

基于生产过程的决策问题模型研究

摘要

在当今竞争激烈的市场环境中，企业为了尽可能地满足消费者需求并获得最佳利润，需要对生产过程中的决策进行分析，采用更为精确、高效的方法进行生产，使得利润最大化。一家电子产品生产企业，需要从供应商处购买两种零配件（零配件 1 和零配件 2）并装配成成品。成品的合格性取决于两个零配件的质量，企业需要对不合格成品进行报废或拆解处理。

研究内容包括四个主要问题：

问题 1 旨在设计一个高效的抽样检测方案，判断供应商提供的零配件次品率是否超过标称值。为采购质量较好的零件以降低产品次品率、控制生产成本，企业对供应商提供的零件进行抽样检测方案应在保证检测准确率的条件下减少检测次数。本文将分别针对 95%置信水平下拒收次品率超过标称值的零配件和 90%置信水平下接收次品率不超过标称值的零配件两种情况，给出具体方案。

问题 2 要求为企业生产过程中的各个阶段做出最优决策，实现在购买零件数量一定的情况下减少不必要检测、拆解、再装配费用支出、提高成品生产数量质量、扩大最终利润额的目标。生产过程中各阶段需要做出的决策包括是否对零配件/成品进行检测，以及不合格产品的处理方式。本研究将综合零配件与成品的次品率、购买单价、检测与装配成本、市场售价、调换损失和拆解费用等等，对最终生产成本和利润的影响做出各生产环节的决策。通过对表 1 中不同情况的分析，研究提出了具体的决策方案、依据及指标结果。

问题 3 在问题 2 的基础上，考虑了普适性更强的生产流程，涉及多个零配件和多道工序。本研究将建立数学模型，为 m 道工序、 n 个零配件的生产过程提供方案，并针对题目要求给出具体分析。

问题 4 要求结合前三问的结论计算该企业根据其检测结果得到的合理的决策方案。本问题中将假设问题 2 和问题 3 中的次品率通过抽样检测方法得到，以该次品率为准重新完成问题 2 和问题 3 的决策分析。

本研究为企业提供了全面的决策支持，旨在优化生产流程，降低成本，提高产品质量，同时考虑了检测成本与潜在损失之间的平衡。研究成果对于提高生产效率和控制成本具有参考意义。

本研究为企业提供了全面的决策支持，旨在优化生产流程，降低成本，提高产品质量，同时考虑了检测成本与潜在损失之间的平衡。研究成果对于实际生产过程具有指导意义，特别是在提高生产效率和降低质量控制成本方面。

关键词 序贯抽样；生产决策

一、问题重述

1.1 问题背景

随着科技的飞速发展，人们对电子产品的需求不断增长，企业对其生产效率和产品质量的要求也日益提升。对于企业来说，产品零配件的质量直接影响到成品的品质。企业需要在生产的各个阶段做出合理的决策，维护企业的市场竞争力。因此，生产过程中的质量控制显得尤为重要。

1.2 问题提出

根据题目中给出的限制条件和相关数据，建立数学模型，研究解决以下问题。

(1) 分析分别在 95% 和 90% 的信度下的次品率，在检测次数尽可能少的前提下，通过抽样检测的方法决定是否接收供应商提供的这批零配件。

(2) 分析两种零配件及其成品次品率，根据表 1 中的情形，分别在零配件是否进行检测、装配成品是否进行检测、检测出的不合格成品是否进行拆解等情况下给出具体的决策方案、相关依据及指标结果，为企业生产过程的各个阶段作出决策。

(3) 分析在 m 道工序、 n 个零配件的情况下，企业在生产中遇到的情况，重复上述问题 (2)，并给出具体决策方案、依据及相应指标。

(4) 分析通过抽样检测的方法得到次品率的情况，重新给出不同情况下的决策方案、相关依据及指标结果。

二、问题分析

问题以电子产品生产过程中使用的零配件、半成品及成品为研究对象，以产品生产及装配过程中各个产品的次品率、采购成本、检测成本、装配成本、市场售价、调换损失和拆解费用等多项数据为研究内容。本文认为难点在于需要对产品生产过程的各个阶段进行决策，通过对相关数据的分析和模型建立，根据不同生产阶段的需求，运用适当的数学模型和统计方法，制定出最优的生产决策方案。

2.1 问题 1

问题 1 设计合理的抽样检测方案，分析分别在 95% 和 90% 的信度下的次品率，在检测次数尽可能少的前提下，通过抽样检测的方法决定是否接收供应商提供的这批零配件。首先，我们需要对题目中所给条件进行分析，明确流程；其次，根据企业的具体情况，依据企业购买的零件个数，考虑几种常用的抽样检测方法；最后，对比分析选出的抽样检测方法，选择最合适的方法并进行计算，选出最优方案。

2.2 问题 2

问题 2 分析两种零配件及其成品次品率，根据表 1 中的情形，分别在生产过程中的各个决策环节给出具体的决策方案、相关依据及指标结果。这要求企业在生产过程中综合考虑每个环节的风险与收益，进而得出针对不同阶段的最优方案。零配件的次品率直接影响成品质量，是否检测将决定最终产品的合格率；检测成品可以减少不合格品的调换损失，但同时也增加了成本；如何处理不合格成品也会影响总成本。综上，企业需要综合考虑产品质量和利润，决定生产过程中的最佳处理方案。

2.3 问题 3

问题 3 需要分析在多道工序、多个零配件的情况下的生产决策，不仅需要针对每个零配件的次品率做出判断，还需要考虑半成品的次品率以及每道工序的装配成本和检测成本。本文构建了一个更加普遍的数学模型，对整个生产流程进行优化，在保证质量的前提下，尽可能降低成本，提高生产效率。

2.4 问题 4

问题 4 在问题 2 和问题 3 的基础上，进一步分析通过抽样检测的方法得到次品率的情况，重新解决上述问题 2 和问题 3。这要求我们重新评估次品率的估计值，以及它们对决策方案的影响，同时考虑抽样误差对最终决策结果的影响。

三、模型假设

- (1) 假设装配过程中没有对零配件造成损坏
- (2) 假设调换前的都是不合格成品，调换后的成品都是合格产品
- (3) 假设进货零配件数量足够多
- (4) 假设所有不合格成品都会被退回
- (5) 假设购买的所有零配件个数相等

四、符号说明

序号	符号	说明
1	n_{10}, n_{20}	初始购买的零件数量
2	nd	成品次品数量
3	nfp, nfp_0	最终成品数量，处理后成品数量
4	nd_1, nd_2	处理后的合格零件

5	fd_1	零配件 1 的不合格率(fraction defective)
6	fd_2	零配件 2 的不合格率
7	fdF	成品的不合格率(fdFinished)
8	fdd	再处理成品次品率
9	ic_1	零配件 1 的检测成本(inspection cost)
10	ic_2	零配件 2 的检测成本
11	icF	成品的检测成本(icFinished)
12	rl	客户退货损失(Replacement loss)
13	dc	拆解成本(Dismantling cost)
14	c_1, c_2	零件 1、零件 2 的购买单价
15	cF	成品的装配成本
16	dis	拆解成品(disassemble)
17	T	最终成本
18	p, P	售价单价，最终利润
19	t	不合格成品的拆解检验装配费用

五、模型的建立与求解

5.1 第一问的模型建立与求解

问题 1 需要分析分别在 95%和 90%的信度下的次品率，设计合理的抽样检测方案。

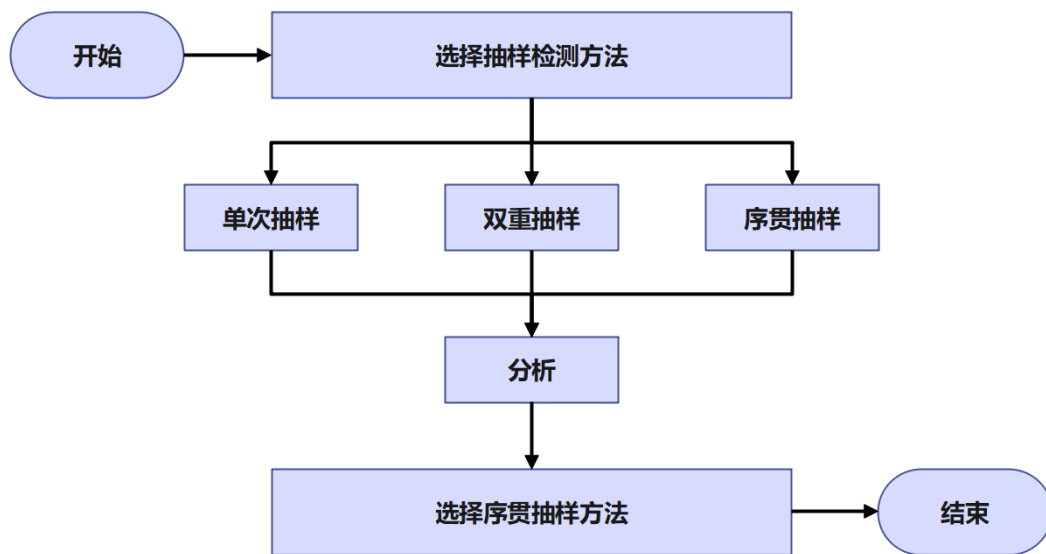


图 1 第一问流程图

5.1.1 第一问题目分析

首先对题目中所给条件进行分析。企业需要采用抽样检测方法对零配件进行检测，我们由此根据企业的具体情况，依据企业购买的零件个数，考虑几种常用的抽样检测方法，如下表所示：

表格 1 抽样检测方法及其优点分析

抽样检测方法	优点
单次抽样	抽样过程只进行一次，计算方便，适用于大多数情况。
双重抽样	可以减少在大多数情况下的抽样次数，增加检测效率，同时保持较高的信度。
序贯抽样	更加灵活，可以在样本较少时做出决策，减少平均抽样数量。

5.1.2 单次抽样

如果选用单次抽样的方法，我们需要一次抽取固定数量的样本，通过检测样本中的次品数来判断零配件的次品率。假设企业从供应商的零配件中抽取 n 个样本，如果次品数超过某标称值 c ，则拒收这批零配件，否则接收。

(1)在 95%的信度下认定零配件次品率超过标称值（10%）

我们使用二项分布来设计抽样方案。零配件次品率为 $p = 0.1$ ，即次品率上限为10%。为了在95%的信度下拒收，计算次品数 x 超过标称值 c 的概率：

$$P(X > c) \leq 0.05$$

(2)在 90%的信度下认定零配件次品率不超过标称值（10%）

同理，我们在 90%信度下认定次品率低于 10%时接收零配件。次品数 x 超过标称值 c 的概率为：

$$P(X \leq c) \geq 0.9$$

5.1.3 双重抽样

如果选用双重抽样的方法，我们需要分两步进行。首先，抽取一批样本 n_1 ，并检测其次品数。如果次品数 x_1 很小（小于某个值 c_1 ），则直接接收这批零配件；如果次品数 x_1 很大（大于某个值 c_2 ），则直接拒收；如果次品数 x_1 介于两者之间，则抽取第二批样本 n_2 ，并基于两次抽样的总次品数来决定是否接收。即：

如果 $x_1 \leq c_1$ 则接收；

如果 $x_1 > c_2$ 则拒收；

如果 $c_1 < x_1 \leq c_2$ ，则进行第二次抽样。

如果进入第二次抽样，则需要抽取第二批样本 n_2 。在第二次抽样后，统计两次抽样的总次品数 $x = x_1 + x_2$ 。根据总次品数 x 来决定最终的接收或拒收标准。即：

如果 $x \leq c_3$ ，则接收这批零配件；

如果 $x > c_3$ ，则拒收这批零配件。

其中， c_3 是根据抽样计划确定的总次品数的上限标准。

5.1.4 序贯抽样

如果选用序贯抽样的方法，我们需要对样本进行逐个检测，每次检测后根据当前的次品数来决定是否继续抽样、接收或拒收。我们设定关于次品率的假设检验：

原假设 H_0 ：次品率 $p \leq p_0$

备择假设 H_1 ：次品率 $p > p_0$

其中， p_0 为供应商提供的次品率， p_1 为企业怀疑成品可能高于的次品率。

接下来，我们根据 SPRT 推导出接收界限 A 和拒收界限 B。二者公式如下：

$$\text{接收界限A: } A = \log\left(\frac{\beta}{1-\alpha}\right)$$

$$\text{拒收界限B: } B = \log\left(\frac{1-\beta}{\alpha}\right)$$

其中， α 为企业拒收合格批次的概率（第一类错误）， β 为企业接收不合格批次的概率（第二类错误）。

下面，我们根据抽样过程中累积检测到的次品数 x ，计算每次检测的累积似然比 $L(x)$ ，来判断接受 H_0 还是 H_1 。对于每个抽样结果，累计似然比的对数形式为：

$$\log L(x) = \log L(x - 1) + \log \left(\frac{P(\text{次品}|p_1)}{P(\text{次品}|p_0)} \right)$$

最后，我们根据累积的似然比与接收界限 A 和拒收界限 B 的大小关系，决定是否接收、拒收或继续抽样。

5.2 第二问的模型建立与求解

问题 2 需要根据企业在生产中遇到的 6 种不同情况（表 1），为企业生产过程中各阶段作出决策，并给出决策方案、依据及相应的指标结果，在尽量满足生产质量的前提下，使得企业获得更高的利润。

5.2.1 生产各阶段流程介绍与分析

首先，我们需要选择是否需要检测零配件 1 和 2。若进行检测，则丢弃次品并将合格品投入生产；否则，零配件将直接投入生产。进行完零配件检测的工序之后，我们对其进行装配，进而获得成品。对于成品，我们需要选择是否进行检测。若得到不合格成品，如果前期未进行成品检测，则需支付调换费用。对于检测出和调换后的不合格成品，我们需要选择是否进行拆解。若进行拆解，则对拆解后未经检验的零配件进行重新检验，再次投入生产；否则直接将其丢弃。

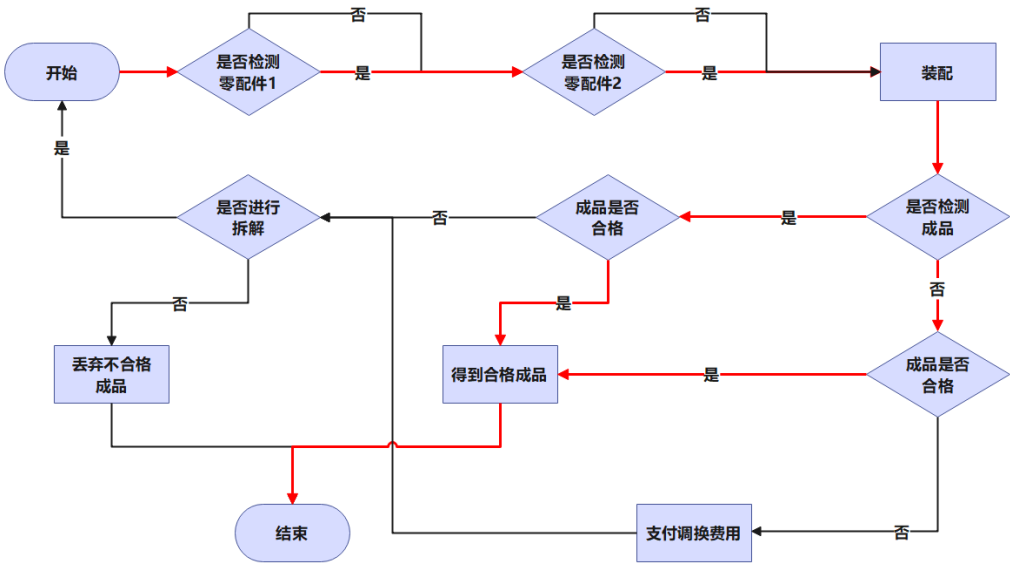


图 2 第二问流程图

根据题目所给表 1 可知，在本题背景下，企业在生产中遇到的情况一共可分为六种。在任意一种情况下，企业的生产过程都需要在保证产品质量的同时降低成本、实现利润最大化。由此，本文通过以下三个步骤来进行选择：

1. 根据零配件的次品率和检测成本选择是否检验零配件。
2. 根据成品的次品率和检测成本选择是否检验成品。
3. 对比不合格成品的调换损失和检验费用，选择是否拆解不合格成品。

下面通过上述三个步骤分别进行分析，计算不同方案的利润，对比分析选出最优方案。

5.2.2 对于是否检验零配件的分析

第一步，我们需要选择是否对两种零配件(零配件 1 和零配件 2)进行检测。如果对某种零配件不检测，这种零配件将会直接投入生产，进入到装配环节；否则，检测出的零配件次品将会被丢弃。如果零配件 1 或零配件 2 其一是次品，则成品必为次品；如果零配件 1 和零配件 2 都不是次品，成品的次品率为 10%（装配时产生的固有次品率）。

对于零配件 1 的检测公式如下：

$$Ic_1 = x_1 * n_{10} * ic_1$$

$$n_{11} = n_{10} * (1 - x_1 * fd_1)$$

其中， $x_1 = 1$ 时表示检测零配件 1， $x_1 = 0$ 时表示不检测零配件 1； n_{11} 为投入生产环节的零配件 1 的个数。

对于零配件 2 的检测公式如下：

$$Ic_2 = x_2 * n_{20} * ic_2$$

$$n_{21} = n_{20} * (1 - x_2 * fd_2)$$

其中， $x_2 = 1$ 时表示检测零配件 2， $x_2 = 0$ 时表示不检测零配件 2； n_{21} 为投入生产环节的零配件 2 的个数。

最终的成品数量为：

$$nF = \min(n_{11}, n_{21})$$

5.2.3 对于是否检验成品的分析

其次，对于生产出的成品，我们可以选择检测或者不检测。如果选择不检测并直接将其投入市场，则需为出售的次品支付一定的调换损失。

成品的检测成本为：

$$IcF = fpi * nF * icF$$

其中， $fpi = 1$ 时表示检测成品， $fpi = 0$ 时表示不检测成品。

如果不检测成品，则需支付的调换损失为：

$$RL = (1 - fpi) * nF * rl * fdF$$

最终成品中的次品数量为：

$$nd = nF * fdF$$

5.2.4 对于是否拆解不合格成品的分析

对于用户购买的不合格品，企业将无条件予以调换，并产生一定的调换损失（如物流成本、企业信誉等）。

最后，检测出的不合格成品和调换回的不合格成品，我们可以选择拆解或者不拆解。如果拆解，则需支付拆解费用，但是拆解后可以获得零配件，次品与否与装配前状态一致。拆解得到的零配件将会再次投入生产。

在拆解之前检测零配件 1、不检测零配件 2 的情况下，拆解之后零配件 2 的次品率为：

$$fdd_2 = \frac{fd_2}{(fd_2 + (1 - fd_2) * fdF)}$$

不合格成品拆解后得到的合格零配件 2 的个数为：

$$nd_2 = (1 - fdd_2) * nd$$

在拆解之前不检测零配件 1、检测零配件 2 的情况下，拆解之后零配件 1 的次品率为：

$$fdd_1 = \frac{fd_1}{(fd_1 + (1 - fd_1) * fdF)}$$

不合格成品拆解后得到的合格零配件 1 的个数为：

$$nd_1 = (1 - fdd_1) * nd$$

在拆解之前不检测零配件 1、零配件 2 的情况下，拆解之后得到的合格零配件 1 和零配件 2 的个数分别为：

$$nd_1 = \frac{fdF * (1 - fd_1) * (1 - fd_2) + fd_2 * (1 - fd_1)}{(1 - (1 - fdF) * (1 - fd_1) * (1 - fd_2))} * nd$$

$$nd_2 = \frac{fdF * (1 - fd_2) * (1 - fd_1) + fd_1 * (1 - fd_2)}{(1 - (1 - fdF) * (1 - fd_1) * (1 - fd_2))} * nd$$

拆解之后得到的成品数量为：

$$nd_0 = \min(nd_1, nd_2)$$

不合格成品的处理费用为：（不合格成品数=拆解费用+零配件 1、2 检验费用+（装配费用+检测成本））

$$t = dc * nd + ic_1 * (1 - x_1) * nd + ic_2 * (1 - x_2) * nd + (cF + icF) * nd_0$$

拆解重装后，得到的合格成品数为：

$$nd_{01} = nd_0 * (1 - fdF)$$

最后，当我们根据上述步骤进行完可能的检验和拆解之后，合格的成品数为：

$$nfp = nF - nd + nfp_0$$

总成本为：

$$T = Ic_1 + Ic_2 + IcF + RL + n_{10} * c_1 + n_{20} * c_2 + nf * cF + t$$

利润为：

$$P = nfp * p - T$$

5.2.6 结论

根据上述步骤进行建模与计算，我们得到如下表格：

表格 2 6 种不同情况下的方案及其最高利润					
情况	是否检验 零配件 1	是否检验 零配件 2	是否检验 成品	不合格成品 是否拆解	利润（取零配件数量 1000 为例）
1	0	0	0	1	27278.76954
2	0	0	0	1	26282.5133
3	0	0	0	1	24878.76954
4	0	0	1	1	25567.26471
5	0	0	0	1	26914.18595
6	0	0	0	1	27438.19066

在上述六种情况下，取每种零配件数量 1000 为例，本文根据上表可知各种情况分别可以获得的最高利润及其对应的方案。不同情况下的最高利润如下图所示：

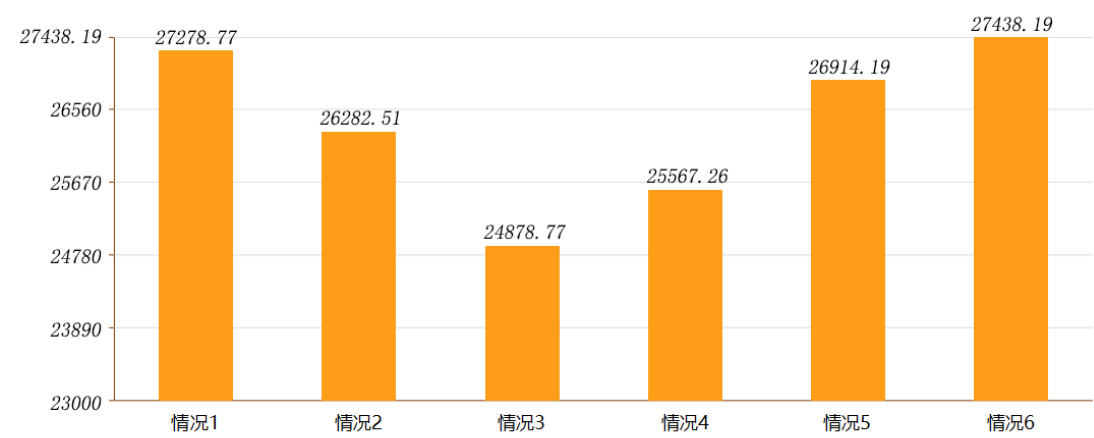


图 3 六种不同情况下的最高利润

据上图分析可知，情况 1、6 获得的最高利润较高，情况 3 获得的最高利润较低。

下面，本文以情况 4 为例，分析在何种情况下更易获得利润最大化的最优方案。我们将情况 4 中的全部 16 种方案列出，计算各方案中利润大小并进行降序排列，如下表所示：

表格 3 16 种不同方案及其对应利润（以情况 4 为例）

情况	是否检验 零配件 1	是否检验 零配件 2	是否检 验成品	不合格成品 是否拆解	利润（零配件比为 1 时， 取数量为 1000 为例）
4	0	0	1	1	25567.26471
4	0	0	1	0	24678.03279
4	0	0	0	1	21567.26471
4	0	0	0	0	20678.03279
4	0	1	1	1	15193.3037
4	1	0	1	1	15193.3037
4	0	1	1	0	14603.55556
4	1	0	1	0	14603.55556
4	1	1	1	1	13625.6
4	1	1	1	0	12608
4	0	1	0	1	11993.3037
4	1	0	0	1	11993.3037
4	0	1	0	0	11403.55556
4	1	0	0	0	11403.55556
4	1	1	0	1	10425.6
4	1	1	0	0	9408

接下来，我们从是否检验零配件、是否检验成品及不合格成品是否拆解三个方面对上述结果进行分析。

(1)是否检验零配件

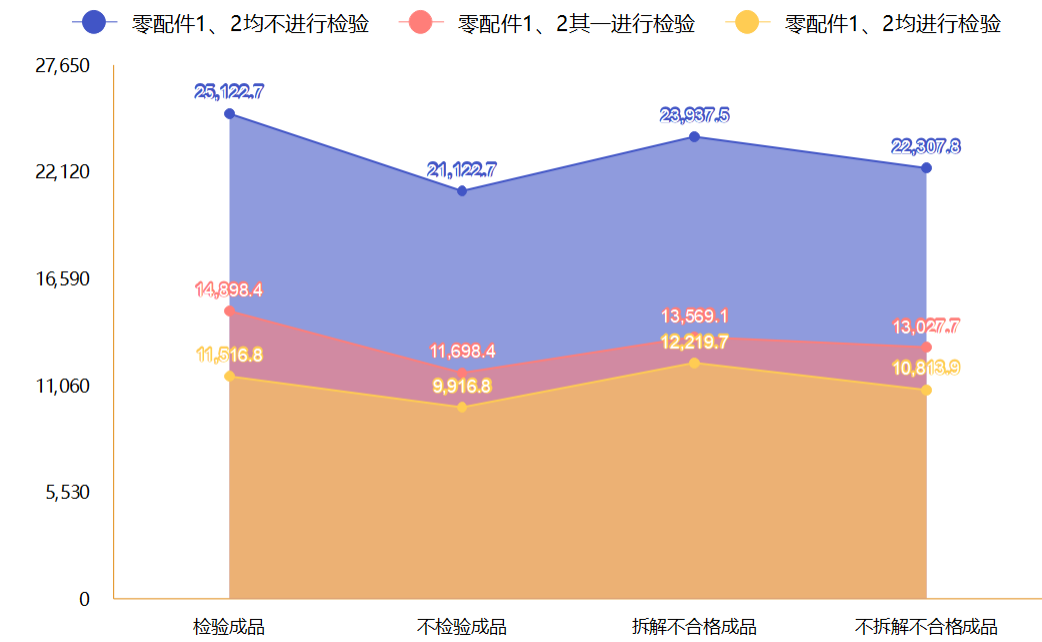


图 4 不同零配件检验方案下的利润面积图

据图分析可知，不检验零配件所获利润较高，两种零配件检验其一所获利润比两种零配件均进行检验所获利润略高。

因此，我们可以得出，在其他条件相同时，不检验零配件所获利润相对较高。

(2)是否检验成品

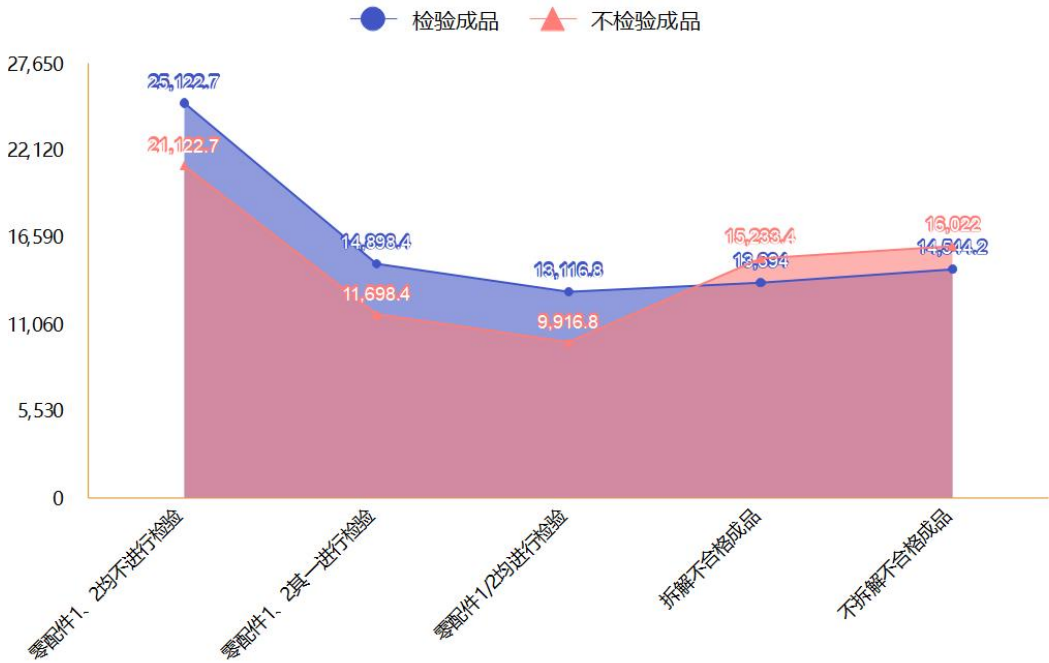


图 5 不同成本检验方案下的利润面积图

据图分析可知，与不检验成品相比，检验成品所获的利润略高。

因此，在其他条件相同时，检验成品所获利润相对较高。

(3)不合格成品是否拆解

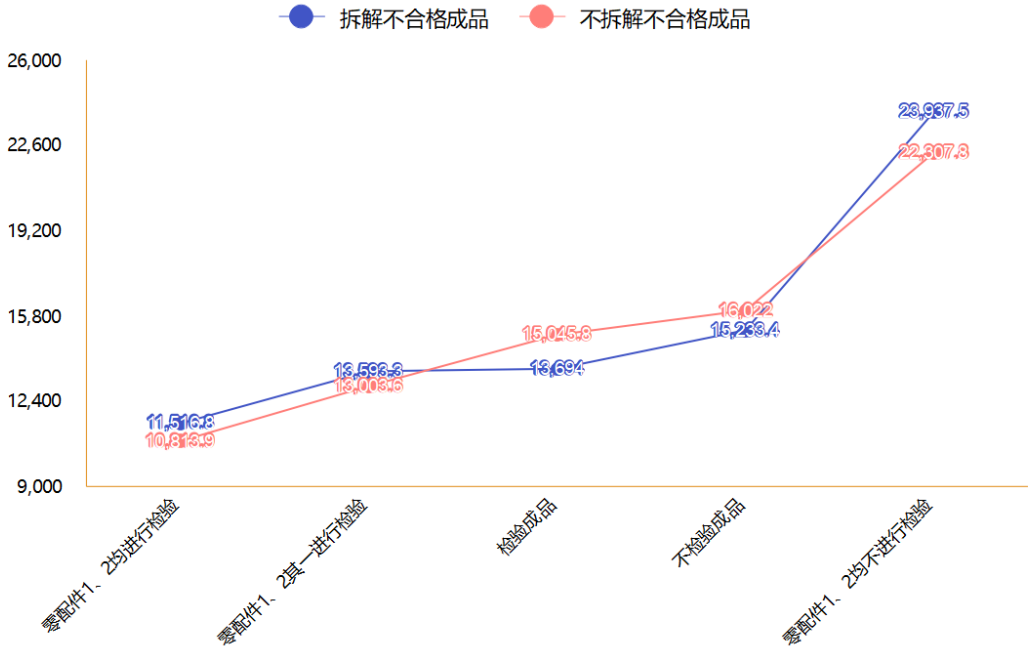


图 6 不同成品拆解方案下的利润折线图

据图分析可知，拆解成品与不拆解成品在不同情况下的变化趋势相同，且各项数据差距较小，可以认为是否拆解成品对利润影响较小。

由于是否拆解成品对于整体利润影响较小，我们将上述各种情况所得利润按顺序排列，如下图所示：

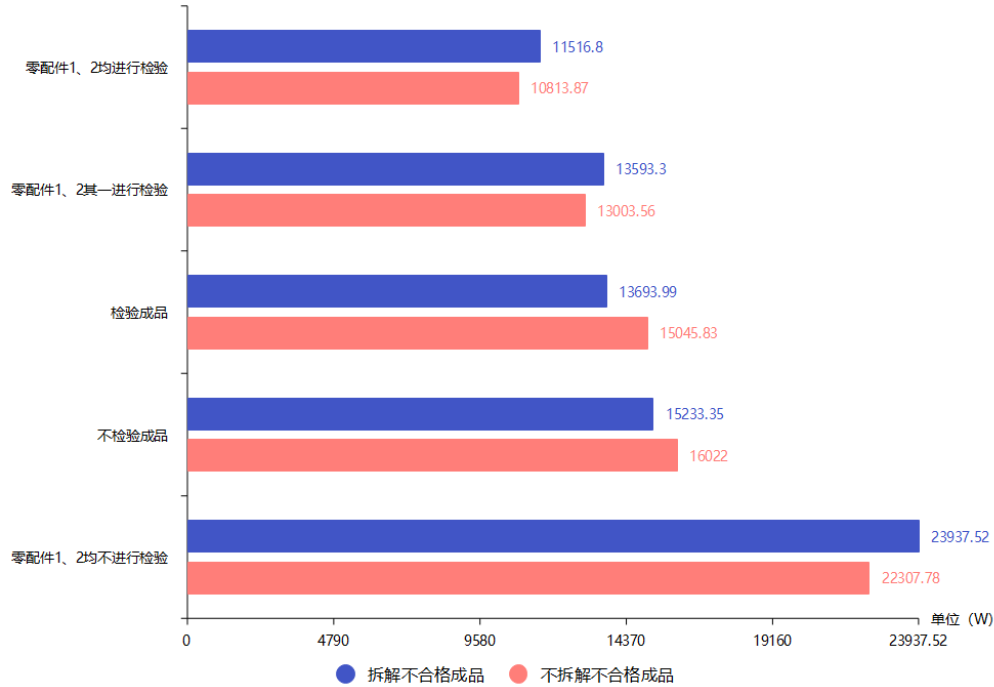


图 7 不同成品拆解方案下的利润柱状图

据上图可知，不检验零配件获得的利润最高，其次是检验成品；检验两种零配件获得的利润最低。

综上，兼顾产品质量和成本，不检验零配件、检验成品是最佳方案。

5.3 第三问的模型建立与求解

问题 3 是问题 2 的扩展和延伸。我们需要根据表 2 中的情况，进一步分析在多道工序、多个零配件情况下的生产决策。本文构建了一个更加普遍的数学模型，对整个生产流程进行了优化，保证了产品质量并尽可能地降低成本，最终选择最优的检测和处理策略。

针对问题 3，本文进行如下模型假设：

- (1)假设没有两种不同的半成品共享一个零配件；
- (2)假设组成每个半成品的零件个数相等，组成每个成品的半成品数与组成每个半成品的零配件数相等；
- (3)假设所有经过检测的不合格、不丢弃的成品/半成品拆成的未经检测的半成品/零配件都会被检测；若半成品/零配件经检测合格则再次投入生产，不合格直接丢弃；

(4)对于半成品的组成零件个数不相同的情况，对最后一个半成品添加空零件；

(5)假设空零件的购买单价、检测成本和次品率均为零；空半成品的次品率、装配成本、检测和拆解费用也为零。

5.3.1 生产各阶段流程介绍与分析

首先，我们需要选择是否需要检测各个零配件。若进行检测，则丢弃次品并将合格品投入生产；否则，零配件将直接投入生产。进行完零配件检测的工序之后，我们对其进行装配，进而获得半成品。其次，对于半成品，我们需要选择是否进行检测。若进行检测，则丢弃次品并将合格品继续投入后续生产；否则，半成品将直接投入后续生产。对于成品，我们需要选择是否进行检测。若得到不合格成品，如果前期未进行成品检测，则需支付调换费用。对于检测出和调换后的不合格成品，我们需要选择是否进行拆解。若进行拆解，则对拆解后未经检验的半成品进行重新检验，并再次投入生产；否则直接将其丢弃。

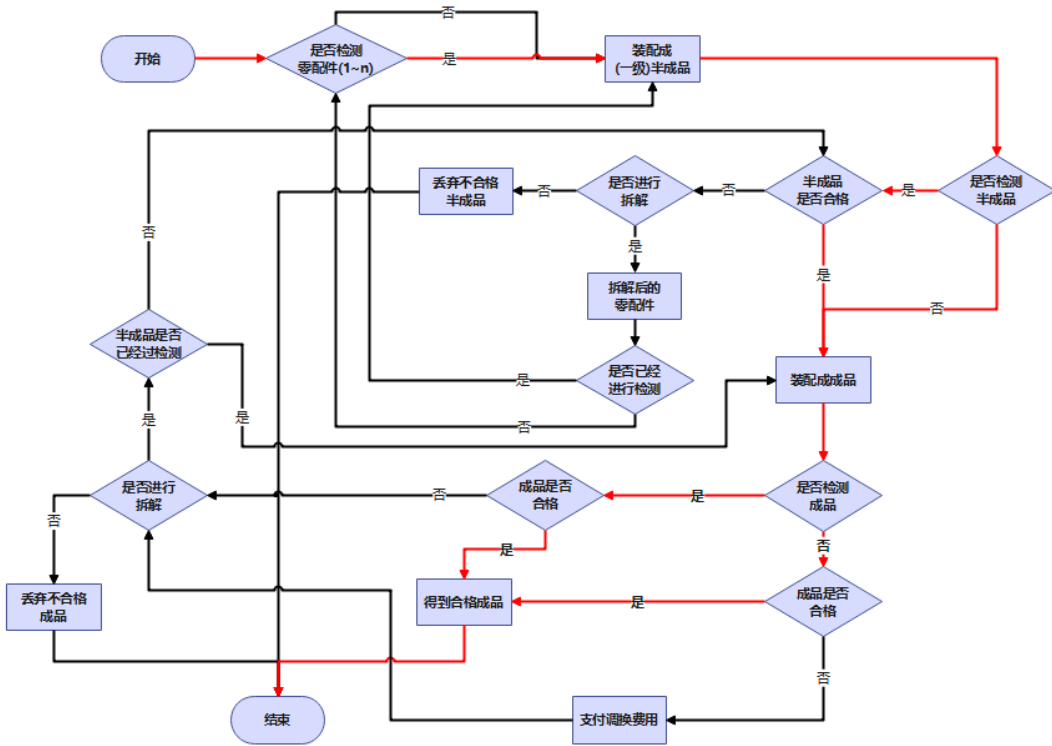


图 8 第三问流程图

根据题目所给表 2 可知，在本题背景下，企业在生产中遇到的情况一共可分为六种。在任意一种情况下，企业的生产过程都需要在保证产品质量的同时降低成本，实现利润最大化。

由此，本文通过以下五个步骤来进行选择：

1. 根据零配件的次品率和检测成本选择是否检验零配件。

2. 根据半成品的次品率和检测成本选择是否检验半成品。
3. 对比不合格半成品的检验费用，选择是否拆解不合格半成品。
4. 根据成品的次品率和检测成本选择是否检验成品。
5. 对比不合格成品的调换损失和检验费用，选择是否拆解不合格成品。

下面通过上述五个步骤分别进行分析，计算不同方案的利润，对比分析选出最优方案。

5.3.2 对于是否检验零配件的分析

第一步，我们需要选择是否对零配件(共 n 个)进行检测。如果对某种零配件不检测，这种零配件将会直接投入生产，进入到装配环节；否则，检测出的零配件次品将会被丢弃。如果零配件其一是次品，则半成品必为次品；如果零配件均为合格品，则半成品的次品率为 10%（装配时产生的固有次品率）。

5.3.3 对于是否检验半成品的分析

其次，对于生产出的半成品，我们可以选择检测或者不检测。若进行检测，则丢弃次品并将合格品继续投入后续生产；否则，半成品将直接投入后续生产。

半成品的次品率由零配件质量和装配过程中的损耗决定。对于是否对半成品进行检测，需要综合考虑半成品的次品率、装配成本、检测成本以及后续成品的质量控制需求。若半成品次品率较高且装配成本较高，建议进行检测；若半成品次品率较低，且检测成本较高，可以不进行检测，直接进行装配。

5.3.4 对于是否拆解不合格半成品的分析

若在检测后得到不合格半成品，如果前期未进行成品检测，则对拆解后未经检验的零配件进行重新检验，并再次投入生产；否则可直接将其丢弃。

当检测出不合格半成品时，企业可以选择拆解或直接丢弃。若拆解费用低于半成品制作成本，且拆解出的零配件能够进行有效利用，建议进行拆解；否则，若拆解费用较高，或者拆解出的零配件质量低劣，建议直接丢弃不合格半成品，避免产生不必要的拆解费用。

5.3.5 对于是否检验成品的分析

对于生产出的成品，我们需要选择是否进行检测。若检测结果合格或选择不检测，则可以直接投入市场进行销售；若在检测后得到不合格成品，则进入拆解选择阶段；如果不合格成品前期并未进行成品检测，则需拆解为半成品并重新进行检测。

成品的检测决策需要考虑成品的市场售价、检测成本、次品率以及调换损失等。若成品的市场售价较高，调换损失大，且检测成本较低时，建议进行成品检测；否则，若成品的次品率较低，且检测成本较高，建议不检测成品，直接将其投入市场。

5.3.6 对于是否拆解不合格成品的分析

对于检测出和调换后的不合格成品，我们需要选择是否进行拆解。若进行拆解，则对拆解后未经检验的半成品进行重新检验，并再次投入生产；否则可直接将其丢弃。

当检测出不合格成品时，企业可以依据拆解费用和回收半成品的质量及价值，选择拆解回收部分合格半成品，或直接丢弃不合格成品。如果拆解费用较低，且拆解出的零配件质量较好，能够有效降低后续的零配件采购成本，建议进行拆解；如果拆解费用较高，或拆解出的半成品不具备再利用价值，建议直接丢弃不合格成品，避免额外的拆解成本。

5.3.7 方案计算与分析

在实际计算过程中，本文将上述流程分为购买、检测、装配、拆解四个步骤。下文详细介绍计算过程：

(1) 购买。

各级产品的购买/装配费用为：

$$C(i, j, a_1 + 1) = num0 * c(i, j)$$

(2) 检测。

对于是否进行检测，我们可以用以下公式进行判断：

$$D(i, j, a_i + 1) = a_i$$

其中， a_i 表示第 i 道工序时是否检验（ a_i 取 1 表示检验第 i 级产品，取 0 表示不检验）。 $i = 1$ 时检验零配件， $i = 2$ 时检验半成品， $i = 3$ 时检验成品。

零配件检测的费用为：

$$Ic(i, j, a_1 + 1) = a_1 * num0 * ic(i, j)$$

进入下一级生产的产品数量为：

$$n_1(i, j, a_i + 1) = (1 - a_i * fd(i, j)) * num0$$

半成品检测费用为：

$$Ic(i, j, a_2 + 1) = a_2 * n_1(i, j, a_2 + 1) * ic(i, j)$$

进入到下一轮生产的半成品个数为：

$$n_1(i, j, a_2 + 1) = n_1(i, j, a_2 + 1) + re_1$$

半成品/成品的合格率为：

$$\begin{aligned} rate(i, j) = & prod \left(1 \right. \\ & - prod \left(fd(i - 1, (j - 1) * k + 1 : j * k) \right. \\ & \left. \left. * \left(1 - D(i - 1, (j - 1) * k + 1 : j * k, a_1 + 1) \right)' \right) \right) * (1 - fd(i, j)) \end{aligned}$$

未检测过直接售卖后被退回的不合格成品的调换费用为：

$$RL = n_1(i, j) * (1 - rate(i, j)) * rl * (1 - D(i, j, a_3 + 1))$$

(3) 装配。

生产半成品的零配件的数量为：

$$n_1(i, j, a_i + 1) = (1 - a_i * fd(i, j)) * num0$$

半成品装配费用为：

$$C(i, j, a_2 + 1) = n_1(i, j, a_2 + 1) * c(i, j)$$

装配得到的半成品个数为：

$$n_1(i, j, a_2 + 1) = \min(n_1(i - 1, (j - 1) * k + 1 : j * k, a_1 + 1))$$

(4) 拆解。

保存决策变量：

$$E(i, j, b_i + 1) = b_i$$

其中， b_i 表示是否拆解不合格的第*i*级产品（ b_i 取 1 表示拆解第*i*级产品，取 0 表示不拆解）

需要拆解的不合格半成品/成品个数为：

$$n_2(i, j, b_2 + 1) = num(i, j, a_2 + 1) * (1 - rate(i, j))$$

需要拆解的零件/半成品的拆解费用为：

$$DC(i, j, b_2 + 1) = E(i, j, b_2 + 1) * dc(i, j) * n_2(i, j)$$

拆解后的零件/半成品合格率为：

$$pd = prod(tdh(1:k))$$

$$pass1 = (1 - pd/tdh(h) * td(i, j)) * tdh(h)$$

$$fail1 = 1 - pd * td(i, j)$$

$$tdd(h) = pass1/fail1$$

未检测过的零配件/半成品送检：

$$icg = ic(i - 1, (j - 1) * k + 1 : j * k) * n_2(i, j)$$

再装配的半成品/成品个数为：

$$re_0 = E(i, j, b_2 + 1) * \min(r) * n_2(i, j, b_2 + 1)$$

最终得到的合格再装配半成品/成品个数为：

$$re = re_0 * (1 - fd(i, j))$$

最终得到的合格再装配半成品/成品检测费为：

$$icg_1 = re_0 * ic(i, j)$$

(5) 最终利润。

一次生产合格的成品数量为：

$$nfp = n_1(i, j, a_3 + 1) * rate(i, j)$$

不合格成品的处理费用为：

$$t(i, j, b_3 + 1) = b_3 * (DC + icg + icg_1 + re * c(i, j) + RL)$$

最终生产成本为：

$$T = sum(C(1, :, a_1 + 1) + Ic(1, :, a_1 + 1) + C(2, :, a_2 + 1) + Ic(2, :, a_2 + 1) + t(2, :, b_2 + 1) + C(3, :, a_3 + 1) + Ic(3, :, a_3 + 1) + t(3, :, b_3 + 1))$$

产品经过市场销售后的最终利润为：

$$P = nfp * p - T$$

5.3.8 结论

根据上述步骤进行建模与计算，我们得到表格(附录 21)。

在表格中，a 表示第几道工序时是否检验（例如 a1 取 1 表示检验零配件，a2 取 0 表示不检验半成品），b 表示是否拆解不合格的 n 级产品（例如在本题中 b2 取 1 表示拆解不合格半成品，b3 取 0 表示不拆解不合格成品）

我们将表格中利润进行降序排列，通过对表中数据的分析可知，最终利润与 a1、a2 高度相关。如果对零配件和半成品进行检验，则更容易获得更高的利润；成品是否检验、不合格产品是否进行拆解对最终利润影响不大。

5.4 第四问的模型建立与求解

5.4.1 根据问题 1 的方法获得抽样检测后零配件、半成品和成品的次品率

问题 1 采用抽样检测的方法。

5.4.2 根据问题 1 所得的零配件次品率重新求解问题 2

根据抽样得到的次品率重做问题 2。

情况	是否检验 零配件 1	是否检验 零配件 2	是否检 验成品	不合格成品 是否拆解	利润（零配件比为 1 时， 取数量为 1000 为例）
1	0	0	0	1	27365.60137
2	0	0	0	1	26232.00394
3	0	0	0	1	25205.60137
4	0	0	1	1	25617.98399
5	0	0	0	1	27029.82454
6	0	0	0	1	27480.76612

情况	是否检验 零配件 1	是否检验 零配件 2	是否检 验成品	不合格成品 是否拆解	利润（零配件比为 1 时， 取数量为 1000 为例）
1	0	0	0	0	27.17782

1	0	0	0	1	27.3656
1	0	0	1	0	24.71782
1	0	0	1	1	24.9056
1	0	1	0	0	20.30438
1	0	1	0	1	20.4546
1	0	1	1	0	18.04118
1	0	1	1	1	18.1914
1	1	0	0	0	20.79866
1	1	0	0	1	20.95184
1	1	0	1	0	18.56006
1	1	0	1	1	18.71324
1	1	1	0	0	17.59582
1	1	1	0	1	17.86826
1	1	1	1	0	15.35722
1	1	1	1	1	15.62966

在 10%(抽样得到)的次品率下，不检测零配件、检测成品是最优决策。

5.4.3 根据问题 1 所得的零配件次品率重新求解问题 3

根据抽样得到的次品率重做问题 3。

是否检测成品影响不大，但是检测零配件可以显著提高利润。

5.4.4 结论

问题 4 中所得结论与问题 2、问题 3 相同。

六、模型的评价与推广

6.1 模型的优点

(1)模型稳健性较好：本模型在设计和分析过程中考虑了包括拆解、检验与否的多种情况，能够在不同次品率和工序下保持稳定的表现，且在最优决策不能实施时可以选择次优决策，具有较强的抗干扰能力。

(2)代码可解释性好：模型代码结构清晰，注释详尽，便于其他研究人员或工程师理解和修改，提高了模型的可维护性和可复用性。

6.2 模型的不足

(1)忽略了生产过程中的动态变化：本文未充分考虑生产过程中可能出现的动态变化，如市场需求波动、供应链中断等，这可能限制模型的实际应用范围。

(2)忽略了生产能力的限制：本文在做出生产决策时未考虑生产线的实际生产能力，这可能导致决策方案在实际操作中难以实现。

6.3 模型的改进

(1)引入更先进的抽样技术：可以根据购买零件的个数采用更高效的抽样方法，以减少抽样误差，提高模型对次品率估计的准确性。

(2)考虑动态因素：在模型中加入时间序列分析或预测模型，以更好地应对生产过程中的动态变化，增强模型的适应性。

6.4 模型的推广方向

(1)其他行业的生产决策：模型可以推广至航空航天、汽车、化工等行业的生产决策问题，通过调整相关假设和参数，满足不同行业的要求。

(2)库存管理与供应链优化：可以考虑半成品的售卖，通过引入半成品的售价和需求波动参数，实现成本效益最大化。

(3)新产品开发：模型可应用于新产品开发过程中的决策支持，通过模拟不同生产方案下的成本和利润，辅助进行新产品的生产决策。

七、参考文献

[1]刘海涛,邵松世,张志华.一种改进的指数型序贯抽样检验方法[J].海军工程大学学报,2019,31(04):95-99.

[2]鲁海波.若干统计模型下序贯压缩估计方法的研究[D].中国科学技术大学,2015.

[3]韦金芬,宋保维,毛昭勇.基于序贯验后加权检验的计数抽样方法[J].计算机工程,2012,38(01):279-280.

[4]韩培强.基于多级决策的选煤生产过程诊断研究[J].煤炭技术,2022,41(11):244-246. DOI:10.13301/j.cnki.ct.2022.11.060.

八、附录

附录 1

基于单次抽样的第一问求解代码（T11.m）

```
% 参数
p = 0.1; % 标称次品率
alpha_reject = 0.05; % 拒收的显著性水平
beta_accept = 0.1; % 接收的显著性水平

% 确定拒收时的样本量和次品数上限
n = randi([15,3000]); % 假设样本量初值
disp(n)
c_reject = binoinv(1 - alpha_reject, n, p); % 95%信度下拒收时允许的最大次品数
fprintf('在 95%%信度下拒收的次品数上限为: %d\n', c_reject);

% 确定接收时的样本量和次品数上限
c_accept = binoinv(beta_accept, n, p); % 90%信度下接收时允许的最大次品数
fprintf('在 90%%信度下接收的次品数上限为: %d\n', c_accept);
```

附录 2

基于双重抽样的第一问求解代码（T12.m）

```
% 参数
p0 = 0.1; % 标称次品率
min_n1 = 15; % 第一次抽样最小样本量
max_n1 = 300; % 第一次抽样最大样本量
c1 = 2; % 第一次抽样时允许的最大次品数
c2 = 5; % 第一次抽样时次品数超过 c2 直接拒收

% 第一次抽样：使用随机函数生成样本量 n1
n1 = randi([min_n1, max_n1]); % 从[min_n1, max_n1]中随机生成 n1
x1 = binornd(n1, p0); % 第一次抽样的次品数
fprintf('第一次抽样样本量: %d, 次品数: %d\n', n1, x1);

% 判断第一次抽样的结果
```

```

if x1 <= c1
    disp('接收该批次零配件');
elseif x1 > c2
    disp('拒收该批次零配件');
else
    % 根据第一次的结果调整第二次样本量
    if x1 > (c1 + c2) / 2
        n2 = randi([max_n1, max_n1 + 10]); % 如果接近拒收边界，增加样本量
    else
        n2 = randi([min_n1, max_n1]); % 样本量保持在正常范围
    end

    % 第二次抽样
    x2 = binornd(n2, p0); % 第二次抽样的次品数
    total_defects = x1 + x2; % 两次抽样的总次品数
    fprintf('第二次抽样样本量: %d, 次品数: %d\n', n2, x2);

    % 最终判断
    if total_defects <= c2
        disp('接收该批次零配件');
    else
        disp('拒收该批次零配件');
    end
end
end

```

附录 3

基于序贯抽样的第一问求解代码（sprt.m）

```

function [n_total,a,b] = sprt(a,b,p0,p1,alpha,beta)
% 计算接收和拒收界限
A = log(beta / (1 - alpha)); % 接收界限
B = log((1 - beta) / alpha); % 拒收界限
fprintf('接收界限 A: %.2f\n', A);
fprintf('拒收界限 B: %.2f\n', B);

% 初始化
logL = 0; % 似然比的对数
n_total = 0; % 累计抽样数量
x_total = 0; % 累计次品数

```

```

% 序贯抽样
while true
    x_new = binornd(1, p1); % 假设每次抽样结果
    x_total = x_total + x_new;
    n_total = n_total + 1;

    % 更新对数似然比
    logL = logL + log(p1 / p0) * x_new + log((1 - p1) / (1 - p0)) * (1 - x_new);

    % 判断是否接收、拒收或继续抽样
    if logL <= A
        disp('接收该批次零配件');
        a = a + 1;
        break;
    elseif logL >= B
        disp('拒收该批次零配件');
        b = b + 1;
        break;
    end
end
end
end

```

附录 4

基于序贯抽样的第一问求解代码（T131.m）

```

% 参数
a = 0; % 接收次数
b = 0; % 拒收次数
p0 = 0.1; % 标称次品率
p1 = 0.15; % 假设次品率
alpha = 0.05; % 第一类错误概率（生产者风险）
beta = 0.1; % 第二类错误概率（消费者风险）
N_total = 0; % 总抽样次数

for i = 1:200
    [n_total, a, b] = sprt(a, b, p0, p1, alpha, beta);
    N_total = N_total + n_total;
end
a

```

b
N_total/200

附录 5

第二问求解代码（CostMinimize.m）

```
function best_decision = CostMinimize(n10,n20,fd1,c1,ic1,fd2,c2,ic2,fdF,cF,icF,p,rl,dc)
M = 0;
nfp0 = 0;
best_decision = [];

for x1 = 0:1% 是否检验零件 1, 2
    lc1 = x1 * n10 * ic1;
    n11 = n10 * (1 - x1 * fd1);
    for x2 = 0:1
        lc2 = x2 * n20 * ic2;
        n21 = n20 * (1 - x2 * fd2);
        for fpi = 0:1% 是否检验成品
            nF = min(n11,n21);
            lcF = fpi * nF * icF;
            RL = (1-fpi) * nF * rl * fdF;% 不检测成品时的调换损失
            nd = nF * fdF;% 成品次品数量
            for dis = 0:1
                [t, nd01, nd, nd0] = DisposalCost(nd,x1,x2,fdF,fd1,fd2,ic1,ic2,icF,cF,dc);
                t = t * dis;
                % 处理后合格成品数
                nfp = nF - nd0 + nd01;
                T = lc1 + lc2 + lcF + RL + n10 * c1 + n20 * c2 + nF * cF + t;
                P = nfp * p - T;
                if P > M
                    M = P;
                    best_decision = [x1, x2, fpi, dis, P];
                    disp(best_decision);
                end
            end
        end
    end
end

end
end
```


end

附录 6

第二问求解代码（DisposalCost.m）

```
function [t, nd01, nd, nd0] = DisposalCost(nd,x1,x2,fdF,fd1,fd2,ic1,ic2,icF,cF,dc)
    switch x1
        case 1
            nd1 = nd;
            switch x2
                case 1
                    nd2 = nd;
                case 0
                    fdd2 = fd2 / (fd2 + (1 - fd2) * fdF);
                    nd2 = (1 - fdd2) * nd;
            end
        case 0
            switch x2
                case 1
                    nd2 = nd;
                    fdd1 = fd1 / (fd1 + (1 - fd1) * fdF);
                    nd1 = (1 - fdd1) * nd;
                case 0
                    nd1 = (fdF * (1 - fd1) * (1 - fd2) + fd2 * (1 - fd1)) / (1 - (1 - fdF) * (1 - fd1)
                    * (1 - fd2)) * nd;
                    nd2 = (fdF * (1 - fd2) * (1 - fd1) + fd1 * (1 - fd2)) / (1 - (1 - fdF) * (1 - fd1)
                    * (1 - fd2)) * nd;
            end
        end
    end
    nd0 = min(nd1,nd2);% 零件检验后可装配成成品的数量
    t = dc * nd + ic1 * (1-x1) * nd + ic2 * (1-x2) * nd + (cF + icF) * nd0;
    %fdd = fdF;% 再处理成品次品率
    nd01 = nd0 * (1 - fdF);
    nd = nd0 * fdF;
    % 不合格成品处理费用=拆解费用+零件 1、2 检验费用+(装配费用+检测成本)
end
```

附录 7
第二问求解代码（t2.m）
<pre> clear;clc disp('情况 1: '); A1 = CostMinimize(1000,1000,0.1,4,2,0.1,18,3,0.1,6,3,56,6,5); disp('情况 2: '); A2 = CostMinimize(1000,1000,0.2,4,2,0.2,18,3,0.2,6,3,56,6,5); disp('情况 3: '); A3 = CostMinimize(1000,1000,0.1,4,2,0.1,18,3,0.1,6,3,56,30,5); disp('情况 4: '); A4 = CostMinimize(1000,1000,0.2,4,1,0.2,18,1,0.2,6,2,56,30,5); disp('情况 5: '); A5 = CostMinimize(1000,1000,0.1,4,8,0.2,18,1,0.1,6,2,56,10,5); disp('情况 6: '); A6 = CostMinimize(1000,1000,0.05,4,2,0.05,18,3,0.05,6,3,56,10,40); A=[A1;A2;A3;A4;A5;A6]; </pre>

附录 8
第二问求解代码（t2_4.m）
<pre> clear;clc disp('情况 4: '); [A,best_decision] =CostMinimize(1000,1000,0.2,4,1,0.2,18,1,0.2,6,2,56,30,5); A </pre>

附录 9
第三问求解代码（CostMinimize7.m）
% 计算各项成本

```

function [C,lc,t,nfp] = CostMinimize7(num0,m,n,fd,c,ic,dc,rl,td,k)

% 定义决策变量
A = zeros(m+1,n);           % 检测
B = zeros(m+1,n);           % 拆解
D = zeros(m+1,n,3);         % 储存决策变量

% 初始化
lc = [];
n1 = NaN;
n2 = zeros(3,9,2);
n1 = zeros(3,9,2);
k = ceil(power(n,1/m)); % 每个本级产品对应其次级产品的数量
t = [];

for i = 1:m+1
    for j = 1:n/(k^(i-1))
        % 零件部分
        if i == 1
            for a1 = 0:1
                D(i,j,a1+1) = a1;           % 存放决策变量
                lc(i,j,a1+1) = a1 * num0 * ic(i,j);
                n1(i,j,a1+1) = (1 - a1 * fd(i,j)) * num0; % n1(i,j,1):不检测，进入下一轮生
                % 产的零配件； n1(i,j,2):检测后进入下一轮生产的零配件；
                C(i,j,a1+1) = num0 * c(i,j);
            end
        end
        % 半成品部分
        if i > 1 && i <= m
            for a1 = 0:1 % 零件是否检测过
                for a2 = 0:1 % 半成品是否检测
                    % 检测？
                    D(i,j,a2+1) = a2;
                    n1(i,j,a2+1) = min(n1(i-1, (j-1)*k+1 : j*k, a1+1)); % 装配得到的
                    % 半成品个数
                    C(i,j,a2+1) = n1(i,j,a2+1) * c(i,j); % 装配费
                    lc(i,j,a2+1) = a2 * n1(i,j,a2+1) * ic(i,j); % 检测费
                    rate(i,j) = prod(1 - prod(fd(i-1, (j-1)*k+1:j*k)*(1 - D(i-1, (j-1)*k+1:j*k,
                    a1+1)))) * (1 - fd(i,j)); % 半成品的合格率
                    % n1(i,j,a2+1) = (1 - A(i,j) * (1 - rate(i,j))) * num(i,j,a2+1); % 进入下
                    % 一轮生产、选择是否检测前的的半成品个数

                    if a2 == 1 % 半成品检测
                        icg = 0;

```

```

for b2 = 0:1 % 拆解? b=0 不拆, 直接丢弃; b=1 拆, 计算拆解费、
检测费及再装配费

    Ch = check(i,j,k,a1,a2,D);
    E(i,j,b2+1) = b2;
    [re1, DC,icg1] =
SelectDis(i,j,k,E,a2,b2,n1,rate,dc,Ch,td,fd,ic); % re:再装配的半成品个数
    for g = 1:k
        if Ch(g) == 0
            icg = sum(ic(i-1, (j-1)*k+1:j*k) * re1); % 未检
测过的零配件/半成品送检
        end
    end
    t(i,j,b2+1) = b2 * (DC(i,j,b2+1) + icg + icg1 + re1 *
c(i,j)); % 不合格半成品处理费用; re1 * c(i,j):再装配费
    end
    end
    if a2 == 0 % 半成品不检测
        C(i,j,a2+1) = n1(i,j,a2+1) * c(i,j);
    end
    end
    end
    n1(i,j,a2+1) = n1(i,j,a2+1) + re1; % 进入到下一轮生产的半成品个数
end
% 成品部分
if i == m+1
    for a2 = 0:1 % 半成品是否检测过
        for a3 = 0:1 % 成品是否检测
            D(i,j,a3+1) = a3;
            n1(i,j,a3+1) = min(n1(i-1, (j-1)*k+1 : j*k, a2+1)); % 生产出的未经
检测的成品个数
            C(i,j,a3+1) = n1(i,j,a3+1) * c(i,j);
            lc(i,j,a3+1) = a3 * n1(i,j,a3+1) * ic(i,j);
            rate(i,j) = rate1(i,j,k,fd,D,a1,a2);
            [nfp, t] = Final(i,j,k,c,a1,a2,a3,D,rate,n1,rl,dc);
            disp(D);
        end
    end
    end
    end
    end
    end
    disp(D);
end

```

附录 10

第三问求解代码（check.m）

```
% 查看零配件/半成品是否检测过
function Ch = check(i,j,k,a1,a2,D)
if i == 2
    Ch = D(i-1, (j-1)*k+1:j*k,a1+1); % a1=0 未检测过;a1=1 检测过
else
    Ch = D(i-1, (j-1)*k+1:j*k,a2+1);
end
```

附录 11

第三问求解代码（SelectDis.m）

```
% 选择是否拆解
function [re,DC,icg1] = SelectDis(i,j,k,E,a2,b2,num,rate,dc,Ch,td,fd,ic)
    % B(i,j) = b2;
    % E(i,j,b2+1) = B(i,j);
    n2(i,j,b2+1) = num(i,j,a2+1) * (1 - rate(i,j)); % 需要拆解的不合格半成品/成品
    个数
    DC(i,j,b2+1) = E(i,j,b2+1) * dc(i,j) * n2(i,j);
    % 拆解后得到新的零件/半成品的合格率
    r = PassRate(i,j,k,Ch,td);
    re0 = E(i,j,b2+1) * min(r) * n2(i,j,b2+1); % 再装配的半成品/成品个数
    re = re0 * fd(i,j); % 最终得到的合格再装配半成品/成品个数
    icg1 = re0 * ic(i,j); % 最终得到的合格再装配半成品/成品检测费
end
```

附录 12

第三问求解代码（PassRate.m）

```
% 计算拆解后的零件/半成品合格率
function tdd = PassRate(i,j,k,Ch,td)

tdh = td(i-1,(j-1)*k+1 : j*k);
for g = 1:k
    if Ch(g) == 1
        tdh(g) = 1;
    end
end
pd = prod(tdh(1:k));

for h = 1:k
    pass1 = (1 - pd/tdh(h) * td(i,j)) * tdh(h);
    fail1 = 1 - pd * td(i,j);
    tdd(h) = pass1 / fail1;
end

end
```

附录 13

第三问求解代码（Final.m）

```
function [nfp,t] = Final(i,j,k,c,a1,a2,a3,D,rate,n1,r1,dc)
    % 成品不检测：不合格产品需要调换，调换后的不合格产品需要选择拆或不拆
    nfp = n1(i,j,a3+1) * rate(i,j); % 一次生产合格成品数量
    icg = 0;
    for b3 = 0:1
```

```

        E(i,j,b3+1) = b3;
        Ch = check(i,j,k,a1,a2,D);
        [re, DC, icg1] = SelectDis(i,j,k,b3,n1,rate,dc,Ch); % re:得到再
        装配的成品个数
        nfp = nfp + re; % 最终成品个数
        for g = 1:k
            if Ch(g) == 0
                icg = ic(i-1, (j-1)*k+1:j*k) * n2(i,j); % 未检测过的零
                配件/半成品送检
            end
        end
        RL = n1(i,j) * (1 - rate(i,j)) * r1 * (1 - D(i,j,a3+1));
        t(i,j,b3+1) = b3 * (DC + icg + icg1 + re * c(i,j) + RL) ;
    end

end

```

附录 14

第三问求解代码（t3.m）

```

clear;clc

m=2;
n=8;
num0=1;
n = power(ceil(power(n,1/m)),m);
A = zeros(1,n);
p = 200;
rl = 40;
B = zeros(m+1,n); % 拆解

% 定义决策变量
A = zeros(m+1,n); % 检测
B = zeros(m+1,n); % 拆解

```

```

D = zeros(m+1,n,3);          % 储存决策变量
D = zeros(m+1,n,3);

% 初始化
lc = [];
n1 = NaN;
n2 = zeros(3,9,2);
num = zeros(3,9,2);
k = ceil(power(n,1/m));% 每个本级产品对应其次级产品的数量
t = [];

% 初始化
lc = [];
n1 = NaN;
n2 = zeros(3,9,2);
num = zeros(3,9,2);
k = ceil(power(n,1/m));% 每个本级产品对应其次级产品的数量
t = [];
clc
% 各零配件半成品成品检验成本
ic1 = [1, 1, 2, 1, 1, 2, 1, 2, 0];
ic2 = [4, 4, 4];
ic3 = 6;
ic = A + InputParameter(ic1,ic2,ic3);

% 各零配件半成品成品次品率
fd1 = [0.10, 0.10, 0.10, 0.10, 0.10, 0.10, 0.10, 0.10, 0];
fd2 = [0.10, 0.10, 0.10];
fd3 = [0.10];
fd = A + InputParameter(fd1,fd2,fd3);
td = ones(1,n) - fd; % 合格率

% 各零配件半成品成品购买单价/装配成本
c1 = [2, 8, 12, 2, 8, 12, 8, 12, 0];
cF2 = [8, 8, 8];
cF3 = 8;
c = A + InputParameter(c1,cF2,cF3);

% 各半成品/成品的拆解费用
dc1 = 0;
dc2 = 6;
dc3 = 10;
dc = A + InputParameter(dc1,dc2,dc3);

```



```
[C,lc,t,nfp] = CostMinimize7(num0,m,n,fd,c,ic,dc,rl,td,k);
best_decision = profit(C,lc,t,nfp,p);
```

附录 15

第三问求解代码（InputParameter.m）

```
% 输入参数
function A = InputParameter(a1,a2,a3)
maxLength = max([length(a1), length(a2), length(a3)]);
a20 = [a2, zeros(1, maxLength - length(a2))];
a30 = [a3, zeros(1, maxLength - length(a3))];
A = [a1;a20;a30];
end
```

附录 16

第三问求解代码（profit.m）

```
function best_decision = profit(C,lc,t,nfp,p)
P0 = -Inf;
T0 = -Inf;
for a1 = 0:1
    for a2 = 0:1
        for a3 = 0:1
            for b2 = 0:1
                for b3 = 0:1
                    T = C(1,:,a1+1) + lc(1,:,a1+1) + C(2,:,a2+1) + lc(2,:,a2+1) +
t(2,:,b2+1) + C(3,:,a3+1) + lc(3,:,a3+1) + t(3,:,b3+1);
                    if T < T0
                        T0 = T;
                        P = nfp*p - T;
                        best_decision = [a1,a2,a3,b1,b2,P];
                        disp(a1,a2,a3,b1,b2);
```



```

        else
            rate = rate * (1 - (fd(i-1, idx) * (1 - D(i-1, idx, a2+1))));
        end
    end
    rate = rate * (1 - fd(i,j)); % 计算成品的合格率

end
end

```

附录 19

第四问求解代码（tt4.m）

```

function [estimated_p, CI] = tt4(true_p, alpha)
% 模拟生成零件并估计次品率
%
% 输入:
% true_p - 真实的次品率（固定但未知）
% alpha - 显著性水平，用于计算置信区间（1 - 置信度）
%
% 输出:
% estimated_p - 估计的次品率
% CI - 次品率的 Clopper-Pearson 置信区间

% 初始化变量
n = 0; % 已检验的零件数量
k = 0; % 检验出的次品数量
LowerBound = 0; % 置信区间的下限
UpperBound = 1; % 置信区间的上限
CI = [0, 1]; % 初始化置信区间

% 序贯抽样过程
while true
    % 生成一个新的零件，检查是否为次品
    if rand() < true_p
        k = k + 1;
    end
    n = n + 1; % 总检查计数增加

    % 计算当前样本的次品率估计值

```

```

        p_hat = k / n;

        % 计算 Clopper-Pearson 置信区间
        [LowerBound, UpperBound] = clopper_pearson(k, n, alpha);

        % 更新置信区间
        CI = [LowerBound, UpperBound];

        % 检查置信区间宽度
        if UpperBound - LowerBound < 0.01
            break;
        end
    end

    % 设置估计的次品率
    estimated_p = p_hat;
end

function [lower, upper] = clopper_pearson(k, n, alpha)
    % 计算 Clopper-Pearson 置信区间
    % k - 次品数量
    % n - 检查的零件总数
    % alpha - 显著性水平

    % 计算分位数
    a = alpha / 2;
    b = 1 - a;

    % 计算置信区间的下限和上限
    lower = betainv(a, k + 1, n - k + 1);
    upper = betainv(b, k, n - k);
end

```

附录 20

第四问求解代码（t4.m）

```

clear;clc
disp('情况 1: ');
A1 = CostMinimize(1000,1000,0.09,4,2,0.08,18,3,0.09,6,3,56,6,5);

```

```
disp('情况 2: ');
A2 = CostMinimize(1000,1000,0.09,4,2,0.08,18,3,0.19,6,3,56,6,5);
disp('情况 3: ');
A3 = CostMinimize(1000,1000,0.09,4,2,0.08,18,3,0.09,6,3,56,30,5);
disp('情况 4: ');
A4 = CostMinimize(1000,1000,0.19,4,1,0.18,18,1,0.19,6,2,56,30,5);
disp('情况 5: ');
A5 = CostMinimize(1000,1000,0.09,4,8,0.18,18,1,0.09,6,2,56,10,5);
disp('情况 6: ');
A6 = CostMinimize(1000,1000,0.045,4,2,0.045,18,3,0.045,6,3,56,10,40);
A=[A1;A2;A3;A4;A5;A6];
```

附录 21																	
第三问图表																	
a	a	b	a	b	c	i	c	i	t	c	i	t3(fi(n	T	P	退	P_FI
1	2	2	3	3	1	c	2	c	2	3	c	仅拆解)	fp)			换	NAL
						1		2			3					货	
0	0	0	0	0	6	0	2	0	0	8	0	0	0.3	9	56	28	-68
					4		4							6	.5	.7	
0	0	0	0	1	6	0	2	0	0	8	0	7.2	0.5	1		20	-27
					4		4							0	97	.6	
0	0	0	1	0	6	0	2	0	0	8	6	0	0.3	1	56	0	-46
					4		4							0	.5		
0	0	0	1	1	6	0	2	0	0	8	6	7.2	0.5	1		0	-12
					4		4							0	97	0	
0	0	1	0	0	6	0	2	0	0	8	0	0	0.3	9	56	28	-68
					4		4							6	.5	.7	
0	0	1	0	1	6	0	2	0	0	8	0	7.2	0.5	1		20	-27
					4		4							0	97	.6	

														3					
														1					
0	0	1	1	0	6	0	2	0	0	8	6	0	0.3	0	56	0	-46		
					4		4							2	.5				
														1					
0	0	1	1	1	6	0	2	0	0	8	6	7.2	0.5	0	97	0	-12		
					4		4							9					
														1					
0	1	0	0	0	6	0	2	1		3				0	11	2.	12.5		
					4		4	2	0	.	0	0	0.6	0	8	13			
										4				3					
										3				1					
0	1	0	0	1	6	0	2	1	0	.	0	0.7	0.7	0	13	2.	25		
					4		4	2		4				4	1	13			
														1					
0	1	0	1	0	6	0	2	1	0	.	.	0	0.6	0	11		10.7		
					4		4	2		4	9			7	8	0			
										3	3			1					
0	1	0	1	1	6	0	2	1	0	.	.	0.7	0.7	0	13		23.2		
					4		4	2		4	9			8	1	0			
														1					
0	1	1	0	0	6	0	2	1		6	3			1	11	2.	6.34		
					4		4	2	.	.	0	0	0.6	1	8	13			
									2	4				0					
									6	3				1					
0	1	1	0	1	6	0	2	1	.	.	0	0.7	0.7	1	13	2.	18.8		
					4		4	2	2	4				0	1	13			
														1					
0	1	1	1	0	6	0	2	1	.	.	.	0	0.6	1	11		4.53		
					4		4	2	2	4	9			4	8	0			
									6	3	3			1					
0	1	1	1	1	6	0	2	1	.	.	.	0.7	0.7	1	13		17		
					4		4	2	2	4	9			4	1	0			
														1					
1	0	0	0	0	6	1	2	0	0	.	0	0	0.6	0	11	12	1.92		
					4	1	2			2				4	8	.4			
1	0	0	0	1	6	1	2	0	0	7	0	5.2	0.6	1	13	12	8.51		

					4	1	2			.				0	0	.4			
										2				9					
										7	2			1					
1	0	0	1	0	6	1	2	0	0	.	.	0	0.6	0	11	0	11.4		
					4	1	2			2	9			7	8				
										7	2			1					
1	0	0	1	1	6	1	2	0	0	.	.	5.2	0.6	1	13	0	18		
					4	1	2			2	9			2	0				
										7				1					
1	0	1	0	0	6	1	2	0	0	.	0	0	0.6	0	11	12	1.92		
					4	1	2			2				4	8	.4			
										7				1					
1	0	1	0	1	6	1	2	0	0	.	0	5.2	0.6	0	13	12	8.51		
					4	1	2			2				9	0	.4			
										7	2			1					
1	0	1	1	0	6	1	2	0	0	.	.	0	0.6	0	11	0	11.4		
					4	1	2			2	9			7	8				
										7	2			1					
1	0	1	1	1	6	1	2	0	0	.	.	5.2	0.6	1	13	0	18		
					4	1	2			2	9			2	0				
										6				1					
1	1	0	0	0	6	1	2	1	0	.	0	0	0.7	1	14	3.	28.7		
					4	1	2	1		5				4	6	24			
										6				1					
1	1	0	0	1	6	1	2	1	0	.	0	0.9	0.8	1	16	3.	42.4		
					4	1	2	1		5				5	0	24			
										6	4			1					
1	1	0	1	0	6	1	2	1	0	.	.	0	0.7	1	14	0	27.1		
					4	1	2	1		5	9			9	6				
										6	4			1					
1	1	0	1	1	6	1	2	1	0	.	.	0.9	0.8	2	16	0	40.7		
					4	1	2	1		5	9			0	0				
									1	6				1					
1	1	1	0	0	6	1	2	1	.	.	0	0	0.7	1	14	3.	27.1		
					4	1	2	1	6	5				6	6	24			

1	1	1	0	1	6	1	2	1	1	6				1	16	3.		
					4	1	2	1		.	.	0	0.9	0.8	1	0	24	40.7
									6	5					6			
1	1	1	1	0	6	1	2	1	1	6	4				1	14		
					4	1	2	1		.	.	.	0	0.7	2	6	0	25.4
									6	5	9				0			
1	1	1	1	1	6	1	2	1	1	6	4				1	16		
					4	1	2	1		.	.	.	0.9	0.8	2	0	0	39.1
									6	5	9				1			

附录 22
基于 xxxx 的第一问求解代码

附录 23
基于 xxxx 的第一问求解代码

附录 24

基于 xxxx 的第一问求解代码