# Research on Artificial Intelligence Search Algorithms for Optimization Problem

**Xuefeng Ge**

Yinchuan University of Energy, Yinchuan, 750100, China

**Keywords:** optimization; artificial intelligence; search algorithm

**Abstract:** Search algorithm has a very important influence on the development of artificial intelligence, and it is also a key technology in the field of artificial intelligence. With the continuous development of information technology in our country today, the future of artificial intelligence has broad prospects for development. Different search algorithms have great performance differences, and the applicable scenarios are different, so it is necessary to study them. In reality, the essence of solving many problems is the optimization process. Firstly, the concept of optimization is introduced, and the special role of dynamic programming algorithm in solving optimization problems is analyzed from the perspective of solving optimization problems.

## 1. Introduction

Artificial intelligence has become a very popular field in China in recent years. Nowadays, there are many concepts and definitions of AI in related fields, and there is no unified definition yet. In recent years, the emergence and development of big data, cloud computing and Internet + to a certain extent make China's development of artificial intelligence technology have been promoted. Although the development of artificial intelligence technology is not yet perfect, it has been widely used in various fields. For example, natural language translation in the field of literature retrieval, robots in the field of industry and image recognition and analysis in the field of medical and health care, etc.

## 2. Principle of optimization

The optimal solution of a problem depends on the optimal solution of the sub-problem of the problem. In other words, if it is not the optimal solution, it will not affect the solution of the problem. In the process of solving many problems or solving them, it is essentially a process of optimization. If a problem is decomposed into many sub-problems, the optimal solution of these sub-problems will mean the optimal solution of the problem. Search refers to a problem that does not understand or have not heard of, and then the process of solving the problem on the network is search. The most basic problem in AI technology is search. Search technology has been fully infiltrated in all fields of our country, so in a sense, without search, artificial intelligence will probably not appear [1].

## 3. Dynamic Programming Algorithms

### 3.1 Dynamic Programming

#### 3.1.1 Examples

In order to understand the optimization problem effectively, we need to understand the dynamic programming algorithm deeply. To understand the dynamic programming algorithm, we need to have a deep understanding of several key terms. As shown in Figure 1.

  DOI: 10.25236/mmmce.2019.042

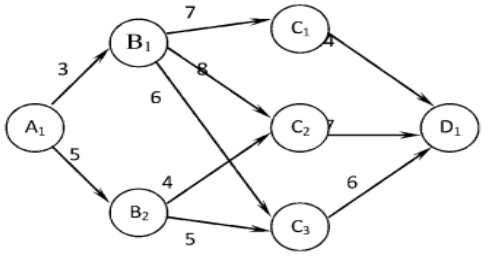


Figure 1 Dynamic programming algorithm diagram

In Figure 1, if the four points A, B, C and D represent the location of a scenic spot, tourists go from scenic spot A to scenic spot D, and then choose an optimal path.

All points are divided into A, B, C and D. All edges in the graph are connected with two adjacent points, and they also have the characteristics of pointing from the former point to the latter point. The edges in the graph can be divided into three categories: A→B, B→C and C→D. First, we need to select an edge in each category, and then combine these edges to make it the shortest path from A1 to D1. Then the shortest road strength is the best path [2].

### 3.1.2 Key concepts of dynamic programming

Usually, dynamic programming algorithm covers the basic relations and attributes.

Stage: It refers to the fact that some properties are very similar and can be processed at the same time. Generally speaking, when dividing the stage of a problem, it can be dealt with in sequence. But in fact, stages only produce a sense of identification for the same processing methods, so they have nothing to do with the processing order. A stage can contain not only one state, but also two or more states.

Stage: Represents the essence of things, mainly used to describe the amount of units in dynamic programming.

State Transfer Equation: The state of the former stage is transferred to the latter one, and a series of evolution laws in this process are state transfer equation.

Decision: A selective activity adopted at each stage, which is very important in the system and must be completed by the system.

Dynamic programming usually deals with multi-stage decision-making problems, starting from the initial state, then choosing the decision-making in the intermediate stage, and finally reaching the end state. This decision will make the decision sequence to be formed, and can also determine the activity route in the process, and the determined activity route is the optimal route.

The premise of using a motion planning is that no matter what the initial state of the optimal strategy and the initial decision of the process are, for the state that has been formed, all the decisions in the future can constitute the optimal strategy. This represents the principle of optimization, that is, the optimal strategy [3].

## 3.2 No aftereffect

A problem is divided into stages. If you want to get the state in stage I, you can use the state transition equation to proceed according to the state in stage I + 1, so there is no relationship with other states.

Generally, the optimization principle can be satisfied in the multi-stage decision-making problem of informatics, but there are often obstacles in the aftereffect.

## 3.3 Optimal Index Function and Programming Equation

### 3.3.1 Optimal Index Function

The essence of the optimal index function is to pay attention to the solution of the problem.

### 3.3.2 Programming equation

Usually, it is a recursive formula from the target state, and then it becomes a programming

equation, which is the solution of the optimal index function. Inversion of the sequential method and backward-forward inverse sequential method will be more suiTable for the problem that the target state has been determined. There are many initial states that have been determined in informatics problems. Of course, for the problems that have been determined, the sequential method can also be used, that is to say, the method of pushing forward and backward in order does not have much difference between the two methods [4].

## 4. Basic Search Algorithms

### 4.1 Backtracking Algorithms

One of the most basic algorithms in search algorithm is backtracking algorithm, which takes the idea of turning direction as the control structure immediately if one path is impassable. This algorithm can be divided into non-recursive algorithm and recursive algorithm [5].

#### 4.1.1 Non-recursive algorithm

Although the structure of the recursive algorithm is relatively simple, the implementation is also very simple, and it has the advantage of readability, but the recursive algorithm has a disadvantage, that is, low efficiency. If the recursive hierarchy is gradually deepening, the stack space occupied will correspondingly become more. Therefore, if the depth of recursion exceeds the stack space, it is not appropriate to adopt recursive algorithm. Recursive algorithm and non-recursive algorithm present relative state in these two algorithms. The non-recursive algorithm is described as follows:

```
Node( Node type )=Recod
Situtation:TSiuation( Current node status )
Way-NO:Integer
End
List( Backtracking table ):Array[1..Max( Maximum depth )]of  Node;
pos( Current Extended Node Number ):Integer;
List<-0;
pos<-1;
List[1].Situation<Initial state;
While( pos>0(There is a way to go) )and( Failure to reach the target )do
Begin
If pos>=M ax then ( Data overflow, jump out of main program );
List[pos].Way-NO<=TotalExpendMethod )then( If extended rules are not used )
Begin
If ( can use the current extension rule )then
Begin
( Using the way Extension Rule )
List[pos+1].Situation:=ExpendNode( List[pos] ).Situation
List[pos].Way-No);
List[pos+1].Way-No:=0;
pos:=pos+1;
End-If;
End-If
Else
Begin
pos:=pos-1;
End-Else
End-While
```

#### 4.1.2 Recursive algorithm

The essence of recursive algorithm is to simplify complex problems before solving them. The recursive algorithm is described as follows:

```
Procedure BaackTrack(Situation:TSituation,deepth:Integer);
```

```
VarI:Integer;
Begin
If deepth > Max then( Space has reached its limit, jump out )
If Situation = Tanget then(find the target );
For I = 1 to To talExpendMethod do
Begin
BaackTrack( ExpendNode(Situation,I),deepth + 1);
End-For;
End
```

### 4.2 Depth search algorithm

Deep search algorithm can produce two kinds of understandings, namely, narrow and broad understandings. The latter understanding is that the newly generated node or the node with the greatest depth needs to be expanded first. The depth search algorithm has two kinds of nodes with all reservations and not all reservations. The former is to preserve all the algorithms that generate nodes. This paper adopts the latter understanding, so it will not retain all the nodes, and belongs to the general backtracking algorithm category in many algorithms. The depth search algorithm can be summarized into three aspects:

#### 4.2.1 Depth search algorithm without reserving all nodes.

After deleting the extended nodes in the database, the stored nodes or depth values will be reduced accordingly, which will make the space occupied to a certain extent. Therefore, if there are more nodes in the search tree, using depth search algorithm is a more effective algorithm, and using other parties. Law is prone to memory overflow [7].

#### 4.2.2 The depth search algorithm has recursive and non-recursive design methods.

When the search depth is small and the recursive method of the problem is obvious, the program structure will become more convenient, concise and easy to understand. However, if the search depth and the amount of data are large, the stack capacity of the system will be limited accordingly, which will lead to recursive overflow. In this case, the non-recursive method is better than the recursive method in design.

### 4.3 Breadth search algorithm

The deep search algorithm and the breadth search algorithm on the control structure and the generation system have very similar characteristics, and the only difference between the depth search algorithm and the breadth search algorithm is that there are differences in the selection of extended nodes. These two search algorithms are mainly aimed at expanding and extending all the sub-nodes in one of the nodes, but there are also differences between the two search algorithms. The depth search algorithm is one of the sub-nodes expanded next time, and the depth search algorithm will extend all the sub-nodes. The successor nodes are retained, so when the successor nodes are generated, the depth search algorithm can remove some of the duplicate nodes, which can effectively improve the search efficiency. In order to improve the search efficiency, different data structures are usually used. The breadth search algorithm can be summarized as follows:

(1) When a new sub-node is generated, if the depth of the node is smaller, then the node will produce its sub-node first, so this means that the node will be the first to be extended. In order to effectively implement the breadth search algorithm, the database storing nodes usually adopts queue structure.

(2) The breadth search algorithm stores all generated nodes, but this means that the storage space occupied will be very large. Therefore, in the process of designing the program, we must give full consideration to the overflow of storage space and memory saving.

(3) Comparing the depth search algorithm with the breadth search algorithm, we can see that the breadth search algorithm belongs to the non-backtracking operation, which means that the breadth

search algorithm does not have the two operation processes of stacking and stacking. This means that breadth search algorithm has two less steps than depth search algorithm, so breadth search algorithm runs faster than depth search algorithm.

(4) When the dissipation value from the node to the root node is proportional to the depth of the node, the solution is the optimal solution; if it is not proportional, the solution is not the optimal solution [10]. There are several main ways to get the optimal solution, for example, to improve the depth search algorithm or the breadth search algorithm. In detail, it is necessary to find a target first, then record the path and dissipation value of the target node, rather than exit immediately. In addition, if there are multiple target nodes, the comparison between multiple target nodes is carried out, and the better nodes are selected from the comparison results.

## 5. Conclusion

In summary, the performance of search algorithm plays an important role in the development of artificial intelligence. Artificial intelligence is still in the ascendant in our country. The development of artificial intelligence will have a great impact on the development of human society. With the continuous innovation in the field of artificial intelligence in China, the value of artificial intelligence has a great influence. This paper mainly analyses the dynamic programming algorithm, and then elaborates the representative algorithm in the basic search algorithm. With the continuous innovation and development in the field of artificial intelligence and information technology in China, it is believed that new algorithms and technologies will emerge in the near future. Therefore, the current research in the field of artificial intelligence in China can not stop.

## References

[1] Chen Xi. Study on the optimization and management of electricity purchase in Jiangxi Province based on the electricity markets[D]. Nanchang: Nanchang University, 2014.

[2] Single-learning, Yang Qinghong, Jolie. Application Research on Recursion Question's Non-recursion Realization Method[J]. Computer and Modernization, 2011, (01): 25-28.

[3] Zhang Guo. Research on Non-recursive Implementation Based on Recursive Algorithms [J].Journal of Yangtze University Natural Science Edition, 2009,6(03): 223-225.

[4] Yang Chunhua, Yao Jin, Zhao Peiying, et al. A non-recursive algorithm framework for recursive algorithms [J]. Computer Applications and Software, 2010, 27 (09): 81-84.

[5] Chen Shengqun, Teng Zhongjian, Hong Yan, etc. Examples of four shortest path algorithms [J]. Computer Knowledge and Technology: Academic Exchange Edition, 2007, 3 (16): 1030-1032.

[6] Fang Jincheng, Zhang Qishan. Overview of Vehicle Routing Problem (VRP) Algorithms for Logistics Distribution [J]. Journal of Shenyang Institute of Engineering: Natural Science Edition, 2006, 2 (04): 357-360.

[7] Zhao Hequn, Wang Pengyu, Li Lei. Parameter selection and convergence analysis of tabu search algorithm [J]. Automation technology and application, 2013, 32 (02): 28-33.

[8] Ouyang Sheng, Hu Wangyu. Research and application of several classical search algorithms [J]. Computer System Applications, 2011, 20 (05): 243-247.

[9] Zhang Guo. Research on non-recursive implementation based on recursive algorithm [J].Journal of Yangtze University: Natural Science Edition, 2009,6(03): 223-225.

[10] Wei Bin, Ma Jihui, Niu Hu. Design and implementation of tree structure diagram based on recursive algorithm [J]. Computer application and software, 2011, 28 (01): 96-98.