

赛区评阅编号（由赛区组委会填写）：

---

## 2020 高教社杯全国大学生数学建模竞赛

### 承 诺 书

我们仔细阅读了《全国大学生数学建模竞赛章程》和《全国大学生数学建模竞赛参赛规则》（2019 年修订稿，以下简称为“竞赛章程和参赛规则”，可从全国大学生数学建模竞赛网站下载）。

我们完全清楚，在竞赛开始后参赛队员不能以任何方式，包括电话、电子邮件、“贴吧”、QQ 群、微信群等，与队外的任何人（包括指导教师）交流、讨论与赛题有关的问题；无论主动参与讨论还是被动接收讨论信息都是严重违反竞赛纪律的行为。

我们完全清楚，抄袭别人的成果是违反竞赛章程和参赛规则的行为；如果引用别人的成果或资料（包括网上资料），必须按照规定的参考文献的表述方式列出，并在正文引用处予以标注。

**我们以中国大学生名誉和诚信郑重承诺，严格遵守竞赛章程和参赛规则，以保证竞赛的公正、公平性。如有违反竞赛章程和参赛规则的行为，我们将受到严肃处理。**

我们授权全国大学生数学建模竞赛组委会，可将我们的论文以任何形式进行公开展示（包括进行网上公示，在书籍、期刊和其他媒体进行正式或非正式发表等）。

我们参赛选择的题号（从 A/B/C/D/E 中选择一项填写）：   A  

我们的报名参赛队号（12 位数字全国统一编号）：   202010038016  

参赛学校（完整的学校全称，不含院系名）：   南京大学  

参赛队员 (打印并签名)：1.   陈万驰 181840033 数学系  

2.   姜玉骅 181870080 工程管理学院  

3.   余帅杰 181860077 计算机科学与技术系  

指导教师或指导教师组负责人 (打印并签名)：   教练组  

（指导教师签名意味着对参赛队的行为和论文的真实性负责）

日期：   2020   年   08   月   1   日

**（请勿改动此页内容和格式。此承诺书打印签名后作为纸质论文的封面，注意电子版论文中不得出现此页。以上内容请仔细核对，如填写错误，论文可能被取消评奖资格。）**

赛区评阅编号（由赛区组委会填写）：

---

**2020 高教社杯全国大学生数学建模竞赛**

**编 号 专 用 页**

赛区评阅记录（可供赛区评阅时使用）：

评 阅 人						
备 注						

送全国评阅统一编号（由赛区组委会填写）：

全国评阅随机编号（由全国组委会填写）：

（请勿改动此页内容和格式。此编号专用页仅供赛区和全国评阅使用，参赛队打印后装订到纸质论文的第二页上。注意电子版论文中不得出现此页。）

# 穿越沙漠

## 摘要

关键字：新冠肺炎疫情 SEIR 模型 混样检测

## 一、问题重述

考虑如下的小游戏：玩家凭借一张地图，利用初始资金购买一定数量的水和食物（包括食品和其他日常用品），从起点出发，在沙漠中行走。途中会遇到不同的天气，也可在矿山、村庄补充资金或资源，目标是在规定时间内到达终点，并保留尽可能多的资金。游戏的基本规则如下：

1. 以天为基本时间单位，游戏的开始时间为第 0 天，玩家位于起点。玩家必须在截止日期或之前到达终点，到达终点后该玩家的游戏结束。
2. 穿越沙漠需水和食物两种资源，它们的最小计量单位均为箱。每天玩家拥有的水和食物质量之和不能超过负重上限。若未到达终点而水或食物已耗尽，视为游戏失败。
3. 每天的天气为“晴朗”、“高温”、“沙暴”三种状况之一，沙漠中所有区域的天气相同。
4. 每天玩家可从地图中的某个区域到达与之相邻的另一个区域，也可在原地停留。沙暴日必须在原地停留。
5. 玩家在原地停留一天消耗的资源数量称为基础消耗量，行走一天消耗的资源数量为基础消耗量的 2 倍。
6. 玩家第 0 天可在起点处用初始资金以基准价格购买水和食物。玩家可在起点停留或回到起点，但不能多次在起点购买资源。玩家到达终点后可退回剩余的水和食物，每箱退回价格为基准价格的一半。
7. 玩家在矿山停留时，可通过挖矿获得资金，挖矿一天获得的资金量称为基础收益。如果挖矿，消耗的资源数量为基础消耗量的倍；如果不挖矿，消耗的资源数量为基础消耗量。到达矿山当天不能挖矿。沙暴日也可挖矿。
8. 玩家经过或在村庄停留时可用剩余的初始资金或挖矿获得的资金随时购买水和食物，每箱价格为基准价格的 2 倍。

需要解决以下三个问题。

1. 假设只有一名玩家，在整个游戏时段内每天天气状况事先全部已知，试给出一般情况下玩家的最优策略。求解附件中的“第一关”和“第二关”，并将相应结果分别填入 Result.xlsx。
2. 假设只有一名玩家，玩家仅知道当天的天气状况，可据此决定当天的行动方案，试给出一般情况下玩家的最佳策略，并对附件中的“第三关”和“第四关”进行具体讨论。
3. 现有  $n$  名玩家，他们有相同的初始资金，且同时从起点出发。若某天其中的任意  $k$  名玩家均从区域  $A$  行走到区域  $B$ ，则他们中的任一位消耗的资源数量均为基础消耗

量的  $2k$  倍；若某天其中的任意  $k$  玩家在同一矿山挖矿，则他们中的任一位消耗的资源数量均为基础消耗量的 3 倍，且每名玩家一天可通过挖矿获得的资金是基础收益的；若某天其中的任意名玩家在同一村庄购买资源，每箱价格均为基准价格的  $\frac{1}{k}$  倍。其他情况下消耗资源数量与资源价格与单人游戏相同。

- (a) 假设在整个游戏时段内每天天气状况事先全部已知，每名玩家的行动方案需在第 1 天确定且此后不能更改。试给出一般情况下玩家应采取的策略，并对附件中的“第五关”进行具体讨论。
- (b) 假设所有玩家仅知道当天的天气状况，从第 2 天起，每名玩家在当天行动结束后均知道其余玩家当天的行动方案和剩余的资源数量，随后确定各自第二天的行动方案。试给出一般情况下玩家应采取的策略，并对附件中的“第六关”进行具体讨论。

## 二、 问题分析

### 2.1 问题一分析

### 2.2 问题二分析

### 2.3 问题三分析

### 2.4 问题四分析

混样检测，顾名思义就是将几个样本混合在一起进行一次核酸检测。某次需要对  $N$  个人进行检测，病毒的携带率为  $p$ 。为了减少工作量，一次性将  $k$  个人的样本混合，如果混合样本为阴性，则说明这  $k$  个人都没有患病。如果混合样本为阳性，说明这  $k$  个人中至少有 1 个人的样本是阳性，那么接下来对这  $k$  个人的分别单独做检测。而单次检测简单直观，即对所有人单独做检测。考虑用基本概率论知识，比较混样检测和单次检测方法对应到平均每人的检测次数，选取更合理的方案。

## 三、 模型假设

1. 假设人群中所有个体都有被感染的概率。
2. 假设被感染个体痊愈后，会产生抗体，不会再被感染。
3. 假设所有感染者是同质的，即病情的严重程度、死亡率相同。
4. 鉴于 1 月 23 日武汉市封城，假设武汉市总人口的不变。
5. 假设核酸检测准确率为 100%。
6. 假设混样检测的结果不会因混样而改变，即若至少有一人感染，混样检测呈阳性；若无人感染，混样检测呈阴性。



## 四、符号说明

表 1 符号说明表

参数	定义	单位
$Weight$	负重上限	千克
$Q$	初始资金	元
$t$	天数	天
$T$	总天数	天
$m_w$	每箱水的质量	千克/箱
$m_f$	每箱食物的质量	千克/箱
$p_w$	水的基准价格	元/箱
$p_f$	食物的基准价格	元/箱
$n_{sw}$	晴朗天气下水的基准消耗量	箱
$n_{hw}$	高温天气下水的基准消耗量	箱
$n_{ow}$	沙暴天气下水的基准消耗量	箱
$n_{sf}$	晴朗天气下食物的基准消耗量	箱
$n_{hf}$	高温天气下食物的基准消耗量	箱
$n_{of}$	沙暴天气下食物的基准消耗量	箱
$n$	玩家数	人
$P_t$	第 $t$ 天开始时玩家所处的位置	/
$W_t$	第 $t$ 天开始时玩家剩余的水	箱
$F_t$	第 $t$ 天开始时玩家剩余的食物	箱
$Q_t$	第 $t$ 天开始时玩家剩余的资金	元
$S_t$	第 $t$ 天玩家所处的地点特征	/
$Weat_t$	第 $t$ 天的天气	/
$Q_{Mine}$	基础收益	元

## 五、模型建立与求解

### 5.1 游戏模型的建立

我们首先将该游戏利用数学语言加以描述。显然该局游戏的最终目的是使玩家到达终点时的收益最大，即

$$\max Q_{30} + \frac{1}{2}p_w W_{30} + \frac{1}{2}p_f F_{30} \quad (1)$$

其中  $Q_t, W_t, F_t$  分别表示第  $t$  天时玩家所剩下的资金、水和食物量。如果玩家在第 30 天前到达终点，则其各个属性将会在未来几天视作不变，所以我们以第三十天为统一结束时间。该目标函数有如下约束：

$$Q_t = Q_{t-1} + Q_{Mine} Mine_t - Shop_t [2p_f Shop F_t + 2p_w Shop W_t] \quad (2)$$

其中

$$Mine_t = \begin{cases} 0, & \text{如果第 } t \text{ 天不挖矿} \\ 1, & \text{如果第 } t \text{ 天挖矿} \end{cases}$$

$$Shop_t = \begin{cases} 0, & \text{如果第 } t \text{ 天不购物} \\ 1, & \text{如果第 } t \text{ 天购物} \end{cases}$$

即每天结束时的资金等于前一天的资金加上当天挖矿获得的 1000 元（如果挖矿的话），再减去在村庄购买食物和水花费的钱（如果购买的话）。其中  $Shop F_t$  和  $Shop W_t$  分别表示玩家在第  $t$  天购买的食物量和水量（如果购买的话）。

$$F_t = F_{t-1} - 2Move_t \Delta F_t - 3Mine_t Move_t \Delta F_t - (1 - Move_t - Mine_t) \Delta F_t + Shop_t Shop F_t \quad (3)$$

即每天结束时的食物量等于前一天的食物量减去当天的食物消耗量再加上在村庄购买的食物量（如果购买的话）。

$$W_t = W_{t-1} - 2Move_t \Delta W_t - 3Mine_t \Delta W_t - (1 - Move_t - Mine_t) \Delta W_t + Shop_t Shop W_t \quad (4)$$

即每天结束时的水量等于前一天的水量减去当天的水消耗量再加上在村庄购买的水量（如果购买的话）。



$$P_t = P_{t-1}(1 - Move_t) + \bar{P}_{t-1}Move_t \quad (5)$$

其中  $P_t$  表示第  $t$  天的位置， $\bar{P}_t$  表示第  $t+1$  天可以到达的几个相邻区域之一，

$$Move_t = \begin{cases} 1, & \text{如果第 } t \text{ 天移动} \\ 0, & \text{如果第 } t \text{ 天不移动（包括挖矿、停留）} \end{cases}$$

$$Shop_t \leq If_0[(P_t - C_1)(P_t - C_2) \cdots (P_t - C_n)] \quad (6)$$

$$Mine_t \leq If_0[(P_{t-1} - K_1)(P_{t-1} - K_2) \cdots (P_{t-1} - K_m)] \quad (7)$$

其中  $C_1, C_2, \dots, C_n$  表示  $n$  个村庄的位置， $K_1, K_2, \dots, K_m$  表示  $m$  个矿山的位置，

$$If_0(x) = \begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{if } x \neq 0 \end{cases}$$

$\Delta F$  和  $\Delta W$  分别表示第  $t$  天基础消耗的食物量和水量，由 Lagrange 插值公式：

$$L(x) = \sum_{i=1}^n f(x_i) \frac{\prod_{j \neq i}(x - x_j)}{\prod_{j \neq i}(x_i - x_j)}$$

可得

$$\Delta F_t = n_{sf} \frac{Wea_t^2 - 3Wea_t + 2}{2} + n_{hf} \frac{Wea_t^2 - 2Wea_t}{-1} + n_{of} \frac{Wea_t^2 - Wea_t}{2} \quad (8)$$

$$\Delta W_t = n_{sw} \frac{Wea_t^2 - 3Wea_t + 2}{2} + n_{hw} \frac{Wea_t^2 - 2Wea_t}{-1} + n_{ow} \frac{Wea_t^2 - Wea_t}{2} \quad (9)$$

其中

$$Wea_t = \begin{cases} 0, & \text{晴朗天气} \\ 1, & \text{高温天气} \\ 2, & \text{沙暴天气} \end{cases}$$

表示第  $t$  天的天气情况。

## 5.2 问题一的分析与求解

### 5.2.1 地图简化

我们首先引入 Dijkstra 算法，这是一种在有向赋权图中求两点之间最短路径的高效算法。

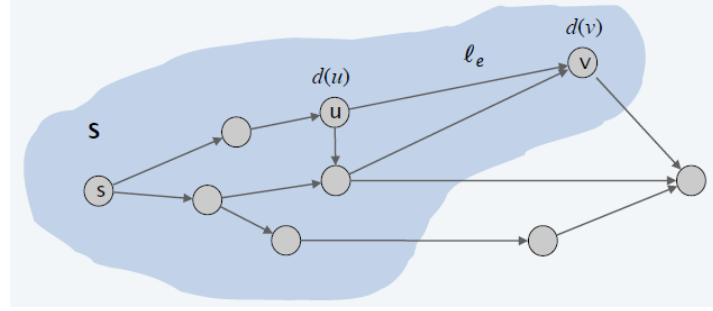


图 1 Dijkstra 算法示意图

如图1, 该图中  $s$  是起点, 最右侧的点是终点。  $d(u)$  表示从点  $s$  到点  $u$  的最短距离。之后执行如下操作:

- (1) 初始化  $S = s, d(s) = 0$ ;
- (2) 反复寻找未探索过的点  $v$ , 使得  $\min \pi(v)$ , 将  $v$  添加到  $S$  中, 并且  $d(v) = \pi(v)$ , 其中

$$\pi(v) = \min_{e=(u,v):u \in S} d(u) + \text{distance}(\arg \min_{e=(u,v):u \in S} d(u), v)$$

最终当终点属于  $S$  时, 可以知道起点到终点的最短路径长度, 从而问题求解。

可以看出, 问题一的核心关键在于玩家是否前往矿山挖矿、在矿山连续挖几天矿、在挖矿之后是否需要前往村庄补给物资以及是否可以在矿山和村庄之间往返。我们将问题作如下简化:

选取图中起点、终点、村庄和矿山作为特殊点, 利用 Dijkstra 算法计算出每两个特殊点之间的最短路径 (即不考虑沙暴天气的影响下, 所需最少的天数), 如图2所示。

该图是一个完全图 (即图中每两点之间都有一条边), 且每条边上的数字代表着两个特殊点之间的最短路径 (受天气影响可能会有变化)。对于问题一, 因为天气是预先给定的, 整局游戏没有随机因素, 所以我们假定玩家在游戏途中始终向着某个特殊点 (终点、村庄、矿山) 前进并且选择最合适的路径, 而非漫无目的地随机移动, 即游戏旅程由几条有向线段叠加而成:

$$\vec{P} = \sum_{k=1}^n \overrightarrow{P_{i_k} P_{i_{k+1}}} \quad (10)$$

其中  $P_0, P_1, P_2, P_3$  分别表示起点、村庄、矿山和终点,  $i_k \leq 3, \quad k = 1, 2, \dots, n$ .

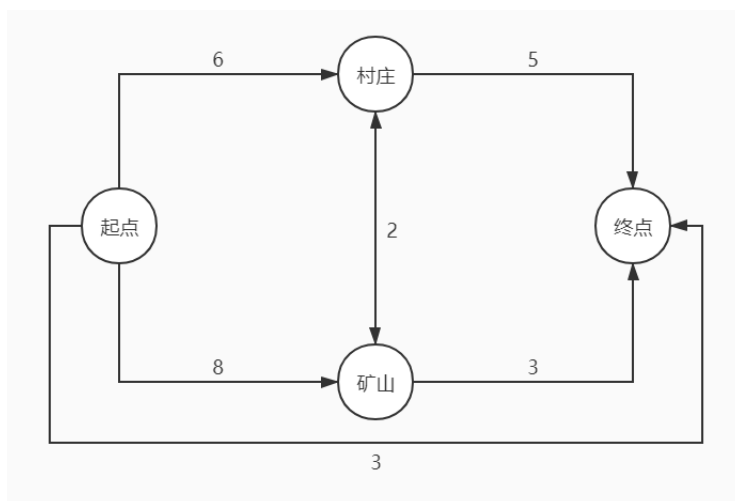


图 2 简化后的 Map1

因此，问题一可简化为玩家在这几个特殊点之间的运动问题。由于第一关地图中村庄在由起点去往矿山的必经之路上，所以应该考虑在玩家第一次到达村庄时适量补充水和食物，然后前往矿山尽可能多地挖矿，在水和食物只够支撑玩家返回村庄时停止挖矿，选择返回村庄。在村庄补给一定量的水和食物（可以通过计算之后几天返回终点时的行程来确定需要购买的水和食物的量，以使得玩家到达终点时水和食物剩余量都为 0，从而杜绝资产损失。）

### 5.2.2 路径方案

我们将针对问题一的策略方案用流程图的形式展示出来, 如图??。根据这样的流程

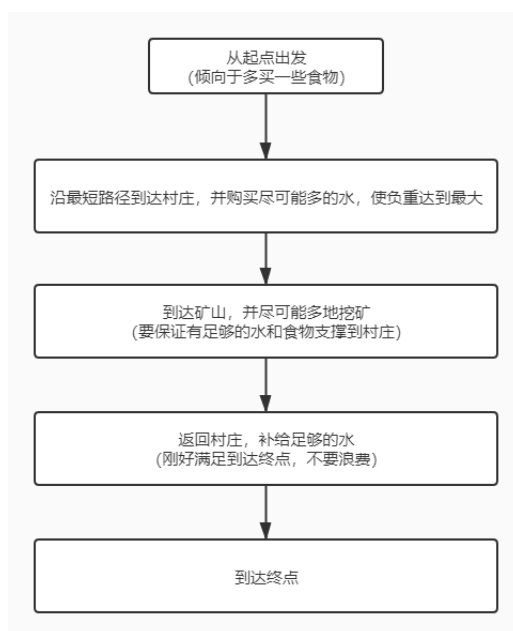


图 3 第一关策略方案流程图

图，我们进行问题分析和代码求解。玩家在游戏过程中的可选择性因素有：起点处的物资储备、在村庄的物资购买、在矿山挖矿的持续天数、返回村庄时的物资购买等。对此，我们对起点处的物资准备和在矿山挖矿的持续天数进行有范围穷举，对两次村庄物资购买分别采用精准计算的方法，借助 Python 编程语言求解。

其中，第一次到达村庄时购买物资的策略为：只购买水，且尽量让负重达到最大。即

$$W_{buy1} = \min\left\{\left[\frac{Weight - m_w W_{c1} - m_f F_{c1}}{m_w}\right], \frac{Q_{c1}}{p_w}\right\} \quad (11)$$

$Weight$  为最大负重， $W_{c1}$  为第一次到达村庄时剩余的水量， $F_{c1}$  为第一次到达村庄时剩余的食物量， $m_w$  和  $m_f$  分别表示一箱水和一箱食物的质量， $Q_{c1}$  为第一次到达村庄时的剩余资金， $p_w$  为一箱水的价格。

第二次到达村庄时购买物资的策略为：准确满足之后到达终点所需的全部水和食物，使得到达终点时不剩余任何水和食物。具体代码实现详见附件。

经过代码验证，最终找到最佳的策略方案，路径如图??所示。

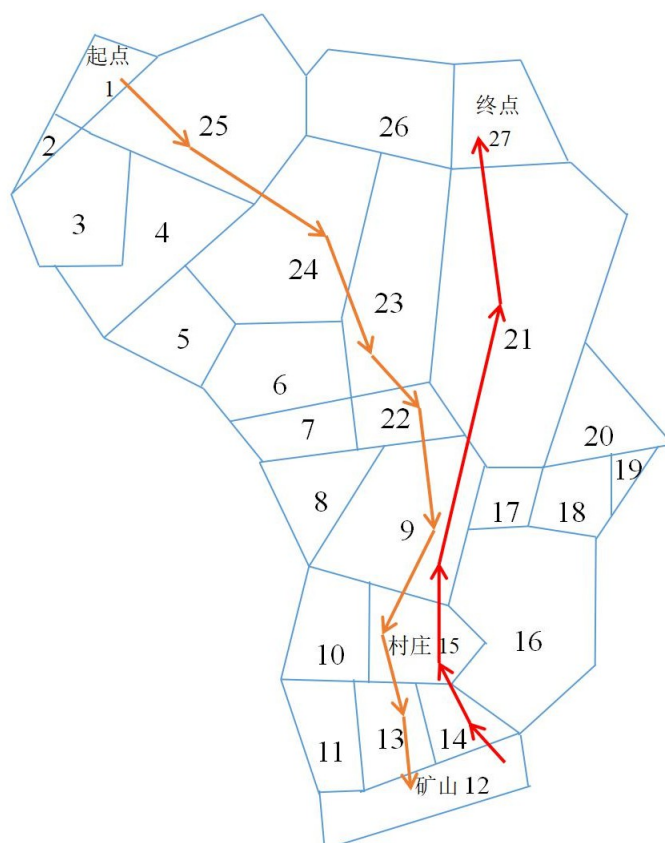


图 4 第一关路径示意图

具体的游戏路径：经历 8 天时间到达村庄 15（其中有 2 天沙暴天气，只能停在原

地), 当天在村庄购买水 163 箱。第二天立即前往矿山, 第 10 天到达矿山, 第 11-17 天进行挖矿, 第 18 天停留在矿山。第 19-20 天前往村庄, 并在第 20 天当天购买水 36 箱、食物 40 箱。第 21-23 天前往终点, 最终于第 23 天到达终点, 水和食物全部消耗完毕。最终剩余资金 10430 元。完整路径详见附件。

## 六、问题三

$$\begin{aligned} \max Q + Q_{Mine}(m_4 + m_5 + m_6 + m_7 + m_8) - 2(n_{sw}p_w + n_{sf}p_f)(n_1 + n_3 + n_4 + n_5 + \\ n_6) - 2(n_{hw}p_w + n_{hf}p_f)(n_2 + n_7 + n_8 + n_9 + n_{10}) - 3(n_{sw}p_w + n_{sf}p_f)(m_4 + m_5 + m_6) - \\ 3(n_{hw}p_w + n_{hf}p_f)(m_7 + m_8) - (n_{hw}p_w + n_{hf}p_f)(1 - n_2 + 1 - n_7 - m_7 + 1 - n_8 - m_8) - \\ (n_{sw}p_w + n_{sf}p_f)(1 - n_4 - m_4 + 1 - n_5 - m_5 + 1 - n_6 - m_6) \end{aligned}$$

## 参考文献