Scientific and Social Research
2022, Volume 4, Issue 4

Design and Research on the Optimal Strategy of Desert-Crossing Outward Bound Training Program

Jing Cong*

Basic Education Department of Liaoning Institute of Science and Technology, Benxi 117004, Liaoning Province, China

Corresponding author: Jing Cong, congjing_2003@163.com

Abstract: Desert-crossing outward bound training program is an effective way to improve an individual's judgment and execution. This paper aims to provide an optimized approach to the training program as it proposes an optimal strategy for crossing the desert with the given weather conditions, map, the upper limit of load, initial funds, deadline, basic income, weight and benchmark price of water and food, as well as basic consumption in various weather conditions.

Keywords: Enumeration method; Recursive method; Dynamic programming model; Floyd-Warshall algorithm

Online publication: April 18, 2022

1. Introduction

In a game setting, the optimal strategy for the desert-crossing outward bound training program requires each player to travel through a desert with a map and fixed initial funds, with which a certain amount of water and food can be purchased. The player will encounter different weathers along the way. Funds or resources can be replenished in mines and villages. The goal is to arrive at the destination within a specified time while saving as much funds as possible. The game rules are as follows: the player will be located at the starting point on the first day and is supposed to arrive at the destination within a time frame, which signals the end of the game; water and food are necessities, but their quantities should not exceed the upper limit of load; if the player runs out of water and food before the game ends, the player will lose the game; during the journey, there may be sunny days, hot days, and even sandstorms; the player may choose to stay or continue to travel to nearby areas every day, except for sandstorm conditions, which force the player to suspend; staying for a day will consume the basic consumption, whereas walking consumes twice the amount; on the first day, the player can buy water and food at the starting point for the benchmark price, and the remainder can be returned at half the benchmark price after reaching the destination; the player can earn basic income by staying and working at the mine, which consumes three times the basic consumption and can only be commenced the following day after arrival; when passing or staying in the village, the player can purchase water and food at twice the benchmark price. The desert-crossing outward bound program is a potent method to enhance one's judgment and execution. Through the guidance and experience of the program, participants will recognize the importance of goal setting and management, realize the impact and damage caused by improper use of resources, as well as strengthen their management, communication, sales, and leadership skills. The game is implemented to reveal problems unwittingly, discover the root causes to these problems, and enable participants to experience the notion of management.

2. Principles and methods of designing an optimal strategy for the desert-crossing outward bound training program

There are two problems that need to be solved in the desert-crossing outward bound training program. First, a player who knows the daily weather, map, upper limit of load, initial funds, time limit, basic income, weight and the benchmark price of water and food, as well as the basic consumption in various weather conditions, which can be used as the optimal strategy for the "First Level." Second, a player who only knows the weather of the current day, map, the upper limit of load, initial funds, time limit, basic income, weight and benchmark price of water and food, as well as the basic consumption in various weather conditions, in order to solve the "Second Level."

In order to address the optimal strategy of desert-crossing under certain conditions, this paper simulates the six levels through the enumeration method, dynamic programming, breadth-first search, recursive algorithm, inequality, and graph theory. Through model creation and analysis by Python [1] and Matlab [2-6] as well as data analysis and processing by Visio, the optimal strategy of each level is obtained, and the results are investigated.

3. Establishment and solution of an optimal strategy model for the desert-crossing outward bound training program

Before establishing the model, the data provided need to be pre-processed; that is, classifying and inputting the data. The base units are defined as global variables; the values that will change along the journey are defined as the player's attributes, such as food, water, funds, and the player's current load. Given the irregularity of the map data, it is necessary to scan and convert them into two-dimensional arrays before inputting, so that the computer can detect and process them easily. Meanwhile, another array must be created to input the weather data provided. The completion of the overall program with the combination of breadth-first search and recursive algorithm realizes the enumerative induction model. Beginning from the coordinate of the initial solution site, 0, the map and daily temperature changes are stored in the matrix, and the following judgments are mandated: whether the player arrives at the village; whether the player needs to purchase necessities; how much water and food the player wants to purchase. Following that, each method is listed one by one. The recursive operation is carried out using the Python program, and all solution elements are combined into a single viable solution to the problem [4].

3.1. Establishment and solution of the enumerative induction model

(1) In order to figure out this problem through programming, it is necessary to divide the program into functional modules. The functions and associations of each program module need to be determined, and the code amount should be reduced to the greatest extent, in order to lower the programming burden and speed up the operation.

(2) The program can be divided into five functional modules: "alive," "jx," "buy," "runmaze," and "action." Among them, "alive" (**Figure 1**) is used to judge whether the player's current conditions meet the game rules, and whether the player should be eliminated.

```
def alive(people):
    a=False
    if (people["水"] >0) and (people["食物"] >0) and (0<people["当前负重"]<fzsx) and (people["钱"]>0):
        a=True
    return a
```

Figure 1. Program diagram of "alive"

The “jx” (**Figure 2**) module is used to compare the final benefits of each path to the destination, in order to filter out the optimal route.

```
def jx(people,erlist):#结算方法，看是不是最优解
    money=people["钱"]+people["水"]*watermoney/2+people["食物"]*foodmoney/2
    global maxmoney
    global maxerlist
    if money>maxmoney:
        maxmoney=money
        maxerlist=erlist
```

Figure 2. Program diagram of “jx”

The “buy” (**Figure 3**) module is used to output all possibilities of water and food purchased with the given funds and remaining load.

```
def buy(money,nowWight):#购买多少水和食物
    a=[]
    if money==10000:
        ratewater=money//watermoney
        ratefood=money//foodmoney
    else:
        ratewater = money // (watermoney*3)
        ratefood = money // (foodmoney*3)
    for i in range(36,ratewater):
        for j in range(36,ratefood):
            if (0<(i*water+j*food+nowWight)<fzsx) and (money>0):
                a.append([i,j])
    return a
```

Figure 3. Program diagram of “buy”

As the main bodies of the model, “runmaze” (**Figure 4**) is used to judge whether the player has arrived at the village, while “action” is the implementation of the breadth-first search and recursive algorithm, which can be used to repeatedly call “runmaze” and “action” to simulate the player’s behavior.

```
def runMaze(people,site,day,erlist):#判定到没到村庄要不要购物
    if day<=30:
        erlist.append([day, site])
        if site in city:
                b = buy(people["钱"], people["当前负重"])
                # b=[[200,200]]
                apeople = copy.deepcopy(people)
                for i in b:
                    people["水"] = apeople["水"]+i[0]
                    people["食物"] = apeople["食物"]+i[1]
                    people["钱"] = apeople["钱"]-(people["水"] * watermoney*2 + people["食物"] * foodmoney*2)
                    people["当前负重"] = people["水"] * water + people["食物"] * food
                    if people["钱"]>0:
                        action(people,site,day,erlist)
                        people = copy.deepcopy(apeople)
        else:
            action(people, site, day, erlist)
        del(erlist[-1])
```

Figure 4. Program diagram of "runmaze"

As the time complexity of the enumeration method is relatively high, the speed of the computer will slow down when the program is running.

$$n + (n - 1) + (n - 2) + \cdots + 1 = \frac{n(n + 1)}{2} = \frac{n^2}{2} + \frac{n}{2}$$

The player's behavior is simulated through the compilation of the Python program. By outputting all possible conditions, the optimal route of the "First Level" is obtained (**Figure 5**).

```
10990.0
[[0, 1], [1, 25], [2, 24], [3, 23], [4, 23], [5,
 21], [6, 9], [7, 9], [8, 15], [9, 14], [10, 12],
 [11, 12], [12, 12], [13, 14], [14, 15], [15, 14],
 [16, 14], [17, 14], [18, 14], [19, 12], [20, 12],
 [21, 12], [22, 12], [23, 12], [24, 12], [25, 12],
 [26, 16], [27, 17], [28, 21], [29, 27]]
```

Figure 5. Output of the "First Level" optimal route

3.2. Establishment and solution of the dynamic programming model

For the "Second Level," the player only knows the current day's weather, map, the upper limit of load, initial funds, time limit, basic income, weight and benchmark price of water and food, as well as the basic consumption in various weather conditions. Dynamic programming and Floyd-Warshall algorithm [7-10] can be used for calculation. Sandstorms are known to occur just once every 30 days, but the exact weather is uncertain. The distance from the starting point to the village is equal to the distance between the starting point to the mine. Through model analysis, the calculated consumption from the mine to the endpoint is higher than that from the village. The back-and-forth relationship between the mine and the village is discussed.

Due to a lack of meteorological data, it must be assessed which path will leave the player with the most money after arriving at the endpoint in the allotted time: the one through the mine, or the one journeying from the starting point to the endpoint directly. It is known that the shortest time required to get to the mine from the starting point is equal to the one to the endpoint. All that is left is to judge whether, no matter what the weather is, the money earned from mining is always greater than the sum of the money consumed while mining and on the journey from the mine to the destination. It will take at least three days to get to the mine from the starting point, and at least two days from the mine to the endpoint. Given that the time limit is within 10 days, the mining time should be less than five days.

The parameters are set as follows: there will be M (days) of sunny weather and N (days) of hot weather during mining. The inequalities can be obtained as follows:

$$200\ (M+N) > 165M + 405N\ (Mining\ income\ is\ greater\ than\ mining\ consumption) \quad (1)$$

$$\mathrm{M} + \mathrm{N} <= 5\ \text{(The total mining time is less than five days)} \quad (2)$$

$$5 \geq \mathrm{M} \geq 0 \quad (3)$$

$$5 \geq \mathrm{N} \geq 0 \quad (4)$$

By calculating the inequalities, the following can be obtained:

$$200\mathrm{M} > 165M + 135N \quad (5)$$

According to the transformation of inequalities, the following can be obtained:

$$\mathrm{M} > 3.85N \quad (6)$$

It can be concluded that there are six possible weather ratios of M:N; namely, 5:0, 4:1, 4:0, 3:0, 2:0, and 1:0. Presumably that the weather is at best condition, i.e., $\mathrm{M} = 5$, the total income from mining for five days is 1,000 yuan, the money consumed is 825 yuan, and the net income from mining is 175 yuan. It will take two days to travel from the mine to the endpoint after mining. Assuming that the weather condition is at the best, which is sunny, the capital consumed in two days will be 220 yuan. Since 175 is less than 220, and the assumption is made under the best weather condition as well as the least consumption of resources, the route from the starting point to the endpoint through the mine is unable to make ends meet, let alone the fact that the consumptions of all other possibilities would only be higher than that of this situation.

4. Conclusion

In view of the optimal strategy for the desert-crossing outward bound training program, this paper examines the optimal strategy under two circumstances of knowing and not knowing the daily weather conditions, which offers an optimal approach to the training program. In addition, the optimal strategy proposed in this paper can be extendedly applied in real-life situations. When there are few and relatively certain influencing variables, the model can be used for quantitative analysis to obtain the optimal solution.

Project

This paper is one of the initial achievements of the general project "Research on the Construction Strategy of Teachers with Double-Qualifications and Double-Capabilities in Applied Universities" (Project Number: JG20DB248), which was established by the 13th Five-Year Plan of Educational Science in Liaoning Province in 2020.

Disclosure statement

The author declares no conflict of interest.

References

[1] Wei D, 2019, Fundamentals and Applications of Python Programming, Posts & Telecom Press, Beijing.

[2] Ai D, 2018, Matlab and Mathematical Experiments, China Machine Press, Beijing.

[3] Shi B, 2004, Matlab Mathematics Calculation Example Tutorial, China Railway Publishing House, Beijing.

[4] Jiang Q, 2005, College Math Experiment, Tsinghua University Press, Beijing.

[5] Wang W, 2006, Detailed Explanation of Mathematical Modeling and Its Basic Knowledge, Wuhan University Press, Wuhan.

[6] Zhang X, 2006, Advanced Mathematics Experiment, East China University of Technology Press, Shanghai.

[7] Quyang L, 2017, The Shortest Path Problem – Floyd Algorithm. Beijing Chuangxin Lezhi Network Technology Co., Ltd. https://blog.csdn.net/qq_35644234/article/details/60875818

[8] Wang Z, 2005, Theory and Application of Control System, Beijing University of Posts and Telecommunications Press, Beijing.

[9] Sheng Z, 2008, Probability Theory and Mathematical Statistics, Higher Education Press, Beijing.

[10] Cheng L, 2000, Course on Models and Methods of Operational Research, Tsinghua University Press, Beijing.