部分丢弃，防止收入仓库影响整体质量。

5.3.2 模型求解

输入：初始化八个零件的数量

代入各项已知参数，包括八个零件的购买价格、次品率以及检测费用，三个半成品的次品率、检测费用以及拆解费用，成品的装配次品率、检测成本、市场售价以及调换费、拆解费。

设置遗传算法参数：设置决策变量上界为 0，下界为 1，只取整数，实现 0/1 整数规划求解，定义种群大小为 100，保留 5%精英个体。

运用决策变量表达利润函数，表达方式如下：

(1) 计算决策一之后进入生产过程的实际零配件数量，以装配半成品 1 的零件 1 为例，若选择检测零件 1，则能进入生产过程的零件 1 数量为零零件 1 中的正品数，使用 0-1 决策变量与取整函数 `round` 进行表示与计算。计算完这三个零件能进入生产过程的零件数后，根据三种零件各一个组成半成品的加工原理，取三个零件数的最小值作为实际进入生产的零件数，同时也是生产的半成品数量，多余的零配件将返回仓库，不作使用。对其他加工单位处理方法一致，最后将本轮装配的半成品数量与拆解的半成品数量相加得到能够生产的半成品数量。剩余零配件的数量通过零配件数量减去本轮装配的零配件数量相减得到，在第 1 轮时，剩余零配件将被重复利用，计算相应减少的成本。

(2) 计算零配件花费的成本。零配件花费的成本包括购买成本、检测成本和装配成本，已知这三项成本对应的单价，乘以实际进入生产过程的零配件数量/装配的半成品数量，得到零配件花费的成本总和，根据是否检测，更新零配件次品率。

(3) 计算半成品花费的成本。半成品花费的成本包括检测成本，拆解成本和装配成本。首先计算每个零配件的合格率，由零配件的合格率和半成品的次品率可计算出组装的半成品的合格率，结合组装的半成品的数量与上一轮拆解的半成品数量和相应次品率可计算出目前半成品的次品率。接下来计算出半成品的检测费用与拆解费用，并根据是否检测对本成品次品率进行更新，对于除最后一轮外的半成品的拆解，计算相应减少的成本。经过拆解后，计算拆解得到的零配件数量和相应次品率。对于装配成本，首先计算可组装的成品数量，乘以组装单价得到组装成本。

由于决策的不同，半成品的数量和次品率会有不同（具体次品率变化见 5.3.1 模型建立部分），进而导致装配成品时可能出现半成品有剩余的情况，多余部分半成品将等待后续轮次的使用。因此当最少的半成品数量为 $minnum$ 时，半成品 1 的剩余数量为 $C_{21} - minnum$ ，其他半成品同理，得到剩余的半成品数量和次品率。

(4) 计算成品花费的费用。成品花费的费用包括检测成本，拆解成本和调换成本，首先由半成品的次品率和成品的次品率计算出生产出的所有成品的次品率，得到相应的正品和次品数量，由次品数量可计算出成品的拆解费用、检测成本、退回的次品数、检测出的次品数以及调换成本。

(5) 拆解回收的半成品数量和次品率更新。由于拆解回收的半成品来源于不合格的成品，其次品率将提高。更新后的半成品数量=不合格成品数量，半成品次品率

$$= \frac{\text{原半成品次品率}}{\text{成品次品率}}。$$

(6) 计算成品的收益。由于售出的不合格品需无条件调换成合格品，故可以认为成品中只有合格品部分成功售出并盈利，故成品收益为成品数量*成品合格率*售价。

(7) 计算未来可能的拆解收益，由于成品拆解收益来自后续轮次的计算，本轮利润不包含拆解收益但具有拆解成本，为了给予其相应决策选择的权重，将拆解的半成品的数量*对应组成的零配件的成本之和*拆解出的半成品合格率的平均值作为隐性拆解收益进行权重的增加，在利润比较完成得到最优决策后将该部分收益减去。

(8) 将各个收益之和减去各个成本之和得到利润，取负值作为优化目标

建立完利润函数后应用 Global Optimization Toolbox 中的 ga 函数（遗传算法）求解得到最优决策及对应的最大利润值。

经过第一轮决策后，选择拆解的方案会剩余一部分回收零部件，将对于这部分回收零部件重复上述操作，区别在于：

1、零配件和半成品数量变化：零件数量等于拆解的不合格品数量，半成品数量等于剩余的半成品数量+拆解的不合格品数量。

2、零配件和半成品次品率变化：由于这一批零配件是由不合格半成品拆解而来，如果上一轮决策没有检测零配件，则零配件的次品率将上升，此时根据贝叶斯公式可以计算已知半成品是不合格品的条件下零配件是次品的概率，拆解的次品率变为零件原次品概率与成品不合格品概率的比值，综合原先剩余的零件可定义总次品率为两种来源的零件次品率根据数量加权方法获得。

3、引入终止条件：对于不断产生回收零件和半成品的决策方案，考虑到企业盈利的目的，设置只有当这部分零部件和半成品的利润为正数且未来的拆解收益大于一定值的情况下继续加工，否则终止加工。对于终止加工后仍剩余的零件和半成品作丢弃处理，这部分零件和半成品仅花费它们的相应成本但没有收益。

修改以上区别后，更新输入为新的零配件数量、零配件次品率、半成品数量、半成品次品率，代入利润函数。引入终止条件，利用循环嵌套 ga 遗传算法函数的方法求解拆解回收的零部件和半成品的最优处理办法，得到最佳决策方案和最大利润。

将一轮的每个最佳方案、所得利润与后续回收零件后的轮次中的最佳处理办法相结合就得到最大总利润的决策方案以及最大利润，考虑企业的盈利性质将推荐这个决策方案为最优方案。

问题 3 最高利润为 52490 元，最优决策方案为：

以一行为一轮的决策变量，决策变量含义依次是：

$S_{11} - S_{18}$ 表示零配件是否检验， $S_{21} - S_{23}$ 表示半成品是否检验， $S_{31} - S_{33}$ 表示不合格半成品是否拆解， S_4 表示成品是否检测， S_5 表示不合格成品是否拆解。

S_{11}	S_{12}	S_{13}	S_{14}	S_{15}	S_{16}	S_{17}	S_{18}	S_{21}	S_{22}	S_{23}	S_{31}	S_{32}	S_{33}	S_4	S_5
1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1
0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1

5.3.3 结果分析

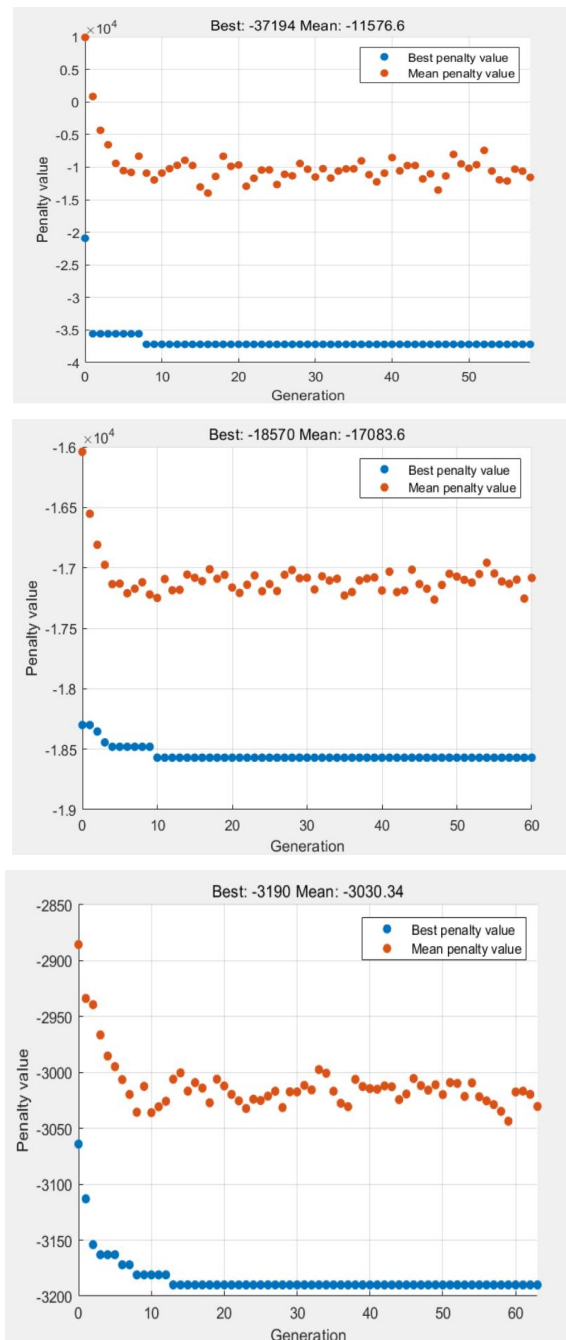


图 3 ga 算法中最优决策和最大利润值的收敛速度

图 3 展示了遗传算法三次迭代中最优决策和最大利润值的收敛速度，其中横坐标为迭代次数，纵坐标为负数，因此纵坐标越小，利润值越高。蓝色表示最优决策的利润值变化情况，红色表示其他决策利润值的平均水平，可以明显看出最优决策的利润值比较稳定且始终远高于平均水平。

同时，实验中对生产成品总数和正品数进行计数，得到最优方案共生产 996 件产品，其中共有 897 件正品，市场次品率仅 0.0994%。这个决策在提高利润值的同时最大化零配件的使用率，同时选择一些检测，在尽量减少成本的情况下降低了市场次品率，总体表现良好。

5.4 问题四模型建立与求解

5.4.1 模型建立

由题目假设知，零配件、半成品、成品的次品率未知，需根据实际采样结果确定次品率，并重新设计问题二和问题三的决策方案。求解思路如下：

（1）获得样品次品率

实际生产过程中，通常已知零配件厂家给出的零配件次品率标称值，但实际样本次品率会因零件批次等原因产生波动，同时装配机器在出厂时会有初始设置，但随机器使用年限增加，机器老化等原因可能使装配机器产生的次品率有波动，因此需对样品进行抽样获得实际次品率。

假设为待检测的样品编号，为减少因机器故障等原因导致一连串编号是次品等事件影响，同时考虑抽样检测成本，决定每次抽样数量为第一问中计算得到的最小样本数量，同时用简单随机抽样多次抽样，抽样次数与目标利润函数有关但需不少于 3 次。实验方法为生成最小样本数量 n 个随机数的办法模拟真实采样过程，设置小于或等于某数值代表次品出现，该数值出现的理论概率应与总体次品率相等（本题中认为题目二和题目三提供的次品率为总体次品），然后计数次品出现的次数 $ccount$ 即可得到一个样本次品率 p_{s1} ：

$$p_s = \frac{ccount}{n} \quad (70)$$

多次抽样可以得到多个样本次品率 $p_{sk} (k = 2, 3 \dots K)$ ， K 表示抽样次数，对所有 p_{sk} 取平均值得到样本次品率：

$$p_s = \frac{\sum_{k=1}^K p_{sk}}{n} \quad (71)$$

（2）将利润函数的参数更新为抽样检测所得零配件、半成品、成品的次品率，补充抽样检测所用花销，用问题二和问题三的模式求解即可。

5.4.2 模型求解

输入：最小样本数量（由第一问中最小样本公式得到结果）

输出：零配件、半成品、成品的次品率

求解思路：

（1）random 函数模拟真实抽样过程。以总体次品率为 10% 为例，用 random 函数随机生成 n 个 $[0, 10]$ 以内的整数随机数，并计数其中数字 1 的个数，因为理论上数字 1 在整数范围 0-10 内出现的概率是 10%，等于总体次品率。此时可以计算出
样本次品频率 = $\frac{\text{数字 1 出现的次数}}{\text{最小样本数量}}$ ，然后综合考虑检测成本和实验准确性，重复进行该实验 3 次，获得三组样本次品频率，将三组频率取平均得到一个平均频率。

由于事件某样本是次品的结果不影响其他实验结果，即实验是独立的，并且 3 次实验的条件是相同的，这里将次品频率等同于次品概率，即完成通过抽样检测方法测得次品率的步骤。对所有情况中涉及到的零件、半成品和成品，都假设题中给出的次品率是总体次品率，则可以采用相同的抽样检测方法计算次品率。

(2) 将抽样检测方法得到的次品率作为新的参数值代入问二、问题三的求解模型中获得最大利润和最优方案。

表 8 题目二中最小采样数量及抽样检测所得次品率

	最小采样数量			次品率		
	零配件 1	零配件 2	成品	零配件 1	零配件 2	成品
情况一	97	97	97	0.1134	0.0721	0.1030
情况二	173	17	173	0.1791	0.1724	0.2011
情况三	97	97	97	0.1237	0.0721	0.1443
情况四	173	173	173	0.1851	0.1896	0.2241
情况五	97	173	97	0.0824	0.1445	0.0618
情况六	51	51	51	0.0192	0.0576	0.0384

表 9 初始零件数量为 1000 的六种情况推荐决策方案结果（抽样检测方法）

情况	最大利润	推荐决策方案	合格品数量	市场次品率	理论 市场次品率
1	13793	$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$	853	20.76%	27.1%
2	4403	$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$	652	25.30%	48.8%
3	10313	$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$	829	0.00%	27.1%
4	11862	$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$	810	0.00%	48.8%
5	13479	$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$	851	0.00%	35.2%
6	20674	$[0 \ 0 \ 0 \ 0]$	889	11.00%	14.3%

表 10 题目三中最小采样数量及抽样检测所得次品率

	零配件 1	零配件 2	零配件 3	零配件 4	零配件 5	零配件 6
采样数	97	97	97	97	97	97
次品率	0.0824	0.1030	0.1443	0.1237	0.1237	0.0927
	零配件 7	零配件 8	半成品 1	半成品 2	半成品 3	成品
采样数	97	97	97	97	97	97
次品率	0.0618	0.1134	0.1030	0.1030	0.1122	0.0918

将问题 3 的次品率更改为抽样检测方法获得的次品率，最高利润为 55318 元，最优决策方案为：

以一行为一轮的决策变量，决策变量含义依次是：

$S_{11} - S_{18}$ 表示零配件是否检验, $S_{21} - S_{23}$ 表示半成品是否检验, $S_{31} - S_{33}$ 表示不合格半成品是否拆解, S_4 表示成品是否检测, S_5 表示不合格成品是否拆解。

S_{11}	S_{12}	S_{13}	S_{14}	S_{15}	S_{16}	S_{17}	S_{18}	S_{21}	S_{22}	S_{23}	S_{31}	S_{32}	S_{33}	S_4	S_5
1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1
0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1

5.4.3 结果分析

分析模拟抽样得到的次品率和在该概率下的重新计算的问题二和问题三结果, 并与题目数据以及之前求得的最大利润进行比较, 发现数据基本相近, 推荐的最优决策方案基本一致, 说明最小样本公式求取的抽样个数合理可行, 简单随机的模拟抽样正确, 最优决策方案具有优越性。

六、模型评价与推广

6.1 模型的优点

- (1) 模型的内部工作原理清晰, 能充分理解题中所给工厂流程, 对每一决策下零件、半成品、成品的次品率变化计算精确, 能对拆解回收的零件和半成品有较合理的处理办法和处理依据。
- (2) 模型运用关键性性能指标思想, 抓住影响企业决策问题的最重要因素为企业所得利润, 将市场次品率等指标边缘化处理, 将复杂的多性能优化问题转化为简单的最大利润问题, 合理设置参数, 模型的输出结果符合题目要求, 能解决实际问题。
- (3) 本文使用的遗传进化算法具有能找到全局最优解、适合复杂的优化问题、能适应不同优化环境等优点, 对于求解企业多种生产流程的最大化利润求解模型非常适用。

6.2 模型的不足

实际应用中, 零配件数量可能充足, 本文为量化指标方便, 为零件数量设置初始值。本文算法中对于拆解回收再利用的零件, 如果其带来的正利润过低, 我们认为这些零件质量不好, 会选择丢弃, 但不排除其中仍有合格零件的可能性, 在实际生产中要根据情况处理这批零件。

参考文献

- [1] 王建康. 二项分布双边收尾概率和假设检验统计量的几处修正[J]. 作物学报, 2024, 50(06): 1361-1372.
- [2] 彭凡, 张力为, 周财亮. 体育科学实验研究如何确定适宜的样本量[J]. 上海体育学院学报, 2023, 47(02): 26-36. DOI: 10.16099/j.sus.2022.03.16.0008.
- [3] 贾俊平, 何晓群, 金勇进. 《统计学》. 第7版, 中国人民大学出版社, 2018.

附录

程序代码

模型 1:

```
x = 6; % 成功次数
n = 59; % 试验次数
alpha = 0.1; % 显著性水平
[phat, pci] = binofit(x, n, alpha);
disp(pci); % 输出 Clopper-Pearson 置信区间
```

模型 2: 以情况 1 为例

```
% 参数赋值
C1 = 1000;
C2 = 1000;
b1 = 4;
b2 = 18;
x1 = 0.1;
x2 = 0.1;
x3 = 0.1;
m = 56;
t1 = 2;
t2 = 3;
t3 = 3;
z = 6;
j = 5;
d = 6;

% 初始化成本矩阵
costMatrix = zeros(1, 7);

% 定义列名称
columnNames1 = {'购买成本', '零件 1 检测成本', '零件 2 检测成本', '装配成本', '成品检测成本', '拆解成本', '调换成本'};

% 初始化收益矩阵
revenueMatrix = zeros(1, 3);

% 定义列名称
columnNames2 = {'零件售出收益', '拆解收益', '回收收益'};

bestnumdefect = 0;
bestnumproduct = 0;
toCz = 0;
```

```

Lp=0;

% 初始化数据存储
bestProfit = -Inf;
bestDecision = [];
num_j = 0;

% 遍历所有决策组合
for S11 = 0:1
    for S12 = 0:1
        for S2 = 0:1
            for S3 = 0:1
                % 计算零件剩余数量
                C1_prime = round(S11 * (1 - x1) * C1 + (1 - S11) * C1);
                C2_prime = round(S12 * (1 - x2) * C2 + (1 - S12) * C2);
                C3 = min(C1_prime, C2_prime);

                % 计算购买、零件检测、装配成本
                cost1 = t1 * S11 * C1 + t2 * S12 * C2 + b1 * C1 + b2 * C2 + z * C3;

                % 计算装配后零件剩余数量
                C1_dprime = C1_prime - C3;
                C2_dprime = C2_prime - C3;

                % 计算成品检测成本
                cost2 = S2 * t3 * C3;

                % 零配件合格率
                PR1 = S11 + (1 - S11) * (1 - x1);
                PR2 = S12 + (1 - S12) * (1 - x2);

                % 成品正品和次品数量
                Cz3 = round(PR1 * PR2 * (1 - x3) * C3);
                Cc3 = C3 - Cz3;

                % 收益计算
                revenue = m * Cz3 + S3 * Cc3 * (b1 + b2) + C1_dprime * b1 + C2_dprime * b2;

                % 拆解和调换成本
                disassemblyCost = S3 * Cc3 * j;
                replacementCost = d * (1 - S2) * Cc3;

                % 总利润
                profit = revenue - cost1 - cost2 - disassemblyCost - replacementCost;

```

```

% 调换次品数
numdefect = (1-S2)*Cc3;

% 生产的产品数
numproduct = C3;

% 更新最佳利润和决策
if profit > bestProfit

    bestProfit = profit;%目前最大利润

    bestDecision = [S11, S12, S2, S3];%目前最优决策

    bestDecisions = bestDecision;%记录最优决策

    LastCz3 = Cz3;%得到生产的正品数

    num_j = S3*Cc3;%得到拆解的产品数

    bestnumdefect = numdefect;%得到目前已生产次品数

    bestnumproduct = numproduct;%得到已生产产品数

    %成本统计
    costMatrix(1) = b1 * C1 + b2 * C2; % 购买成本
    costMatrix(2) = t1 * S11 * C1; % 零件 1 检测成本
    costMatrix(3) = t2 * S12 * C2; % 零件 2 检测成本
    costMatrix(4) = z * C3; % 装配成本
    costMatrix(5) = S2 * t3 * C3; % 成品检测成本
    costMatrix(6) = S3 * Cc3 * j; % 拆解成本
    costMatrix(7) = d * (1 - S2) * Cc3; % 调换成本

    %收益统计
    revenueMatrix(1) = m * Cz3; % 零件售出收益
    revenueMatrix(2) = S3 * Cc3 * (b1 + b2); % 拆解收益
    revenueMatrix(3) = C1_dprime * b1 + C2_dprime * b2; % 首次检测收益

end
end
end
end
end

toCz = LastCz3;%计算生产的所有正品数

```

```

while LastCz3>=1&&num_j~=0%若新生产的正品数大于 1 且拆解的产品数不为 0
    C1=num_j;%更新零配件数
    C2=num_j;

    % 将前两个元素转换为字符串
    decisionStr = num2str(bestDecision(1:2));
    decisionStr = strrep(decisionStr, ' ', ''); % 移除字符串中的空格

    % 使用 switch 语句更新次品率
    switch decisionStr
        case '10'
            x1=0;
            x2=x2/(x2+x3*(1-x2));
        case '01'
            x1=x1/(x1+x3*(1-x1));
            x2=0;
        case '11'
            x1=0;
            x2=0;
        case '00'
            x1 = x1/(1-(1-x1)*(1-x2)+x3*(1-x1)*(1-x2));
            x2 = x2/(1-(1-x1)*(1-x2)+x3*(1-x1)*(1-x2));
    end

    %以新的参数求得新的最优情况下的利润，决策，生产的正品数，拆解的产品数，生产的次品数和生
    产的产品数
    [fbestProfit, fbestDecision, fLastCz3, fnum_j, fbestnumdefect, fbestnumproduct, fLp, fcostMatrix,
    frevenueMatrix] = evaluateDecisions(C1, C2, b1, b2, x1, x2, x3, m, t1, t2, t3, z, j, d);

    if fbestProfit<=0
        break;
    end

    Lp=fLp;%记录最后一次拆解收益

    bestDecision=fbestDecision;%更新目前最优决策

    bestProfit = bestProfit+fbestProfit;%累计计算利润

    bestnumdefect = bestnumdefect+fbestnumdefect;%累计计算生产的次品数

    bestnumproduct = bestnumproduct+fbestnumproduct;%累计计算生产的产品数

    bestDecisions = [bestDecisions;fbestDecision];%储存决策

    LastCz3 = fLastCz3;%更新生产的正品数

```



```

toCz = toCz+LastCz3;%计算生产的所有正品数

num_j = fnum_j;%更新拆解的产品数

costMatrix = [costMatrix;fcostMatrix];

revenueMatrix=[revenueMatrix;frevenueMatrix];

end

% 设置最后一行第二列的元素为 0
revenueMatrix(end, 2) = 0;

bestProfit = bestProfit - Lp;

defectrate=bestnumdefect/bestnumproduct;%计算次品率

% 打印最佳利润和决策
fprintf('\nBest Profit: $%.2f\n', bestProfit);

% 打印最优决策矩阵
fprintf('\nBest Decisions Matrix:\n');
disp(bestDecisions);

% 打印生产次品率
fprintf('Defect Rate: %.2f%%\n', defectrate * 100);

% 打印生产的正品数
fprintf('Total Quality Products Produced: %d\n', toCz);

% 创建表格
costTable = array2table(costMatrix, 'VariableNames', columnNames1);

% 显示表格
disp(costTable);

% 创建表格
revenueTable = array2table(revenueMatrix, 'VariableNames', columnNames2);

% 显示表格
disp(revenueTable);

```

模型 3:

```

populationSize = 100; % 定义种群大小
eliteCount = ceil(0.05 * populationSize); % 计算保留的精英个体数量，假设为 5%的种群大小
%创建初始空间

```

```

bestDecisions=[];%最优决策
bestProfits=[];%利润记录
toCz3=0;%生产的总正品数
toC3=0;%生产的总产品数
lastj=0;%最后一次半成品拆解收益

% 给定的初始数据
C1i = [1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000]; % 零件的初始数量
x1i = [0.082474, 0.10309, 0.14433, 0.12371, 0.12371, 0.092784, 0.061856, 0.1134]; % 零件的次品率
Co2i = [0, 0, 0]; % 半成品的初始数量
xo2i = [0, 0, 0]; % 半成品的次品率

% 遗传算法参数设置
nvars = 16; % 决策变量的数量
LB = zeros(1, nvars); % 变量的下界
UB = ones(1, nvars); % 变量的上界
IntCon = 1:nvars; % 所有变量都是整数（0-1 决策变量）

% 使用遗传算法求解
options = optimoptions('ga', 'Display', 'iter', 'PopulationSize', populationSize, 'MaxGenerations', 100, 'PlotFcn',
@gaplotbestf, 'EliteCount', eliteCount);
[bestDecision, bestProfit] = ga(@(decision) fitnessFunctionQ3(decision, C1i, x1i, Co2i, xo2i), nvars, [], [], [], [],
LB, UB, [], IntCon, options);
bestProfit=-bestProfit;
%% 输出最优解和对应的利润
% fprintf('最优决策: %s\n', mat2str(bestDecision));
% fprintf('最大利润: %f\n', -bestProfit); % 注意利润在目标函数中取了负值

%取出最优值
[profit, Cj1i, Cj2i, revenue2, revenuep, Cz3, C3, C21_dprime, C22_dprime, C23_dprime, pj1i, p2i, pj2i] =
fitnessFunctionQ3(bestDecision, C1i, x1i, Co2i, xo2i);

%更新利润
bestProfit=bestProfit-revenuep;
bestProfits=[bestProfits;bestProfit];

%更新最优决策
bestDecisions=[bestDecisions;bestDecision];

%更新生产的总正品数，产品数和最后一次半成品拆解收益
toCz3=toCz3+Cz3;
toC3=toC3+C3;
lastj=revenue2;

%更新零配件数量，半成品数量及其次品率
C1i = Cj1i;

```

```

Co2i = Cj2i+C21_dprime+C22_dprime+C23_dprime;
x1i = pj1i;
xo2i =
[(Cj2i(1)*pj2i(1)+C21_dprime*p2i(1))/(Cj2i(1)+C21_dprime),(Cj2i(2)*pj2i(2)+C22_dprime*p2i(2))/(Cj2i(2)+C
22_dprime),(Cj2i(3)*pj2i(3)+C21_dprime*p2i(3))/(Cj2i(3)+C23_dprime)];

while -profit>0&&revenuep>500
    % 遗传算法参数设置
    nvars = 16; % 决策变量的数量
    LB = zeros(1, nvars); % 变量的下界
    UB = ones(1, nvars); % 变量的上界
    IntCon = 1:nvars; % 所有变量都是整数 (0-1 决策变量)

    % 使用遗传算法求解
    options = optimoptions('ga', 'Display', 'iter', 'PopulationSize', 100, 'MaxGenerations', 100, 'PlotFcn',
@gaplotbestf, 'EliteCount', eliteCount);
    [bestDecision, bestProfit] = ga(@(decision) fitnessFunctionQ3_p(decision, C1i, x1i, Co2i, xo2i), nvars, [],
[], [], [], LB, UB, [], IntCon, options);
    if isnan(bestProfit)
        break;
    end
    bestProfit=-bestProfit;

    %取出最优值
    [profit, Cj1i, Cj2i, revenue2, revenuep, Cz3, C3, C21_dprime, C22_dprime, C23_dprime,pj1i, p2i, pj2i] =
fitnessFunctionQ3_p(bestDecision, C1i, x1i, Co2i, xo2i);

    %更新利润
    bestProfit=bestProfit-revenuep;
    bestProfits=[bestProfits;bestProfit];

    %更新最优决策
    bestDecisions=[bestDecisions;bestDecision];

    %更新生产的总正品数，产品数和最后一次半成品拆解收益
    toCz3=toCz3+Cz3;
    toC3=toC3+C3;
    lastj=revenue2;

    %更新零配件数量，半成品数量及其次品率
    C1i = Cj1i;
    Co2i = Cj2i+C21_dprime+C22_dprime+C23_dprime;
    x1i = pj1i;
    xo2i =
[(Cj2i(1)*pj2i(1)+C21_dprime*p2i(1))/(Cj2i(1)+C21_dprime),(Cj2i(2)*pj2i(2)+C22_dprime*p2i(2))/(Cj2i(2)+C
22_dprime),(Cj2i(3)*pj2i(3)+C21_dprime*p2i(3))/(Cj2i(3)+C23_dprime)];
end

```

```

% 减去最后一次拆解收益
bestProfits(end) = bestProfits(end) - lastj;

% 计算利润总和
totalP = sum(bestProfits)-lastj;

%计算次品率
defectrate=1-(toCz3/toC3);

% 显示结果
disp('最优决策组合为: ');
disp(bestDecisions);
disp('总利润为: ');
disp(totalP);
fprintf('共生产了 %d 件产品, 其中有 %d 件正品\n', toC3, toCz3);
disp('次品率是:');
disp(defectrate);

function [profit, Cj1i, Cj2i, revenue2, revenuep, Cz3, C3, C21_dprime, C22_dprime, C23_dprime,pj1i, p2i, pj2i]
= fitnessFunctionQ3(decision, C1i, x1i, Co2i, xo2i)
S11 = decision(1);
S12 = decision(2);
S13 = decision(3);
S14 = decision(4);
S15 = decision(5);
S16 = decision(6);
S17 = decision(7);
S18 = decision(8);
S21 = decision(9);
S22 = decision(10);
S23 = decision(11);
S31 = decision(12);
S32 = decision(13);
S33 = decision(14);
S4 = decision(15);
S5 = decision(16);

% 这里插入之前的计算利润的代码
%半成品初始数量
Co21=Co2i(1);
Co22=Co2i(2);
Co23=Co2i(3);
%半成品初始次品率
xo21=xo2i(1);
xo22=xo2i(2);

```

xo23=xo2i(3);

% 参数赋值

C11=C1i(1);%零件购买数量（函数输入）

C12=C1i(2);

C13=C1i(3);

C14=C1i(4);

C15=C1i(5);

C16=C1i(6);

C17=C1i(7);

C18=C1i(8);

x11=x1i(1);%次品率（函数输入）

x12=x1i(2);

x13=x1i(3);

x14=x1i(4);

x15=x1i(5);

x16=x1i(6);

x17=x1i(7);

x18=x1i(8);

t11=1;%检测费用

t12=1;

t13=2;

t14=1;

t15=1;

t16=2;

t17=1;

t18=2;

b1=2;%购买费用

b2=8;

b3=12;

b4=2;

b5=8;

b6=12;

b7=8;

b8=12;

z11=8;%组装费用

z12=8;

z13=8;

x21=0.10309;%次品率

x22=0.10309;

x23=0.11224;

t21=4;%检测费用

t22=4;

t23=4;

j11=6;%拆解费用

```

j12=6;
j13=6;
z2=8;%组装费用
x3=0.091837;%次品率
t3=6;%检测费用
j2=10;%拆解费用
m=200;%售价
d=40;%调换费用

```

```

%决策 1: S1(i)代表第 i 个零配件是否进行检测

```

```

%进行决策 1 后, 第 i 个零配件的数量为 C1(i)_prime

```

```

C11_prime = round(S11 * (1 - x11) * C11 + (1 - S11) * C11);
C12_prime = round(S12 * (1 - x12) * C12 + (1 - S12) * C12);
C13_prime = round(S13 * (1 - x13) * C13 + (1 - S13) * C13);
C14_prime = round(S14 * (1 - x14) * C14 + (1 - S14) * C14);
C15_prime = round(S15 * (1 - x15) * C15 + (1 - S15) * C15);
C16_prime = round(S16 * (1 - x16) * C16 + (1 - S16) * C16);
C17_prime = round(S17 * (1 - x17) * C17 + (1 - S17) * C17);
C18_prime = round(S18 * (1 - x18) * C18 + (1 - S18) * C18);

```

```

%进行决策 1 后, 组装的第 i 个半成品的数量为 C2(i)

```

```

C21 = round(min([C11_prime, C12_prime, C13_prime])) + Co21;
C22 = round(min([C14_prime, C15_prime, C16_prime])) + Co22;
C23 = round(min([C17_prime, C18_prime])) + Co23;

```

```

%进行决策 1 和组装后, 计算成本 1, 有购买成本, 检测成本, 组装成本, t1(i)为第 i 个零配件的检测成本,

```

```

%bi 为第 i 个零配件的购买数量, z1(i)为第 i 个半成品的装配费用

```

```

cost1=t11 * C11 * S11+t12 * C12 * S12+t13 * C13 * S13 + t14 * C14 * S14 + t15 * C15 * S15 + t16 * C16 * S16 + t17 * C17 * S17 + t18 * C18 * S18 ...
+b1 * C11+b2 * C12+b3 * C13 + b4 * C14 + b5 * C15 + b6 * C16 + b7 * C17 + b8 * C18 ...
+z11 * round(min([C11_prime,C12_prime,C13_prime]))+z12 * round(min([C14_prime,C15_prime,C16_prime]))+z13 * round(min([C17_prime,C18_prime]));

```

```

%进行装配后, 零配件 i 剩余的数量为 C1(i)_dprime

```

```

C11_dprime=C11_prime-round(min([C11_prime, C12_prime, C13_prime]));
C12_dprime=C12_prime-round(min([C11_prime, C12_prime, C13_prime]));
C13_dprime=C13_prime-round(min([C11_prime, C12_prime, C13_prime]));
C14_dprime=C14_prime-round(min([C14_prime, C15_prime, C16_prime]));
C15_dprime=C15_prime-round(min([C14_prime, C15_prime, C16_prime]));
C16_dprime=C16_prime-round(min([C14_prime, C15_prime, C16_prime]));
C17_dprime=C17_prime-round(min([C17_prime, C18_prime]));
C18_dprime=C18_prime-round(min([C17_prime, C18_prime]));

```

```

%收益 1: 检测收益

```

```

revenue1=C11_dprime * b1 + C12_dprime * b2+ C13_dprime * b3 + C14_dprime * b4 + C15_dprime * b5 + C16_dprime * b6 + C17_dprime * b7 + C18_dprime * b8;

```



```

%半成品 i 的正品率为 p1(i),经过决策 1 和组装后正品数量为 Cz2(i), 次品数量为 Cc2(i)
%RPi 为第 i 个零件的合格率, x1(i)为第 i 个零配件的原次品率, x2(i)为第 i 个半成品的原次品率
RP1 = S11 + (1 - S11) * (1 - x11);
RP2 = S12 + (1 - S12) * (1 - x12);
RP3 = S13 + (1 - S13) * (1 - x13);
RP4 = S14 + (1 - S14) * (1 - x14);
RP5 = S15 + (1 - S15) * (1 - x15);
RP6 = S16 + (1 - S16) * (1 - x16);
RP7 = S17 + (1 - S17) * (1 - x17);
RP8 = S18 + (1 - S18) * (1 - x18);
%半成品 1
p11=RP1*RP2*RP3*(1-x21);%组装的正品率
Cz21=round(min([C11_prime, C12_prime, C13_prime])*p11+Co21*(1-xo21));
Cc21=C21-Cz21;
p21=Cc21/C21;%半成品的次品率
%半成品 2
p12=RP4*RP5*RP6*(1-x22);
Cz22=round(min([C14_prime, C15_prime, C16_prime])*p12+Co22*(1-xo22));
Cc22=C22-Cz22;
p22=Cc22/C22;
%半成品 3
p13=RP7*RP8*(1-x23);
Cz23=round(min([C17_prime, C18_prime])*p13+Co23*(1-xo23));
Cc23=C23-Cz23;
p23=Cc23/C23;

%决策 2: S2(i)代表第 i 个半成品是否检测
%进行决策 2 后半成品 i 的数量变为 C2(i)_prime
C21_prime=S21*Cz21+(1-S21)*C21;
C22_prime=S22*Cz22+(1-S22)*C22;
C23_prime=S23*Cz23+(1-S23)*C23;

%计算成本 2: 半成品的检测费用
cost2=S21*t21*C21+S22*t22*C22+S23*t23*C23;

%计算半成品的合格率
RP1_prime = S21 + (1 - S21) * (1 - p21);
RP2_prime = S22 + (1 - S22) * (1 - p22);
RP3_prime = S23 + (1 - S23) * (1 - p23);

%决策 3: S3(i)代表对检测出的第 i 个不合格半成品是否拆解
%计算成本 3: 半成品的拆解费用, j1(i)为第 i 个半成品的拆解费用
cost3=S31*j11*S21*Cc21+S32*j12*S22*Cc22+S33*j13*S23*Cc23;

```

%收益 2: 拆解收益

revenue2=S31*S21*Cc21*(b1+b2+b3)+S32*S22*Cc22*(b4+b5+b6)+S33*S23*Cc23*(b7+b8);

%计算拆解得到的零配件数量

Cj1i(1)=S31*S21*Cc21;

Cj1i(2)=S31*S21*Cc21;

Cj1i(3)=S31*S21*Cc21;

Cj1i(4)=S32*S22*Cc22;

Cj1i(5)=S32*S22*Cc22;

Cj1i(6)=S32*S22*Cc22;

Cj1i(7)=S33*S22*Cc23;

Cj1i(8)=S33*S22*Cc23;

%计算拆解出的零配件的次品率

pj1i(1)=S21*S31*(1-RP1)/p21;

pj1i(2)=S21*S31*(1-RP2)/p21;

pj1i(3)=S21*S31*(1-RP3)/p21;

pj1i(4)=S22*S32*(1-RP4)/p22;

pj1i(5)=S22*S32*(1-RP5)/p22;

pj1i(6)=S22*S32*(1-RP6)/p22;

pj1i(7)=S23*S33*(1-RP7)/p23;

pj1i(8)=S23*S33*(1-RP8)/p23;

%进行组装, 计算成品数量

C3=min([C21_prime,C22_prime,C23_prime]);

%计算组装之后半成品的剩余数量及其次品率

C21_dprime=C21_prime-C3;

C22_dprime=C22_prime-C3;

C23_dprime=C23_prime-C3;

p2i=[1-RP1_prime,1-RP2_prime,1-RP3_prime];

%成本 4: 成品的组装成本,z2 为成品的组装费用

cost4=z2*C3;

%计算成品的正品数量 Cz3 和次品数量 Cc3, x3 为成品的原次品率

Cz3=round(RP1_prime*RP2_prime*RP3_prime*(1-x3)*C3);

Cc3=C3-Cz3;

%决策 4: S4 表示是否对成品进行检测

%成本 5: 检测费用, t3 为成品的检测费用

cost5=S4*t3*C3;

%计算退回的产品数 Ct

Ctui=(1-S4)*Cc3;

```

%计算检测出的次品数
Cjian=S4*Cc3;

%决策 5: S5 表示对检测出的以及退回的次品是否进行拆解
%成本 6: 拆解费用,j2 代表成品的拆解费用
cost6=S5*j2*Cc3;

%收益 3: 售卖收入,m 为成品的售价
revenue3=m*Cz3;

%成本 7: 调换损失,d 为调换费用
cost7=(1-S4)*Cc3*d;

%计算拆解出的半成品的数量
Cj2i(1)=S5*Cc3;
Cj2i(2)=S5*Cc3;
Cj2i(3)=S5*Cc3;

%计算拆解出的半成品的次品率
pj2i=[0,0,0];
pj2i(1)=S5*(1-RP1_prime)/(Cc3/C3);
pj2i(2)=S5*(1-RP2_prime)/(Cc3/C3);
pj2i(3)=S5*(1-RP3_prime)/(Cc3/C3);

%计算隐性拆解收益
revenuep=S5*Cc3*(b1+b2+b3+b4+b5+b6+b7+b8)*((1-pj2i(1)+1-pj2i(2)+1-pj2i(3))/3);

%计算利润
profit=-(revenue1+revenue2+revenue3+revenuep-cost1-cost2-cost3-cost4-cost5-cost6-cost7);% 使用负值因为
ga 默认最小化
end

-----

function [profit, Cj1i, Cj2i, revenue2, revenuep, Cz3, C3, C21_dprime, C22_dprime, C23_dprime,pj1i, p2i, pj2i]
= fitnessFunctionQ3_p(decision, C1i, x1i, Co2i, xo2i)
S11 = decision(1);
S12 = decision(2);
S13 = decision(3);
S14 = decision(4);
S15 = decision(5);
S16 = decision(6);
S17 = decision(7);
S18 = decision(8);
S21 = decision(9);
S22 = decision(10);
S23 = decision(11);

```

```

S31 = decision(12);
S32 = decision(13);
S33 = decision(14);
S4 = decision(15);
S5 = decision(16);

```

```

% 这里插入之前的计算利润的代码

```

```

%半成品初始数量

```

```

Co21=Co2i(1);

```

```

Co22=Co2i(2);

```

```

Co23=Co2i(3);

```

```

%半成品初始次品率

```

```

xo21=xo2i(1);

```

```

xo22=xo2i(2);

```

```

xo23=xo2i(3);

```

```

% 参数赋值

```

```

C11=C1i(1);%零件购买数量（函数输入）

```

```

C12=C1i(2);

```

```

C13=C1i(3);

```

```

C14=C1i(4);

```

```

C15=C1i(5);

```

```

C16=C1i(6);

```

```

C17=C1i(7);

```

```

C18=C1i(8);

```

```

x11=x1i(1);%次品率（函数输入）

```

```

x12=x1i(2);

```

```

x13=x1i(3);

```

```

x14=x1i(4);

```

```

x15=x1i(5);

```

```

x16=x1i(6);

```

```

x17=x1i(7);

```

```

x18=x1i(8);

```

```

t11=1;%检测费用

```

```

t12=1;

```

```

t13=2;

```

```

t14=1;

```

```

t15=1;

```

```

t16=2;

```

```

t17=1;

```

```

t18=2;

```

```

b1=2;%购买费用

```

```

b2=8;

```

```

b3=12;

```

```

b4=2;
b5=8;
b6=12;
b7=8;
b8=12;
z11=8;%组装费用
z12=8;
z13=8;
x21=0.10309;%次品率
x22=0.10309;
x23=0.11224;
t21=4;%检测费用
t22=4;
t23=4;
j11=6;%拆解费用
j12=6;
j13=6;
z2=8;%组装费用
x3=0.091837;%次品率
t3=6;%检测费用
j2=10;%拆解费用
m=200;%售价
d=40;%调换费用

```

%决策 1: S1(i)代表第 i 个零配件是否进行检测

%进行决策 1 后, 第 i 个零配件的数量为 C1(i)_prime

```

C11_prime = round(S11 * (1 - x11) * C11 + (1 - S11) * C11);
C12_prime = round(S12 * (1 - x12) * C12 + (1 - S12) * C12);
C13_prime = round(S13 * (1 - x13) * C13 + (1 - S13) * C13);
C14_prime = round(S14 * (1 - x14) * C14 + (1 - S14) * C14);
C15_prime = round(S15 * (1 - x15) * C15 + (1 - S15) * C15);
C16_prime = round(S16 * (1 - x16) * C16 + (1 - S16) * C16);
C17_prime = round(S17 * (1 - x17) * C17 + (1 - S17) * C17);
C18_prime = round(S18 * (1 - x18) * C18 + (1 - S18) * C18);

```

%进行决策 1 后, 组装的第 i 个半成品的数量为 C2(i)

```

C21 = round(min([C11_prime, C12_prime, C13_prime])) + Co21;
C22 = round(min([C14_prime, C15_prime, C16_prime])) + Co22;
C23 = round(min([C17_prime, C18_prime])) + Co23;

```

%进行决策 1 和组装后, 计算成本 1, 有购买成本, 检测成本, 组装成本, t1(i)为第 i 个零配件的检测成本,

%bi 为第 i 个零配件的购买数量, z1(i)为第 i 个半成品的装配费用

```

cost1=t11 * C11 * S11+t12 * C12 * S12+t13 * C13 * S13 + t14 * C14 * S14 + t15 * C15 * S15 + t16 * C16 *
S16 + t17 * C17 * S17 + t18 * C18 * S18 ...
+b1 * C11+b2 * C12+b3 * C13 + b4 * C14 + b5 * C15 + b6 * C16 + b7 * C17 + b8 * C18 ...

```

```
+z11 * round(min([C11_prime,C12_prime,C13_prime]))+z12 *
round(min([C14_prime,C15_prime,C16_prime]))+z13 * round(min([C17_prime,C18_prime]));
```

```
%进行装配后，零配件 i 剩余的数量为 C1(i)_dprime
C11_dprime=C11_prime-round(min([C11_prime, C12_prime, C13_prime]));
C12_dprime=C12_prime-round(min([C11_prime, C12_prime, C13_prime]));
C13_dprime=C13_prime-round(min([C11_prime, C12_prime, C13_prime]));
C14_dprime=C14_prime-round(min([C14_prime, C15_prime, C16_prime]));
C15_dprime=C15_prime-round(min([C14_prime, C15_prime, C16_prime]));
C16_dprime=C16_prime-round(min([C14_prime, C15_prime, C16_prime]));
C17_dprime=C17_prime-round(min([C17_prime, C18_prime]));
C18_dprime=C18_prime-round(min([C17_prime, C18_prime]));
```

```
%收益 1：检测收益
revenue1=0;
```

```
%半成品 i 的正品率为 p1(i),经过决策 1 和组装后正品数量为 Cz2(i)，次品数量为 Cc2(i)
%RPi 为第 i 个零件的合格率，x1(i)为第 i 个零配件的原次品率，x2(i)为第 i 个半成品的原次品率
```

```
RP1 = S11 + (1 - S11) * (1 - x11);
RP2 = S12 + (1 - S12) * (1 - x12);
RP3 = S13 + (1 - S13) * (1 - x13);
RP4 = S14 + (1 - S14) * (1 - x14);
RP5 = S15 + (1 - S15) * (1 - x15);
RP6 = S16 + (1 - S16) * (1 - x16);
RP7 = S17 + (1 - S17) * (1 - x17);
RP8 = S18 + (1 - S18) * (1 - x18);
```

```
%半成品 1
```

```
p11=RP1*RP2*RP3*(1-x21);%组装的正品率
Cz21=round(min([C11_prime, C12_prime, C13_prime])*p11+Co21*(1-xo21));
Cc21=C21-Cz21;
p21=Cc21/C21;%半成品的次品率
```

```
%半成品 2
```

```
p12=RP4*RP5*RP6*(1-x22);
Cz22=round(min([C14_prime, C15_prime, C16_prime])*p12+Co22*(1-xo22));
Cc22=C22-Cz22;
p22=Cc22/C22;
```

```
%半成品 3
```

```
p13=RP7*RP8*(1-x23);
Cz23=round(min([C17_prime, C18_prime])*p13+Co23*(1-xo23));
Cc23=C23-Cz23;
p23=Cc23/C23;
```

```
%决策 2：S2(i)代表第 i 个半成品是否检测
```

```
%进行决策 2 后半成品 i 的数量变为 C2(i)_prime
```

```
C21_prime=S21*Cz21+(1-S21)*C21;
C22_prime=S22*Cz22+(1-S22)*C22;
C23_prime=S23*Cz23+(1-S23)*C23;
```

%计算成本 2： 半成品的检测费用

$cost2 = S21 * t21 * C21 + S22 * t22 * C22 + S23 * t23 * C23;$

%计算半成品的合格率

$RP1_prime = S21 + (1 - S21) * (1 - p21);$

$RP2_prime = S22 + (1 - S22) * (1 - p22);$

$RP3_prime = S23 + (1 - S23) * (1 - p23);$

%决策 3： S3(i)代表对检测出的第 i 个不合格半成品是否拆解

%计算成本 3： 半成品的拆解费用， j1(i)为第 i 个半成品的拆解费用

$cost3 = S31 * j11 * S21 * Cc21 + S32 * j12 * S22 * Cc22 + S33 * j13 * S23 * Cc23;$

%收益 2： 拆解收益

$revenue2 = S31 * S21 * Cc21 * (b1 + b2 + b3) + S32 * S22 * Cc22 * (b4 + b5 + b6) + S33 * S23 * Cc23 * (b7 + b8);$

%计算拆解得到的零配件数量

$Cj1i(1) = S31 * S21 * Cc21;$

$Cj1i(2) = S31 * S21 * Cc21;$

$Cj1i(3) = S31 * S21 * Cc21;$

$Cj1i(4) = S32 * S22 * Cc22;$

$Cj1i(5) = S32 * S22 * Cc22;$

$Cj1i(6) = S32 * S22 * Cc22;$

$Cj1i(7) = S33 * S22 * Cc23;$

$Cj1i(8) = S33 * S22 * Cc23;$

%计算拆解出的零配件的次品率

$pj1i(1) = S21 * S31 * (1 - RP1) / p21;$

$pj1i(2) = S21 * S31 * (1 - RP2) / p21;$

$pj1i(3) = S21 * S31 * (1 - RP3) / p21;$

$pj1i(4) = S22 * S32 * (1 - RP4) / p22;$

$pj1i(5) = S22 * S32 * (1 - RP5) / p22;$

$pj1i(6) = S22 * S32 * (1 - RP6) / p22;$

$pj1i(7) = S23 * S33 * (1 - RP7) / p23;$

$pj1i(8) = S23 * S33 * (1 - RP8) / p23;$

%进行组装， 计算成品数量

$C3 = \min([C21_prime, C22_prime, C23_prime]);$

%计算组装之后半成品的剩余数量及其次品率

$C21_dprime = C21_prime - C3;$

$C22_dprime = C22_prime - C3;$

$C23_dprime = C23_prime - C3;$

$p2i = [1 - RP1_prime, 1 - RP2_prime, 1 - RP3_prime];$

%成本 4： 成品的组装成本,z2 为成品的组装费用


```

cost4=z2*C3;

%计算成品的正品数量 Cz3 和次品数量 Cc3， x3 为成品的原次品率
Cz3=round(RP1_prime*RP2_prime*RP3_prime*(1-x3)*C3);
Cc3=C3-Cz3;

%决策 4: S4 表示是否对成品进行检测

%成本 5: 检测费用, t3 为成品的检测费用
cost5=S4*t3*C3;

%计算退回的产品数 Ct
Ctui=(1-S4)*Cc3;

%计算检测出的次品数
Cjian=S4*Cc3;

%决策 5: S5 表示对检测出的以及退回的次品是否进行拆解
%成本 6: 拆解费用,j2 代表成品的拆解费用
cost6=S5*j2*Cc3;

%收益 3: 售卖收入,m 为成品的售价
revenue3=m*Cz3;

%成本 7: 调换损失,d 为调换费用
cost7=(1-S4)*Cc3*d;

%计算拆解出的半成品的数量
Cj2i(1)=S5*Cc3;
Cj2i(2)=S5*Cc3;
Cj2i(3)=S5*Cc3;

%计算拆解出的半成品的次品率
pj2i=[0,0,0];
pj2i(1)=S5*(1-RP1_prime)/(Cc3/C3);
pj2i(2)=S5*(1-RP2_prime)/(Cc3/C3);
pj2i(3)=S5*(1-RP3_prime)/(Cc3/C3);

%计算隐性拆解收益
revenuep=S5*Cc3*(b1+b2+b3+b4+b5+b6+b7+b8)*((1-pj2i(1)+1-pj2i(2)+1-pj2i(3))/3);

%计算利润
profit=-(revenue1+revenue2+revenue3+revenuep-cost1-cost2-cost3-cost4-cost5-cost6-cost7);% 使用负值因为
ga 默认最小化
end

```

模型 4：对问题 2，3 的求解使用模型 2，3

```
% 请求用户输入 n 的值
n = input('请输入整数 n: ');

% 生成 n 个 1 到 10 之间的随机整数
randomNumbers = randi([1, 10], 1, n);

% 计算生成的随机数中 1 的个数
numOnes = sum(randomNumbers == 1);

% 计算 1 的个数与 n 的比值
ratio = numOnes / n;

% 显示结果
disp(['生成的随机数中 1 的个数: ' num2str(numOnes)]);
disp(['1 的个数与 n 的比值: ' num2str(ratio)]);
```