

# 题目 猪场连续生产销售计划

## 摘要:

文根据猪场种猪配种信息、猪只生产全过程、猪场饲料投喂程序以及猪舍约束等条件,对猪的各个阶段的状态进行分析,建立了猪只生长、副产品销售的数学模型,以及确定了生产母猪和生产公猪的更新策略,同时考虑了各品种的市价、车辆运载能力、产能约束等条件,制定了猪只的配种计划,解决了猪场连续生产带来的难题。

针对问题一,首先分析猪只的生长数据,对相同日龄猪只的质量求和之后再利用 Logistic 曲线对猪只的生长情况进行拟合,最终建立猪只的日龄与质量的生长曲线模型,即 $y = 135.70369 + (5.1932 - 135.70369) / [1 + (\frac{x}{123.274})^{2.83968}]$ 。其次确定生产公猪、生产母猪的更新策略。通过对附件 6—猪只存栏信息以及公猪配种条件的分析,得到公猪可用生产公猪的时间为 43 周—52 周,生产公猪的年更新率为 100%,且现状态生产公猪的最大日龄为 560 天。本文采用“预留更新法”选出的公猪作为后备公猪,将当前生产公猪最大日龄作为选取时间,即提前 15 周选取同品系日龄在 28 周以上且质量小于 120kg 的公猪作为备用生产公猪,并在生产公猪的使用到达一年时进行更新,这样就得到生产公猪的更新计划。生产母猪的更新概率为 65%,本文采用日龄大优先的原则更新生产母猪,即从备用母猪中选取同品系的母猪,同时更新备选母猪。最后再结合生产母猪、生产公猪更新计划以及种猪销售计划,制定了猪只的配种计划。

问题二是典型的多目标多约束优化问题,首先分析了车辆的运载能力,以及以采用优先考虑价高月份的价格为策略确定了每周副产品的价格,最后再分析猪舍的产能约束,提出了系列约束条件。综合考虑各个因素,建立以连副产品收益最大为目标函数的数学模型,并利用遗传算法进行求解。通过遗传算法的迭代,最终得到了一个较优的副产品销售计划。

针对问题三,本文采用时间驱动的方法对应饲养程序依次每周的每周的饲料消耗量。

关键词: 遗传算法 多约束规划 拟合曲线 时间驱动 种群结构

# 1 问题重述

## 1.1 问题的背景

自 80 年代猪的流通体制改革以来,养猪业走向市场,从而具备了市场经济的发展特征,生产量和销售价格呈周期性变化,猪的出栏时间处在价格高峰期,养猪效益显著,处在低谷期,养猪无利甚至亏本。尤其规模化养猪场,受其影响愈深。

并且自 2018 年 8 月,中国爆发了非洲猪瘟疫情,截至 2019 年 7 月底,全国已有 31 个省份城市发生非洲猪瘟疫情 150 起,共扑杀生猪 116 万头。非洲猪瘟传入我国的风险日益加大。对养殖场而言一旦爆发非洲猪瘟,将面临不小的损失,造成的猪价市场波动,也会给百姓的“菜篮子”添加重担[1]。所以研究猪的生产周期性变化规律,对制定猪的生产计划、指导养猪场进行科学管理,都具有重要的现实意义。

就一个养殖场的发展而言猪群结构是非常重要的指规模猪场杂交组合各种猪群的组成、数量和质量,不同的杂交方式,有相应不同的种群组合,数量亦有差异,特别是猪种的质量要与生产指标的要求相适应,任何一个规模猪场都应该重视猪群结构的优化与种猪的更新。其中猪群更新最主要的方式就是配种,掌握和控制猪只的生产规律制定连续的生产计划、销售计划、更新等,以猪舍的容量,运输以及各项成本为约束,充分利用现有资源使利润最大化。

## 1.2 要解决的问题

问题一:制定每周配种计划,及生产公猪、生产母猪的更新计划。

问题二:制定四种副产品每周的销售计划。

问题三:测算每周各类饲料消耗量。

# 2 模型假设

假设 1:假设猪舍中的每个栏位只养相同重量级的猪,不同重量级的猪只不混养。

假设 2:假设 A,B,C,D 系母猪每胎平均产仔 10 只。

假设 3:根据附件 6 猪只状态推断假设种猪出售在保育期后。

假设 4:假设体重超过 120Kg 的猪也在育肥舍饲养,育肥舍每栏可以饲养 $\geq 120\text{kg}$ 的猪只约 23 头。

假设 5:假设附件 6 所有猪只都未怀孕

假设 6:假设当某周横跨两个月份时该周出售的副产品的价格取两个月中较高的那个价格。

### 3 符号说明

符号	说明
$W_d$	平均日龄体重
$R_i$	产房类型 (i=1, 2, 3)
$T_p$	全年总盈利
$Income_z$	全年出售种猪总收入
$Income_f$	全年出售副产品总收入
$F_{cost}$	全年总饲料成本
$F_{ij}$	编号为 <i>i</i> 的猪在 2019 年第 <i>j</i> 天的累计饲料成本
$T_{cost}$	全年总运输成本
$q_{ij}$	第 <i>i</i> 种副产品在第 <i>j</i> 周的销售量 $i = 1, 2, 3, 4; j = 1, 2, 3 \dots 52$
$t_{j1}$	第 <i>j</i> 周车 1 的使用次数
$t_{j2}$	第 <i>j</i> 周车 2 的使用次数
$d_{ij}$	编号为 <i>i</i> 的猪在 2019 年第 <i>j</i> 天的日龄

### 4 问题建模

本题的优化目标就是在满足各项约束的情况下使 2019 年的收益最大化。根据题意，因为题目中没有提及到其他收入或成本项，所以目前我们只考虑全年种猪收入，副产品收入，饲料成本和运输成本。该模型的收益目标函数为  $Max T_p$ ，如下式所示。

$$Max T_p = Income_z + Income_f - F_{cost} - T_{cost}$$

#### 4.1 生长规律建模

对已有附件 2 生长数据分析后我们对生长曲线的拟合，由于生长数据的不规范和一些缺失我们对已有的数据进行处理，将附件 2 日龄与体重关系数据绘制成散点图如下图 1-a 所示，可见前期一百天以下的数据较为零散没有办法进行很好的拟合。我们简单先处理数据，对同一日龄的体重求平均值  $\bar{W}_d$ ，绘制出日龄—平均体重散点图如图 1-b 所示。可以看到 1-a 图 150-200 日龄的数据分布很散，求平均后从 1-b 图可以看出猪的生长曲线的趋势性还是比较明显的。

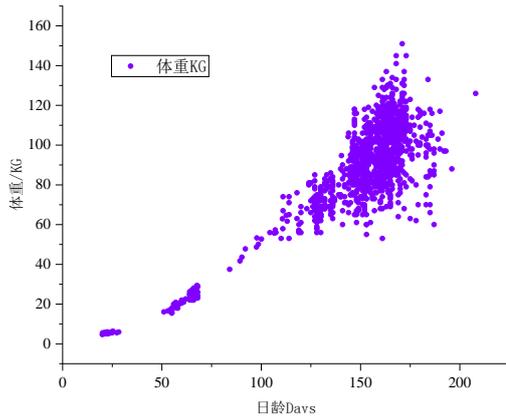


图 1-a 日龄—体重散点图

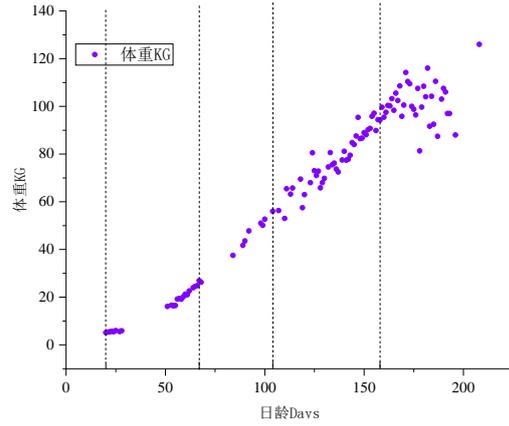


图 1-b 日龄—平均体重 $\bar{W}_d$ 散点图

我们对猪的生长曲线进行拟合，采用 Logistic 曲线拟合，表 1 中是我们在拟合过程中的一些参数设置。如下图 2 中的红色曲线所示，日龄为自变量  $x$ ，体重  $y$  为因变量，拟合曲线的各个参数如下表所示。最终的拟合函数：

$$y = 135.70369 + (5.1932 - 135.70369) / [1 + \left(\frac{x}{123.274}\right)^{2.83968}] \quad (1)$$

整体的拟合度  $R^2=0.96751$ 。这个拟合程度还是比较高的，使用这个生长曲线较好解决了部分生长数据缺失的问题。在下面的解题过程中我们都将采用该拟合函数来计算猪只日龄对应的体重。

表 1 Logistic 曲线拟合参数

模型	Logistic
方程	$y = A_1 + (A_1 - A_2) / [1 + \left(\frac{x}{X_0}\right)^P]$
$A_1$	$5.1932 \pm 2.45041$
$A_2$	$135.70369 \pm 9.05724$
$X_0$	$123.274 \pm 6.1434$
$P$	$2.83968 \pm 0.30027$
Reduced Chi-Sqr	36.68817
$R^2$ (COD)	0.96841
调整后 $R^2$	0.96751

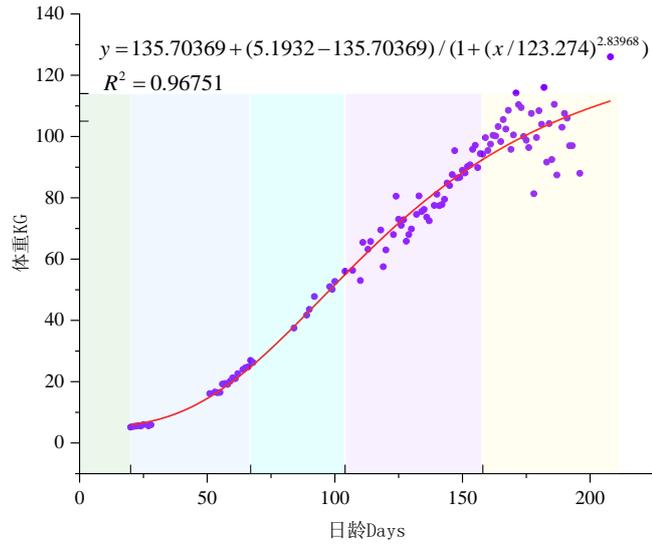


图 2 日龄—体重 Logistic 曲线拟合

#### 4.2 猪只种群建模

基于猪品种的我们进行建模,如下 $all$ 矩阵表示该养猪场所有猪的集合,按照品系和公母分为 12 个大类。如下 $A_{boar}$ 表示所有 A 品系公猪的集合, $A_{sow}$ 表示所有 A 系母猪的集合。

$$all = \begin{bmatrix} A_{boar} & B_{boar} & C_{boar} & D_{boar} & E_{boar} & F_{boar} \\ A_{sow} & B_{sow} & C_{sow} & D_{sow} & E_{sow} & F_{sow} \end{bmatrix} \quad (2)$$

结合猪的体重和生长阶段(仔猪期,保育期,育肥期)又将每个同系同性大类划分为处于六个不同生长状态猪的集合,六个状态分别满足如下约束:

状态一的猪满足条件 1. 处于哺乳期的仔猪。

状态二的猪满足条件 1. 处于保育阶段; 2. 体重在 6-22.7kg 间; 3. 在保育舍饲养。

状态三的猪满足条件 1. 处于保育阶段; 2. 体重在 22.7-25kg 间; 3. 在保育舍饲养。

状态四的猪满足条件 1. 处于保育阶段(10 周龄以下); 2. 体重超过 25kg; 3. 在育肥舍饲养。

状态五的猪满足条件 1. 处于育肥阶段(10 周龄以上); 2. 体重在 22.7-34kg 之间; 3. 在育肥舍饲养。

状态六的猪满足条件 1. 体重在 34kg 以上; 2. 在育肥舍饲养。

如 $A_{boar}\{a_{b1} a_{b2} a_{b3} a_{b4} a_{b5} a_{b6}\}$ 表示所有 A 系公猪的集合, $a_{b1}$ 表示所有 A 系公仔猪, $a_{b6}$ 表示所有 A 系公猪中所有处于状态六的猪只。其它系猪只集合类似,这里不再赘述。

$$A \text{ 系公猪集合: } A_{boar} = \{a_{b1} \ a_{b2} \ a_{b3} \ a_{b4} \ a_{b5} \ a_{b6}\}$$

$$A \text{ 系母猪集合: } A_{sow} = \{a_{s1} \ a_{s2} \ a_{s3} \ a_{s4} \ a_{s5} \ a_{s6}\} \quad (3)$$

如下表 c 所示是对附件 6 所给初始状态按照以上我们定义的六个生长状态、种猪和副产品以及品系进行归类汇总后得到的结果。如处于状态一也就是仔猪阶段的 A 品系种猪共 120 只。

表 2 附件 6 生长状态统计表

	品系	生长状态					
		状态一	状态二	状态三	状态四	状态五	状态六
种猪	A	120	260	9	67	9	463
	B	128	206	28	63	5	1127
	C	5	8	0	18	10	83
	D	76	79	34	50	0	439
	E	104	171	15	28	22	350
	F	211	553	28	62	15	490
副产品	A	164	176	22	38	11	872
	B	152	180	19	36	1	841
	C	84	98	8	22	6	332
	D	78	74	8	22	0	406
	E	83	88	11	25	0	303
	F	33	32	3	7	0	129

自留种猪建模：如下三个集合分别表示生产公猪，生产母猪，后备母猪。具体的  $P_{boar}$  表示养猪场中所有生产公种猪组成的种群， $P_{boar}$  中的  $a$  表示所有 A 系生产公猪， $b$  表示所有 B 系生产公猪，其他类似。 $P_{sow}$  表示所有生产母猪组成的集合， $a$  表示所有 A 系生产母猪， $b$  表示所有 B 系生产母猪， $c, d$  类似。 $R_{sow}$  表示所有后备母猪组成的集合， $a$  表示所有 A 系后备母猪， $b$  表示所有 B 系后备母猪， $c, d$  同理。

$$\begin{aligned}
 P_{boar} &= \{a \ b \ c \ d\} \\
 P_{sow} &= \{a \ b \ c \ d\} \\
 R_{sow} &= \{a \ b \ c \ d\}
 \end{aligned}
 \tag{4}$$

### 4.3 产能约束

根据题意知该养猪场共 8 间产房，每间产房 24 个产床。每头母猪在产房里待 24 天。由于房冲洗、干燥、维修、消毒等操作约需 5 天周转，故可正常使用的产房只有 6 间。即产房内正常母猪存栏总数不超过  $24 \times 6$  头。14 间保育舍，每间 12 个栏位，每个栏位最多养 6-22.7kg 的猪只约 24 头，或者 22.7-25kg 的猪只约 18 头。又根据附件 3 的实际面积等保育房每间实际容纳量与理论容纳量不同，具体理论与实际容纳量如下表。14 间育肥舍，每间 24 个栏位，每个栏最多养 22.7-34kg 的猪只约 46 头，或者 34-120kg 的猪只约 23 头。具体容量如下表 3。

表 3 养猪场的容纳情况

猪舍类型 ( $R_i$ )	间数	每间栏位	每栏容纳量	每间理论容纳量/头	每间实际容纳量/头
产房	8 间	24	1	24	24
保育舍	14 间	12	24 (6-22.7kg)	288	289
			18 (22.7-25kg)	216	221
育肥舍	14 间	24	46 (22.7-34kg)	1104	1104
			23 (34-120kg)	552	486

结合附件 6 现有的状态,  $x_i, i = 1,2,3,4,5,6$  分别代表 A-F 系的仔猪数量。  $x_7$  表示体重 6-22.7kg 猪的数量。  $x_8$  表示体重 22.7-25kg 猪的数量。  $x_9$  表示体重 22.7-34kg 猪的数量。  $x_{10}$  表示体重 34-120kg 猪的数量。  $x_i$  的具体计算过程如下:

$$\sum_{i=1}^4 x_i = a_{b1} + a_{s1} + b_{b1} + b_{s1} + c_{b1} + c_{s1} + d_{b1} + d_{s1} + e_{b1} + e_{s1} + f_{b1} + f_{s1} \quad (5)$$

$$x_5 = e_{b1} + e_{s1} \quad (6)$$

$$x_6 = f_{b1} + f_{s1} \quad (7)$$

$$x_7 = a_{b2} + a_{s2} + b_{b2} + b_{s2} + c_{b2} + c_{s2} + d_{b2} + d_{s2} + e_{b2} + e_{s2} + f_{b2} + f_{s2} \quad (8)$$

$$x_8 = a_{b3} + a_{s3} + b_{b3} + b_{s3} + c_{b3} + c_{s3} + d_{b3} + d_{s3} + e_{b3} + e_{s3} + f_{b3} + f_{s3} \quad (9)$$

$$x_9 = a_{b4} + a_{s4} + b_{b4} + b_{s4} + c_{b4} + c_{s4} + d_{b4} + d_{s4} + e_{b4} + e_{s4} + f_{b4} + f_{s4} \quad (10)$$

$$+ a_{b5} + a_{s5} + b_{b5} + b_{s5} + c_{b5} + c_{s5} + d_{b5} + d_{s5} + e_{b5} + e_{s5} + f_{b5} + f_{s5} \quad (11)$$

$$x_{10} = a_{b6} + a_{s6} + b_{b6} + b_{s6} + c_{b6} + c_{s6} + d_{b6} + d_{s6} + e_{b6} + e_{s6} + f_{b6} + f_{s6} \quad (12)$$

综合考虑猪场每栏实际容量和现有猪只情况得出如下约束。以下四个公式以猪舍为约束, 式 (13) 表示猪只存栏总数不低于 9000 头。式 (14) 表示在产房中的正常母猪存栏总数不超过 24\*6 头。式 (15) 表示在保育舍中体重 6-22.7kg、22.7-25kg 的猪数量不超过保育舍容纳量, 公式中计算时采用 E 系仔猪每胎 18 只, F 系仔猪每胎 22 头, 若取下限都可以满足约束则取上限必然满足。式 (16) 表示体重 22.7-34kg 的猪、34-120kg 的猪减去在产房中母猪后的数量不超过 14 间育肥舍的容纳量。  $ceil$  表示相除后向上取整, 如计算某重量级的猪占 2.5 个猪舍, 则向上取整后占 3 个猪舍。

$$\sum_{i=1}^{10} x_i \geq 9000 \quad (13)$$

$$\frac{\sum_{i=1}^4 x_i}{10} + \frac{x_5}{18} + \frac{x_6}{22} \leq 6 \times 24 \quad (14)$$

$$ceil\left(\frac{x_7}{289}\right) + ceil\left(\frac{x_8}{221}\right) \leq 14 \quad (15)$$

$$ceil\left(\frac{x_9}{1104}\right) + ceil\left(\frac{x_{10} - \left(\frac{\sum_{i=1}^4 x_i}{10} + \frac{x_5}{18} + \frac{x_6}{22}\right)}{486}\right) \leq 14 \quad (16)$$

#### 4.4 生产公/母猪更新策略

生产公猪更新策略:

生产公猪每年 100%更新。因为公猪长至 43 周龄时可以开始采精使用, 我们原计划在第 43 周时筛选生产公猪但是发现根据我们的生长曲线计算得在第 43 周时猪只的体重必然超过了 120Kg, 由于我们的副产品销售时的最大体重约束猪只体重  $\leq 120$ Kg, 以及公种猪销售体重约束猪只体重  $\leq 114$ Kg, 我们在销售时优先选体重大的猪只优先出售, 因此在第 43 周再选生产种猪取精时无法得到生产公猪, 所以我们采用预定法选择生产公猪。自留公种猪在 43 周取精, 第一次取精后就代表此猪只成为生产公猪, 使用一年 (52 周) 后更替掉。目前生产公猪中日龄最大的公猪为 560 日龄, 该公猪还有 112 天被淘汰。为了有猪更替这只日龄最大的生产公猪, 我们需要在其同系的公猪中找一只至多 112 后到达 43 周龄的猪备用, 所以这只备用公猪的日龄必然要大于 196 日龄。当生产公猪中有使用达到一年的就淘汰并在备用公猪中选一只来替换。

预留更新法: 如下纯系公猪生命周期示意图所示, 因为目前附件 6 所给的最大生产公猪日龄为 560 天, 也就是 80 周。为了保证生产公猪的接替不间断, 我们至少

要在 196 日龄后的公猪中选后备预留公猪。同时保证体重在 120kg 以下，根据日龄—体重曲线计算得 120kg 的猪只对应日龄为 284 天左右。所以我们将自留后备纯系公种猪的选种时间定在 196-248 日龄之间。用预定更新法选出来的公猪我们称之为备用公猪。

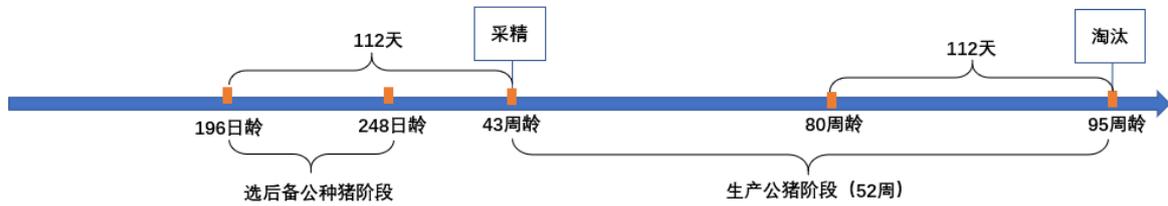


图 3 纯系公猪生命周期示意图

生产母猪更新策略：

备用母猪的预选在母猪到达 21 周时进行，由于副产品四的单价在五月份最高所以我们尽量将生产母猪在五月份更新。结合考虑生产母猪从第一次配种起，2 年内可不更新，使用 2 年以上可考虑更新替换，通常每年更新替换率约 65%。我们给出的更新策略为：

- (1) 每年替换掉使用超过两年的这部分生产母猪的 65%。
- (2) 更新时从日龄大的开始，日龄大的先更新。
- (3) 尽量在五月中集中更新。

#### 4.5 运输能力约束

原有的车辆的装载能力如下表所示，下表中只有三个重量级，与四种副产品的重量不完全匹配。又因为不同体重和品种的猪可以混装所以我们对两种车辆的装载能力  $C_1$ ， $C_2$  做了如下简化：

表 4 养猪场的容纳情况

Kg/头	100kg	50kg	15kg	单次运输成本/元
最高运载头数				
车 1 (C1)	102	200	600	550
车 2 (C2)	45	100	400	500

平均装载能力：

$$C_1 = \frac{100 \times 102 + 50 \times 200 + 15 \times 600}{3} = 9733.3 \quad (17)$$

$$C_2 = \frac{100 \times 45 + 50 \times 100 + 15 \times 400}{3} = 5166.7 \quad (18)$$

由装载量与单次运输成本计算得每公斤运输成本，车 1 的每公斤运输成本约为 0.057，车 2 的每公斤运输成本约为 0.097。根据每周销售的副产品总重量  $q_j$  选择装载方案。如下式 (19) 所示，用第  $j$  周的副产品总重量除以车一的装载能力，商为  $m$ ，余数为  $n$ 。我们根据余数和商的不同情况进行讨论，当  $0 < n \leq 600.1$  时用两辆车 2

可以装下，这样比用一辆车 1 一辆车 2 更便宜。下表为车辆选择策略，未在表中的情况都使用 3 辆车 1，3 辆车 2。总的全年运输成本如式（20）所示。

$$\frac{q_j}{c_1} = m \cdots n \quad (19)$$

$$T_{cost} = \sum_{j=1}^{52} 550 \times t_{j1} + 500 \times t_{j2} \quad (20)$$

表 5 每周装载方式组合

余数n的值	m的值	车 1 周使用次数 ( $t_{j1}$ )	车 2 周使用次数 ( $t_{j2}$ )
$0 < n \leq 600.1$	0	0	1
	1	0	2
	2	1	2
	3	2	2
	4	3	2
$600.1 < n \leq C_2$	0	0	1
	1	1	1
	2	2	1
	3	3	1
	4	3	3
$n > C_2$	0	1	0
	1	2	0
	2	3	0
	3	3	2
其他情况		3	3

#### 4.6 饲料成本

猪场按照标准饲喂程序对不同猪只投喂饲料。因为生产母猪及生产公猪耗用的饲料 1、饲料 2，成本摊至出生的小猪。2. 其它品种饲料按饲喂天数、日消耗、饲料单价计算。如下式  $f$  表示除去生产公猪和生产母猪所有剩下的目前在养殖场中的猪只数量。 $F_{i1}$  表示编号为  $i$  的猪在 2019 年第一天其日龄所对应的饲料成本， $F_{ij}$  表示  $i$  这只猪在 2019 年第  $j$  天其日龄所对应的饲料成本，该猪只在第  $j$  天离开该养猪场。全年饲料成本就等于在该养猪场厂出现过的每只猪只产生的饲料成本之和，如下式（21）。

$$F_{cost} = \sum(F_{ij} - F_{i1}) \quad (21)$$

#### 4.7 副产品构成及其销售收入

副产品通常分 4 个品种对外销售：副产品一销售体重约  $\leq 20\text{kg}$ ，副产品二约  $25-50\text{kg}$ ，副产品三约  $50-100\text{kg}$ ，副产品四约  $\geq 100\text{kg}$  且  $< 120\text{kg}$ ，并且我们将淘汰的生产公/母猪也归类为第四类副产品。

表 6 副产品类型表

副产品类型	体重/Kg
副产品一	≤20
副产品二	25-50
副产品三	50-100
副产品四	100-120, 淘汰的生产公/母猪

我们归纳不同时期每种副产品可出售的量的范围如下，我们根据生产公/母猪的更新计划可以预先得知哪些猪只是用于替换生产公/母猪的，这类猪只做为养猪场自己使用的配种猪只始终不会被做为副产品出售。将作为销售种猪被出售的猪只也可以提前预知，这类猪只也不会作为副产品出售。具体的副产品上限计算方式见式（25）-（27）。

副产品一：

出售上限：所有体重≤20kg 的猪只—将来会作为销售种猪出售的猪只—将来用于配种的猪只。

出售下限：从产房向保育舍转移时保育舍容纳不下的。

副产品二：

出售上限：所有体重 25-50kg 的猪只—将来会作为销售种猪出售的—将来用于配种的猪只。

出售下限：从保育舍向育肥舍转移时育肥舍容纳不下的。

副产品三：

出售上限：所有体重 50-100kg 的猪只—将来会作为销售种猪出售的—将来用于配种的猪只。

副产品四：

出售上限：所有体重 100-120kg 的猪只—作为销售种猪出售的猪只—将来用于配种的猪只+淘汰后的生产公/母猪

出售下限：淘汰后的生产公猪+淘汰后的生产母猪

副产品销售收入：

副产品分 4 个品种对外销售：副产品一销售体重约≤20kg，副产品二约 25-50kg（20kg-25kg 的猪不能销售），副产品三约 50-100kg，副产品四约≥100kg 且<120kg。根据题意由于更新替换掉的生产猪也可以作为副产品出售，我们将这类副产品的价格定为与副产品四相同的价格。由于副产品的销售价格随季节波动，由于车辆按周调度所以我们副产品的销售计划按周制定，4 种副产品，全年共 52 周。所以副产品价格矩阵  $bpp$  是一个 4\*52 的矩阵。

$$b_{pp} = \begin{bmatrix} 16.45 & 16.45 & 16.45 & 16.45 & 17.60 & 17.60 & \dots & 16.27 \\ 14.14 & 14.14 & 14.14 & 14.14 & 14.62 & 14.62 & \dots & 14.15 \\ 15.83 & 15.83 & 15.83 & 15.83 & 16.00 & 16.00 & \dots & 15.95 \\ 54.18 & 54.18 & 54.18 & 54.18 & 55.20 & 55.20 & \dots & 47.46 \end{bmatrix} \quad (22)$$

副产品销售量  $b_{pq}$  也用矩阵表示，也是一个 4\*52 的矩阵，表示 4 个不同种类的副产品分别在一年中 52 周每周的销售量。如下矩阵中  $q_{34}$  就表示第三类副产品在第 4 周的销售量。全年总的副产品销售收入如下式（24）所示

$$bpq = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} & q_{15} & q_{16} & \cdots & q_{152} \\ q_{21} & q_{22} & q_{23} & q_{24} & q_{25} & q_{26} & \cdots & q_{252} \\ q_{31} & q_{32} & q_{33} & q_{34} & q_{35} & q_{36} & \cdots & q_{352} \\ q_{41} & q_{42} & q_{43} & q_{44} & q_{45} & q_{46} & \cdots & q_{452} \end{bmatrix} \quad (23)$$

$$Income_f = \sum b_{pp} \cdot bpq \quad (24)$$

#### 4.8 销售种猪盈利

种猪的销售根据附件 7 月销售计划来制定，如下表 种猪盈利是一个固定值，如上表所示全年种猪总收入  $Income_z = 5707200$  元。

表 7 不同品系种猪销售收入表

种猪系列	年需求量/只	种猪收入/元
A 系公猪	196	245000
C 系公猪	83	103750
E 系公猪	841	1051250
B 系母猪	300	480000
D 系母猪	220	352000
F 系母猪	2,172	3475200
总量统计	3,812	5707200

## 5 问题分析与求解

### 5.1 问题一分析与求解

问题一需要制定每周配种计划，及生产公猪、生产母猪的更新计划，根据对题目的分析我们采用上述模型建立中生产公猪和生产母猪更新策略。由于副产品四的单价在五月份最高所以我们尽量将生产母猪在五月份更新。结合考虑生产母猪从第一次配种起，2 年内可不更新，使用 2 年以上可考虑更新替换，通常每年更新替换率约 65%。我组给出的更新策略为：

- 1、每年替换掉使用超过两年的这部分生产母猪的 65%。
- 2、更新时从日龄大的开始，日龄大的先更新。
- 3、尽量在五月份集中更新。

表 8 是每周具体的替换类型，从第 31 周开始替换，如下表第一行表示第 31 周替换掉 30 头 B 系生产母猪和 20 头 D 系生产母猪。生产母猪具体的替换方案如下表 9 所示，表中生产母猪 ID 代表将被淘汰掉的生产母猪，对应的替换母猪 ID 代表将用于替换的母猪编号。如表 9 中第一组表示将用 615 号母猪代替 3513 号母猪。表中的 0 表示该母猪是 19 年新出生的猪，在附件 6 中没有编号所以我们用 0 替代。

表 8 生产母猪替换周计划

替换周数	替换类型 1	替换类型 2
31	30B	20D
35	52A	30B
36	30B	
37	30B	
39	30B	
41	30B	
43	30B	
45	4B	

表 9 生产母猪具体替换方案

生产母猪 ID	替换母猪 ID						
3513	615	1866	0	1822	0	5781	0
152	5435	5628	917	351	0	5300	0
4202	4225	1885	1629	3851	0	2561	0
3848	2936	3221	0	760	0	2155	0
1658	3957	802	0	4101	0	5234	0
1814	2471	3447	0	4841	0	4913	0
2189	5025	2764	0	858	0	4581	0
1006	5370	143	0	1578	0	3891	0
5754	424	4620	0	3085	0	3839	0
5439	2097	5201	0	2790	0	3598	0
2909	2044	1416	0	1573	0	3326	0
3685	4572	3228	4579	6089	0	3255	0
3662	3618	3172	1086	4960	0	2674	0
3162	3031	2612	6082	2955	0	2307	0
3009	2711	2242	5759	2604	0	1043	0
2018	2115	691	5219	3856	0	3258	0
436	1610	5689	3811	5510	0	2111	0
5212	3645	2619	0	2566	0	1973	0
5014	1718	2053	0	2556	0	4532	0
4559	4078	4972	2803	582	0	4124	0
5343	6114	3647	1268	3500	0	3903	0
2025	5506	3330	524	5364	0	1133	0
711	3943	2724	1983	1635	0	5589	0
57	1664	5662	0	1671	0	1481	0
2323	3795	3635	0	513	0	3377	0
4309	446	2550	0	726	0	1977	0
3348	3312	2528	0	4829	0	877	0
686	2753	1803	0	4363	0	815	0
3254	4605	1292	0	5082	0	291	0
1211	3664	956	0	4964	0	5582	0
3661	1784	490	0	985	0	2207	0
2422	4896	159	0	292	0	814	0

68	5524	6092	0	5461	0	868	0
1285	2985	3428	0	2395	0	3332	0
5556	0	1970	0	3306	0	2908	0
2690	0	2876	0	5215	0	2532	0
2104	0	876	0	2001	0	5366	0
5631	3673	4595	0	6090	0	2397	0
5552	0	2665	0	3858	0	5702	0
5327	0	3928	0	2850	0	4366	0
3725	0	2679	0	3469	0	1359	0
4840	0	1069	0	1944	0	3329	0
2774	0	406	0	1079	0	1050	0
2371	0	1958	0				

公猪的更新具体方案如下表，采用的是模型分析中的预留更新法，下表中第一列生产公猪 ID 表示将被替换掉的生产公猪的编号（采用的是附件 6 的编号），第二列替换公猪 ID 是用预留更新法在 112-248 日龄中选出来的将用于替换生产公猪的猪只编号，该替换公猪被选中作为预留生产公猪的时间为表格中的第三列，具体的替换时间为第四列。如第一个更新方案为：在 2019 年第 1 天选中用编号为 5484 的公猪替换编号为 3746 的生产公猪，而实际替换的时间是 2019 年的第 112 天。并且第一列编号所对应的生产公猪品系与第二列编号对应的替换公猪品系相同。下表最后一个方案中配种 A 公表示今年新生的一只 A 系公猪，由于该公猪没有编号，所以用配种 A 公来代替。

生产公猪 ID	替换公猪 ID	确定该替换公猪的时间	替换时间	生产公猪 ID	替换公猪 ID	确定该替换公猪的时间	替换时间
3746	5484	1	112	2626	6848	145	256
1279	3284	5	116	2089	7216	146	257
2257	3702	26	137	5998	6917	153	264
619	9324	29	140	4022	7140	159	270
3065	8740	31	142	4888	6588	170	281
345	2236	35	146	4325	6505	194	305
4460	8686	37	148	6125	6549	200	311
2693	9014	43	154	4783	6620	211	322
3972	9306	47	158	1392	6614	214	325
1812	344	50	161	5578	6621	223	334
471	88	63	174	331	6769	230	341
1896	158	64	175	1223	6824	233	344
2240	8714	73	184	1126	6653	234	345
3204	8640	104	215	1391	6485	240	351
4013	790	108	219	6024	配种 A 公	258	365

配种计划如下表 10 所示，配种主要集中于前 28 周，该配种计划可以满足全年的销售计划和实现利润的最大化。考虑到配种成功率 90%和分娩成功率 88%，所以在原

计算得到的配种计划除以 79.2%再向上取整。

表 10 周配种计划表

周数	配种类型 1	头数 1	配种类型 2	头数 2	配种类型 2	头数 2
第 1 周	A 公 B 母	22	B 公 B 母	5	D 公 D 母	4
第 3 周	A 公 B 母	7	A 公 A 母	8	B 公 B 母	5
第 5 周	A 公 B 母	18	B 公 B 母	5		
第 7 周	A 公 B 母	22	B 公 B 母	5		
第 9 周	A 公 B 母	22	B 公 B 母	5		
第 11 周	A 公 B 母	22	B 公 B 母	5		
第 13 周	A 公 B 母	22	B 公 B 母	3		
第 14 周	C 公 D 母	12				
第 15 周	A 公 B 母	16				
第 16 周	C 公 D 母	12				
第 18 周	C 公 D 母	5				
第 19 周	A 公 B 母	13	C 公 D 母	7		
第 20 周	C 公 D 母	5				
第 21 周	C 公 D 母	5				
第 23 周	A 公 B 母	20	C 公 D 母	12		
第 25 周	C 公 D 母	12				
第 27 周	C 公 D 母	9				
第 28 周	A 公 B 母	20				

## 5.2 问题二分析与求解

确定副产品过程中考虑三个具体内容。

1、副产品每周的销售上限：每周统计的猪只状态矩阵 $Sta$ ，该矩阵“只进不出”，意思是不考虑任何的销售情况，即不出售种猪也不出售任何副产品。如第一列表示第一周分别符合四种副产品的猪只数量。这里我们每次出售都选择每类副产品中体重最大的猪。为了简化模型我们这里假设每次作为副产品卖出的猪只都处于其该类副产品对应体重的上限。如卖出的副产品一的体重都是 20kg 的。所以将每周卖出的每类副产品的重量 $bpq$ 矩阵转换为数量矩阵 $n_{ij}$ 。式 (26) 中的除法意思是第一行的每一个数都除以 20，其他行类似。最后得到副产品可用矩阵  $w$ ， $w$  是一个  $4*52$  的矩阵，每一列表示该周可出售的各类副产品的上限。

$$Sta = \begin{bmatrix} S_{11} & S_{12} & S_{13} & S_{14} & S_{15} & S_{16} & \cdots & S_{152} \\ S_{21} & S_{22} & S_{23} & S_{24} & S_{25} & S_{26} & \cdots & S_{252} \\ S_{31} & S_{32} & S_{33} & S_{34} & S_{35} & S_{36} & \cdots & S_{352} \\ S_{41} & S_{42} & S_{43} & S_{44} & S_{45} & S_{46} & \cdots & S_{452} \end{bmatrix} \quad (25)$$

$$\mathbf{n}_{ij} = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} & q_{15} & q_{16} & \cdots & q_{152} \\ q_{21} & q_{22} & q_{23} & q_{24} & q_{25} & q_{26} & \cdots & q_{252} \\ q_{31} & q_{32} & q_{33} & q_{34} & q_{35} & q_{36} & \cdots & q_{352} \\ q_{41} & q_{42} & q_{43} & q_{44} & q_{45} & q_{46} & \cdots & q_{452} \end{bmatrix} \div \begin{bmatrix} 20 \\ 50 \\ 100 \\ 120 \end{bmatrix} \quad (26)$$

$$\mathbf{w} = \begin{bmatrix} S_{11} & S_{12} & S_{13} & \cdots \\ S_{21} & S_{22} - \mathbf{n}_{11} & S_{23} & \cdots \\ S_{31} & S_{32} - \mathbf{n}_{21} & S_{33} - \mathbf{n}_{11} - \mathbf{n}_{22} & \cdots \\ S_{41} & S_{42} - \mathbf{n}_{31} - \mathbf{n}_{41} & S_{43} - \mathbf{n}_{21} - \mathbf{n}_{32} - \mathbf{n}_{41} - \mathbf{n}_{42} & \cdots \end{bmatrix} \quad (27)$$

2、养猪场存栏数量在 9000 以上:在出售副产品还有应该考虑到猪场所有猪的数量应大于 9000 只,每周猪场猪的总数量矩阵 $Q_j$ 如下,是一个 1\*52 的矩阵。如 $Q_1$ 表示第一周所有猪的数量。 $Q_j$ 必须满足约束式 (29)。

$$[Q_1 \quad Q_2 \quad Q_3 \quad Q_4 \quad Q_5 \quad Q_6 \quad \cdots \quad Q_{52}] \quad (28)$$

$$3Q_j > 9000 \quad (29)$$

3、保证产能约束:为了保证该养猪场的三类猪舍能容纳下其对应状态的猪只,每次进行状态转移时,比如说从将猪只从产房转移到保育房可能会出现保育房不能容纳下所有产房转移来的猪只,此时就将保育房中体重最重的猪依次挑出来作为副产品卖掉。由题意可知副产品一肯定从产房和保育舍中的猪只中选取的,从育肥舍中产生的必然属于副产品二、三、四。建立矩阵 $r\_sta$ ,该矩阵是一个 3\*52 的矩阵,表示“只进不出”状态下每周三类猪舍不能容纳的猪只数量。如第二列 $r_{12}$ 表示第二周时产房不能容纳的仔猪数量、 $r_{22}$ 表示第二周时保育舍不能容纳的处于保育阶段猪只的数量、 $r_{32}$ 表示第二周时育肥舍不能容纳的处于育肥阶段猪只的数量。

$$\mathbf{r\_sta} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} & r_{15} & r_{16} & \cdots & r_{152} \\ r_{21} & r_{22} & r_{23} & r_{24} & r_{25} & r_{26} & \cdots & r_{252} \\ r_{31} & r_{32} & r_{33} & r_{34} & r_{35} & r_{36} & \cdots & r_{352} \end{bmatrix} \quad (30)$$

将每次不能容纳的数量减去得到每周对应副产品的最低销售量矩阵 $r_{ij}$ ,该矩阵也是一个 3\*52 的矩阵,是由上面的 $r\_sta$ 推导出来的,具体的约束公式如式 (32) 所示。注意式 (32) 中的 $q_{i1}$ 第 i 周出售的副产品 1 的数量, $q_{i2}$ 表示第 i 周出售的副产品 2 的数量,其它类似。

$$\mathbf{r}_{ij} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & \cdots \\ r_{21} & r_{22} - r_{11} & r_{23} - r_{12} & \cdots \\ r_{31} & r_{32} - r_{31} - r_{21} & r_{33} - r_{32} - 2r_{31} - r_{21} - r_{22} - 2r_{11} & \cdots \end{bmatrix} \quad (31)$$

$$\begin{aligned}
r_{1j} &\leq q_{i1} \\
r_{2j} &\leq q_{i1} \\
r_{3j} &\leq q_{i2} + q_{i3} + q_{i4}
\end{aligned}
\tag{32}$$

此问我们采用遗传算法求解，考虑产能约束，运输能力约束，以及副产品价格约束、存栏数量约束以及销售的上下限约束产生初始种群[3][4]。将初始解放入遗传算法中进行优化后得到副产品销售计划，见附件 9。

### 5.3 问题三分析与求解

饲料消耗量的求解采用遍历的方式，将 19 年时间段内在该养猪场存在过的猪所消耗的各类饲料量都分别累加起来就得到各类饲料总的消耗量。第三问的求解过程分为三个部分，第一个求解原有猪只全过程所有的饲料食用，第二个是求解配种后新猪只加入所食的饲料，第三个是销售副产品和销售种猪所食的饲料。很明显，每周饲料食用量可第一部分加上第二部分减去第三部分得到。通过计算，问题三的求解结果见附件 10。

## 参考文献

- [1] 张金顺, 董新月, 杨美广, 猪生产周期性变化规律研究, 中国畜牧杂志, 36 卷第 2 期: 35-36, 2000 年。
- [2] 中小猪场种群的选留与更新, <http://www.doc88.com/p-9972628477701.html>, 访问时间 (2019 年 11 月 3 日)。
- [3] 毕晓君, 王朝. 一种基于参考点约束支配的 NSGA-III 算法 [J]. 控制与决策, 2019, 2:369-376.
- [4] 虞蕾, 赵红, 赵宗涛. 一种基于遗传算法的航迹优化方法 [J]. 西北大学学报: 自然科学版, 2006, 36 (2):205-208.

## 附录

Main

%% 对猪进行建模(8 个属性)

```

%猪的品系 [1:6] = [A:F]

pig.id = 1;

pig.type = 1;

pig.sex = 1; %1:boar; 2:sow

pig.begintime = 0; %开始日龄

pig.reartime = 0; %饲养时间

pig.quality = 0; %质量

%pig.amount = 1; %属性相同猪的头数

%猪的生长状态 1: 猪仔; 2:保育猪 1; 3: 保育猪 2; 4: 保育猪 3; 5: 育肥猪 1;
6: 育肥猪 2

pig.grow = 1;

%猪的功能状态 1: 待售种猪; 2: 生产母猪; 3: 生产公猪; 4: 后备母猪; :5: 后
备公猪; 0: 其他

pig.function = 0;

%是否为副产品 0: 不是; 1: 是

pig.coproduct = 0;

%% 数据输入

data = xlsread('data.xlsx');

disp('data is gathered!')

number = size(data,1); %猪的数量,属性个数

%统计参数

pig_sell = []; %待售种猪

```

```
len_pig_sell = 0;

pro_sow = [];      %生产母猪

len_pro_sow = 0;

optionsow = [];   %备选母猪

len_optionsow = 0;

pro_boar = [];    %生产公猪

len_pro_boar = 0;

optionboar = [];  %备选公猪

len_optionboar = 0;

%初始化猪的状态

pigs = repmat(pig,number,1);

for i = 1:number

    pigs(i).id = data(i,1);

    pigs(i).type = data(i,2);

    pigs(i).sex = data(i,3);

    pigs(i).begintime = data(i,4);

    pigs(i).reartime = data(i,5);

    pigs(i).quality = data(i,6);

    pigs(i).grow = data(i,7);

    pigs(i).function = data(i,8);

    pigs(i).coproduct = data(i,9);

    switch pigs(i).function
```

```

    case 1

        len_pig_sell = len_pig_sell + 1;

        pig_sell(len_pig_sell) = i;

    case 2

        len_pro_sow = len_pro_sow + 1;

        pro_sow(len_pro_sow) = i;

    case 3

        len_pro_boar = len_pro_boar + 1;

        pro_boar(len_pro_boar) = i;

    case 4

        len_optionsow = len_optionsow + 1;

        optionsow(len_optionsow) = i;

    case 5

        len_optionboar = len_optionboar + 1;

        optionboar(len_optionboar) = i;

    end

end

%公猪更新计划

% planboar;

%母猪更新计划

% plansow

```

```

%开始日龄

baoyu = find(data(:,4)<=77);

len_baoyu = length(baoyu);

a = [];

l_a = 0;

b = [];

l_b = 0;

c = [];

l_c = 0;

d = [];

l_d = 0;

e = [];

l_e = 0;

f = [];

l_f = 0;

for i = 1:len_baoyu

    switch data(baoyu(i),2)

        case 1

            l_a = l_a + 1;

            a(l_a) = baoyu(i);

        case 2

            l_b = l_b + 1;

```

```

        b(l_b) = baoyu(i);

    case 3

        l_c = l_c + 1;

        c(l_c) = baoyu(i);

    case 4

        l_d = l_d + 1;

        d(l_d) = baoyu(i);

    case 5

        l_e = l_e + 1;

        e(l_e) = baoyu(i);

    case 6

        l_f = l_f + 1;

        f(l_f) = baoyu(i);

    end

end

function solution = change(solution)

    for i = 1:4

        for j = 1:52

            switch i

                case 1

                    solution(i,j) = ceil(solution(i,j)/20);

                case 2

```

```

        solution(i,j) = ceil(solution(i,j)/50);

    case 3

        solution(i,j) = ceil(solution(i,j)/100);

    case 4

        solution(i,j) = ceil(solution(i,j)/120);

    end

end

end

end

function [optionboar,pigs,tmp2] = chooseboar(pro_boar,optionboar,pigs)

    tmp1 = pro_boar(end);

    tmp2 = update2(tmp1,pigs);

    if tmp2 ~= 0

        pigs(tmp2).function = 5;

        %防止重选

        pigs(tmp2).type = 10;

        optionboar = inserthead(optionboar,tmp2);

    end

end

function flag = classnumber(solution_ch)

    flag = 1;

    sta = xlsread('classnumber.xlsx');

```

```

for j = 1:52
    for i = 1:4
        tmp = sideacc(solution_ch,i,j);
        if sta(i,j) < solution_ch + tmp
            flag = 0;
            return;
        end
    end
end
end

function tmp = crossover(p1,p2)
%实值重组

[len,col] = size(p1);
tmp = zeros(len,col);

for i = 1:len
    for j = 1:col
        a = -3 + 7*rand();
        tmp(i,j) = p1(i,j) + a*(p2(i,j)-p1(i,j));
    end
end

% flag = vertificate(tmp);
% if flag == 1

```

```

%         p1 = tmp;

%     end

end

function value = evaluate(p1,price)

    value = 0;

    [row,col] = size(p1);

    for i = 1:row

        for j = 1:col

            value = value + p1(i,j)*price(i,j);

        end

    end

end

function quality = getquality(t)

    quality = 135.70369 + (5.1932-135.70369)/(1 + (t/123.274)^2.83968);

end

function [a,b] = getx(x)

    a = x/(0.4*0.9*0.95*9);

    b = x/(0.4*0.9*0.95*9.5);

end

function pigs = growup(pigs,day)

    number = size(pigs,1);

    for i = 1:number

```

```

        pigs(i).reartime = pigs(i).reartime + day;

        t = pigs(i).reartime + pigs(i).begintime;

        pigs(i).quality = getquality(t);

    end

end

function p = initial()

% 每周的载重不超过 44700

    row = 4;

    col = 52;

    p = zeros(row,col);

    for i = 1:col

        p(4,i) = rand()*44700;

        p(3,i) = rand()*(44700-p(4,i));

        p(2,i) = rand()*(44700-p(4,i)-p(3,i));

        p(1,i) = rand()*(44700-p(4,i)-p(3,i)-p(2,i));

    %         for j = 1:4

    %             p(j,i) = rand();

    %         end

    end

    %     flag = vertificate(p);

    %     if flag == 1

    %         p1 = tmp;

```

```

%     end

End

function list = inserthead(list,p)

%表头插入元素 p

    len = length(list);

    new = zeros(1,len+1);

    if len ~= 0

        for i = len:-1:1

            new(i+1) = list(i);

        end

    end

    new(1) = p;

    list = new;

end

function p1 = mutation(p1)

%     [row,col] = size(p1);

    for j = 1:52

        r = randperm(4,2);

        r1 = r(1);

        r2 = r(2);

        tmp = p1(r1,j);

        p1(r1,j) = p1(r2,j);

```

```

        p1(r2,j) = tmp;

    end

%     flag = verificate(tmp);

%     if flag == 1

%         p1 = tmp;

%     end

End

function p1 = select(P,f,q)

    mu = length(P);

%     Ptournament = zeros(1,q);

    Ftournament = zeros(1,q);

    for i = 1:q

        p = ceil(rand() * mu);

        Ptournament(i, :) = P(p).solution;

        Ftournament(i) = f(p);

    end

%收益最大

[~, maxIndex] = max(Ftournament);

row = length(Ptournament(maxIndex, :));

col = length(Ptournament(maxIndex, 1, :));

for i = 1:row

    for j = 1:col

```

```

        p1(i,j) = Ptournament(maxIndex,i,j);
    end

end

% mid=Ptournament(maxIndex,,:)
% p1 = Ptournament(maxIndex,,:);

End

function tmp = sideacc(solution_ch,i,j)

    row = i;

    col = j;

    tmp = 0;

    while row ~= 1 && col ~= 1

        row = row - 1;

        col = col - 1;

        tmp = tmp + solution_ch(row,col);

    end

    if i == 4 && j ~= 1

        tmp = tmp + sum(solution_ch(4,i-1));

    end

end

function flag = totalnumber(solution_ch)

    flag = 1;

```

```

%1*52 的矩阵

total = xlsread('totalnumber.xlsx');

t = zeros(1,52);

%计算每周卖的总头数

for i = 1:52

    for j = 1:4

        t(i) = t(i) + solution_ch(j,i);

    end

end

for i = 1:52

    s = sum(t(1:i));

    if total(i) - s < 9000

        flag = 0;

        break;

    end

end

end

function place = update(pig_old,pigs)

%更新备选母猪,返回作为更新猪的序号

number = length(pigs);

for i = 1:number

    if (pigs(i).begintime+pigs(i).reartime) >= 217 %&&

```

```

(pigs(i).begintime+pigs(i).reartime) <= 248

        if pigs(i).type == pigs(pig_old).type && pigs(i).sex == pigs(pig_old).sex

            place = i;

            break;

        end

    end

end

end

function place = update2(pig_old,pigs)

%更新公猪,返回作为更新猪的序号

    number = length(pigs);

    flag = 0;

    for i = 1:number

        if (pigs(i).begintime+pigs(i).reartime) >= 196 &&

(pigs(i).begintime+pigs(i).reartime) <= 248

            if pigs(i).type == pigs(pig_old).type && pigs(i).sex == pigs(pig_old).sex

                place = i;

                flag = 1;

                break;

            end

        end

    end

end

```

```

    if flag == 0
        place = 0;
    end
end
end

function [pro_boar,optionboar,pigs] = updateboar(pro_boar,optionboar,pigs)

    tmp = pro_boar(end);

    pigs(tmp) = [];

    len_op = length(optionboar);

    for i = 1:len_op

        if pigs(optionboar(i)).sex == pigs(tmp).sex && pigs(optionboar(i)).type ==
pigs(tmp).type

            tmp2 = optionboar(i);

            pro_boar = inserthead(pro_boar,tmp2);

            optionboar(i) = [];

            break;

        end

    end

end

end

function [pro_sow,optionsow,pigs] =
updatesow(pro_sow,tmp_pro_sow,optionsow,pigs)

%更新生产母猪,同时更新后备母猪

    n = length(tmp_pro_sow);

```

```

%      up2 = zeros(2,n);

up2 = [];

len_up2 = 0;

len_option = length(optionsow);

for i = 1:n

    tmp1 = tmp_pro_sow(i);

    %更新种群

    len_up2 = len_up2 + 1;

    up2(1,len_up2) = pigs(tmp1).id;

    for j = len_option:-1:1

        if pigs(tmp1).type == pigs(optionsow(j)).type %&& pigs(tmp1).sex ==
pigs(optionsow(j)).sex

            %更新生产母猪

            tmp2 = optionsow(j);

            up2(2,len_up2) = pigs(tmp2).id;

            pro_sow = inserthead(pro_sow,tmp2);

            %更新后备母猪

            optionsow(j) = [];

            pigs(tmp1).type = 10;

            pigs(tmp2).type = 10;

            place = update(tmp2,pigs);

            pigs(place).function = 4;

```



```

pc = 0.9;          %交叉概率

pm = 0.1;         %变异概率

q = 20;           %竞标赛选取个体大小

eval_budget = 500; %遗传代数

p = xlsread('price.xlsx');

price = p';

%统计数据

opt_tour_earn = 0;

opt_tour = zeros(4,52); %4*52 4 种副产品， 52 周的销量

evalcount = 0;      %遗传代数记录

f = zeros(1,lambda);

p.value = 0;

p.solution = [];

% P = repmat(p,lambda,1);

% Pnew = repmat(p,lambda,1);

%% 初始化种群

for i = 1:lambda

    intial_solution = initial();

    P(i).value = evaluate(intial_solution,price);

    P(i).solution = intial_solution;

    if (P(i).value > opt_tour_earn)

        opt_tour_earn = P(i).value;

```

```

        opt_tour = P(i).solution;
    end

end

evalcount = evalcount + 1;

best_value(evalcount) = opt_tour_earn;

best_solution(evalcount,,:) = opt_tour;

%% 进化循环

while evalcount <= eval_budget

    for i = 1:lambda

        p1 = select(P,f,q);

        %交叉

%        if rand()<pc

%            %进行交叉

%            p2 = select(P,f,q);

%            p1 = crossover(p1,p2);

%        end

%        %变异

%        if rand()<pm

%            p1 = mutation(p1);

%        end

        temp=p1;

        Pnew(i).solution = temp;
    end
end

```

```

end

P = Pnew;

%计算该代最优个体

opt_tour_earn = 0;

for i = 1:lambda

    solution = P(i).solution;

    P(i).value = evaluate(solution,price);

    if (P(i).value > opt_tour_earn)

        opt_tour_earn = P(i).value;

        opt_tour = P(i).solution;

    end

end

end

evalcount = evalcount + 1;

%    opt_tour_earn

best_value(evalcount) = opt_tour_earn;

best_solution(:,evalcount) = opt_tour;

x = 1:1:evalcount;

figure(1)

plot(x,best_value);

end

%% 参数输出

```

```
% opt_tour

[opt_tour_earn,maxindex] = max(best_value);

opt_tour = best_solution(:,maxindex);

opt_tour_earn

maxindex

xlswrite('solution.xlsx',opt_tour');
```