Tourism Management and Technology Economy (2021) 4: 33-37
Clausius Scientific Press, Canada

DOI: 10.23977/tmte.2021.040107
ISSN 2616-2199

# *Desert Route Planning Based on Dynamic Programming*

**Lingnan Xu**

*North China Electric Power University, Baoding, Hebei 071000*

***Keywords:*** optimal strategy, dynamic programming, backtracking

***Abstract:*** For the problem of crossing desert, the choice of path is very important. In this paper, the regions and adjacent parts of a map are regarded as nodes and edges respectively, so that a connected undirected graph is established. Under this circumstance, we develop two strategies to determine the **optimal strategy** to be adopted by players. The first strategy is to solve the Dijkstra algorithm of the shortest path between specified nodes. While the second strategy is to get out of the desert after replenishing resources and obtaining funds through key nodes within a specified period of time. A simplified road map represented by weights is obtained through equivalence, and by using the idea of **dynamic programming**, the problem is abstracted into solving the optimal solution of each stage. Meanwhile the resource allocation of initial knapsack is considered through 0-1 programming, and the idea of **backtracking** is used. Under the constraint of the upper limit of load, all feasible schemes are considered to obtain the maximum profit.

## 1. Introduction

In a game to cross the desert, players can start from the starting point with a map and use the initial funds to purchase a certain amount of water and food. They can also replenish funds or resources in mines and villages. Players will encounter different weather on their way across the desert, but they need to reach the destination within the specified time and keep as much money as possible.

To facilitate analysis, each area in the map is regarded as a node, and each pair of adjacent parts is regarded as an edge, so that a map represented by nodes is obtained.

Strategy 1: going directly out of the desert in the shortest time, which is equivalent to solving the Dijkstra algorithm of the shortest path between the specified nodes [1], from which the shortest path length from the starting point to the end point can be obtained, and then according to the parameter setting and weather conditions, the remaining funds are calculated at the end of the game.

Strategy 2: going out of the desert after replenishing resources and obtaining funds through key nodes (village, mine) within the specified time. The key to this strategy is to deal with the relationship between the optimal model of the backpack and the optimal model of the route. Divided by the key nodes, four phases are established so that the stay at the ordinary nodes is equivalent to the stay at the key nodes at the beginning of the current phase, and then a simplified road map represented by the weights is got.

## 2. Model Establishment

Regard areas in the map and adjacent parts between areas as nodes and edges, a connected undirected graph is obtained (Figure 1), where the key nodes, including the starting point, village, mine, and the end point, are marked with different colors.

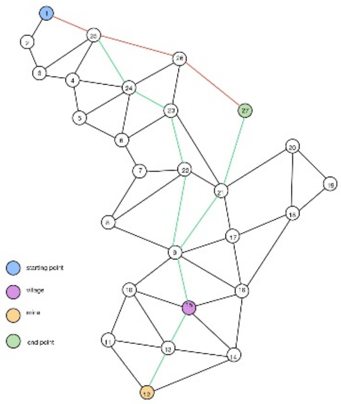


*Figure 1: A connected undirected graph to demonstrate the map*

### 2.1 Strategy One

Strategy one aims only to minimizing the consumption of food and water, without considering the funds earning by mining. Therefore, the best strategy must not include unnecessary stays despite the stays in sandstorms, that is, to follow the principle that the earlier the arrival, the better. It can be seen that:

$$remain = initial - \sum_{i=1}^{n} cost_i \tag{1}$$

Among them, $remain$ represents the remaining funds at the end of the game, $initial$ represents the initial funds, $n$ is the deadline, and $cost_i$ is the value of resources consumed on day $i$ (yuan).

By solving the Dijkstra algorithm of the shortest path between the specified nodes, it can be obtained that the shortest distance between the starting point (1) and the end point (27) of the map is 3, which means players at the starting point need to pass through three areas before reaching the end point. The shortest path is 1→25→26→27.

Thus, according to the weather conditions, we can get:

$$cost_i = \begin{cases} 200, i = 1,2 \\ 190, i = 3 \\ 0, i > 3 \end{cases} \tag{2}$$

Substituting $initial$ with 10000 we calculate the value of $remain$ is 9410.

Therefore, in strategy one, under the condition of meeting the restriction of the initial backpack, player's maximum remaining funds are 9410 yuan, which is taken as a reference standard. If there is at least one path in strategy two, whose remaining funds are greater than 9410 yuan, then it can be concluded that strategy two is better than strategy one.

### 2.2 Strategy Two

#### *2.2.1 Simplification of the Model*

The shortest path between the key nodes is obtained by the Dijkstra algorithm. Since the shortest distance (8) of the starting point → mine is equal to that of the starting point → village (6) adding the distance of village → mine (2), it can be concluded that there must exists a shortest path from the starting point to mine that passes through village. In the same way, there must be a shortest path from mine to the end point passing through village. Hence, the map can be simplified by eliminating unnecessary nodes to narrow the scope of discussion. Furthermore, the simplified map can be divided into 4 stages where the key nodes are regarded as the boundaries. The improved model is shown in Figure 2.

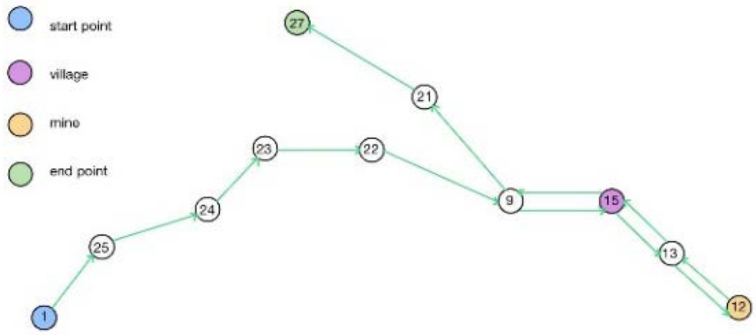


*Figure 2: Simplified route*

According to the simplified circuit diagram, the remaining funds' function defined in the previous section is modified as

$$remain = initial - cost + earn \tag{3}$$

Where, $initial$ is the initial capital (10000 yuan), $cost$ is the sum of capital consumption, and $earn$ is the money earned from mining.

### 2.2.2 Consumption and Acquisition of Funds

Since the remaining funds are directly related to the consumption of water and food and limited by the capacity of the backpack, the water consumption $water_i$ and food consumption $food_i$ are introduced, and the resource consumption under different weather conditions is defined as below:

$$water_i = \begin{cases} 5, i = 0 \\ 8, i = 1 \\ 10, i = 2 \end{cases} \tag{4}$$

$$food_i = \begin{cases} 7, i = 0 \\ 6, i = 1 \\ 10, i = 2 \end{cases} \tag{5}$$

Suppose that the initial backpack only contains the resources necessary to reach the village, while all the other resources needed on the road are purchased in the village. In that way a unique decision-making path is determined, and each day's consumption in the whole process is known. Since the food and water supplies cannot be replenished on the way of village → mine → village, the water-to-food ratio of this part can be obtained, and that of the remaining space in the initial backpack can be deduced. This part's value difference is subtracted from the final consumption.

The improved $cost$ function is:

$$cost = cost - water_{initial} \times 5 - food_{initial} \times 10 \tag{6}$$

The $earn$ function is:

$$earn = \sum_{j=t}^{t'} \frac{(ifwork_j - 1)}{2} \times income \tag{7}$$

### 2.2.3 Problem Solving

Since the starting point → village stage meets the principle of "the earliest arrival", the resource consumption of this phase can be determined directly. Meanwhile, for the stage of village → mine, seven initial states are determined according to different duration of the stay in village. Similarly, mine → village → the end point stage also meets the principle of strategy one, so the resource consumption of this phase is determined on the basis of mine residence time, which is known. Thus, 12 end states are established. Since each state combination uniquely determines a decision-making sequence, the search space is optimized to the Cartesian product of the start states and the end states.

Next, we only need to consider the revenue and expenditure of mine residence period. We define the resource value utilization rate $U_i$ under the $i$th weather condition:

$$U_i = \frac{income}{3\times(water_i\times3)+(food_i\times2)} \tag{8}$$

Through the analysis of resource value utilization rate and net income under various weather conditions, it is determined that during the stay in mine, players must work in sunny and high temperature days to obtain the maximum income. Since the duration of each state in the search space is certain, and the order of the decision of work or rest does not affect the final revenue, the discussion on whether to work or not in sandstorm can be simplified as a combinatorial problem (Table 1).

### 2.2.4 Results

The analysis shows that the minimum consumption is -470 yuan (remaining 470 yuan), which corresponds to the first starting state (departing from the village on the 9th day) and the seventh ending state (departing from the mine on the 20th day). Players work for 1 day and rest for 3 days during the 3 sandstorms. The initial backpack contains 178 boxes of water and 333 boxes of food. For the first time players reaching village, 167 boxes of water were supplied, while for the second time players reaching the village 32 boxes of water and 16 boxes of food were supplied. In the end, the remaining funds are 10470 yuan. After comparing the result with that of strategy one, it is concluded that strategy two is the optimal strategy. The decision is shown in Figure 3:

*Table 1: Sandstorm days*

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 3 | 3 | 3 | x | x | x | x |
| x | x | 0 | 0 | 0 | 2 | 2 | 2 | 2 | x | x | x |
| x | x | x | 0 | 0 | 2 | 2 | 2 | 2 | 2 | x | x |
| x | x | x | 0 | 0 | 2 | 2 | 2 | 2 | 2 | x | x |
| x | x | x | x | 0 | 2 | 2 | 2 | 2 | 2 | 2 | x |
| x | x | x | x | x | 2 | 2 | 2 | 2 | 2 | 2 | 3 |
| x | x | x | x | x | 2 | 2 | 2 | 2 | 2 | 2 | 3 |

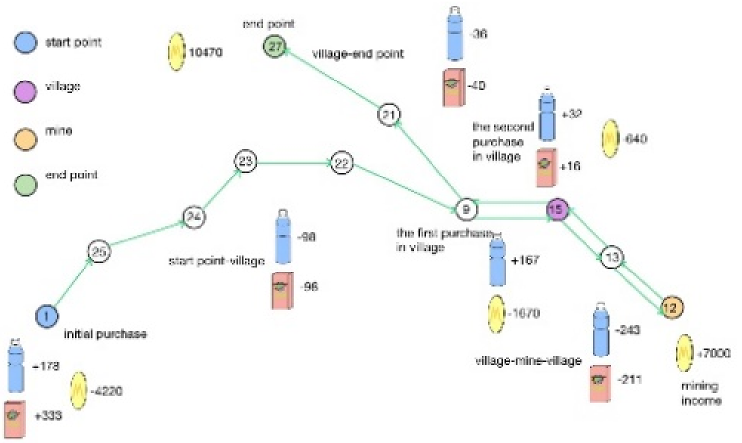


*Figure 3: Decision diagram*

## 3. Evaluation of the Model

When using the dynamic programming model, considering that the optimal solution of the global route must include the optimal solution of its sub-routes, the problem is divided into different stages. Then the problem is discussed under different circumstances, which makes the process of solving linked to each other and gradually improved.

However, there are some limitations. First of all, there is no unified method to deal with the problem. In addition, when the dimension of the variable increases, the total amount of calculation and storage increases sharply. Therefore, the computer still cannot use the model to solve large-scale problems, resulting in "dimension obstacle" [2].

## References

*[1] Xu Guogen, Zhao Housui, Huang Zhiyong. Optimization method and its MATLAB implementation: Xu Guogen, Zhao Housui, Huang Zhiyong, 2018-07*

*[2] Li Shengbing, Li Hangmin. Interpretation of management terminology: Enterprise Management Press, 2007*