Path Planning of Architectural Robot Based on Active Inference Algorithm

Xun Li [*], Tongying Guo

Shenyang Jianzhu University, Shenyang, China

Abstract

In order to solve the problem of slow convergence of traditional reinforcement learning algorithm, a quasi-reinforcement learning algorithm Active Inference algorithm is proposed to solve the path planning problem of construction transportation robot. Simplify the complexity of the construction site and construct a two-dimensional grid map. In the improved Frozen-lake environment, the same starting point and end point are set in the same building map, and the algorithm and the Q-Learning algorithm with different parameters are simulated respectively. The results show that in the case of a given step size, the three algorithms can converge to get the equivalent optimal solution, while the Active Inference algorithm converges faster and obtains the optimal solution early.

Keywords

Active Inference algorithm; reinforcement learning; path planning; architectural robot; Q-Learning; grid map;

1 Introduction

As the pillar industry of our country, the total output value of the national construction industry will reach 29 trillion yuan in 2021[1]. The complex construction site brings many unsafe factors, and casualties occur from time to time. Construction has become the third largest factor of casualties after coal mining and traffic accidents [2]. Heavy physical work and construction sites with hidden safety dangers are eroding the physical and mental health of the construction personnel. Under the strategic background of "Construction Industry 4.0 [3]" put forward by our country, combined with the industry characteristics of the construction industry, a new production mode with high degree of freedom, high flexibility and the combination of personalized and digital construction products and services is established. Design and develop machines and equipment with targeted functions in the building construction environment, which can be assisted by manual instructions or predetermined procedures instead of manual work, that is, construction robots. The strategy is considered to be an effective way to solve difficulties and problems in the future development of the construction industry. The introduction of construction robot not only makes the construction process more safe and reliable, but also improves the construction efficiency, reduces human labor and reduces the comprehensive cost, which has become the inevitable choice to ensure the safety of personnel and improve the quality of the project. At present, robots have been widely used in industry, national defense and smart homes. Although a large number of machinery and equipment have been put into use in the construction site, the construction industry is still a labor-intensive industry, and construction operations mostly rely on manpower. As an important link in the process of construction industrialization, construction robot is still in its infancy in China. Take the European and American standard housing as an example, the construction cycle of traditional manual construction operations is 6 to 9 months. If construction robots such as reinforced material manufacturing (that is, 3D printing technology) are used, the construction cycle can be reduced to about 2 days at the earliest[2].

AIoTC2022@International Conference on Artificial Intelligence, Internet of Things and Cloud Computing Technology

EMAIL: [*]Corresponding author's email:61355474@qq.com (Li Xun)

Path planning [4] is one of the research hotspots of mobile robot technology. When the map is known or unknown, providing the optimal solution of the collision-free path for the robot is an important method in the navigation technology of the mobile robot, and it is also the most important for the mobile construction robot with autonomous operation function. Path planning methods can be divided into two categories: global path planning and local path planning [4]. The path planning algorithm proposed in this paper belongs to the former, that is, path planning under the condition of known environmental information (map). Local path planning is that when all or part of the environmental information is unknown, the robot carries sensors to collect environmental information and carry out path planning in a dynamic environment. Path planning algorithms mainly include traditional algorithms based on search and graph theory, heuristic algorithms, intelligent bionics algorithms, reinforcement learning algorithms and deep learning algorithms, as well as the fusion and improvement of the above algorithms[4].

Traditional reinforcement learning algorithm is a kind of artificial intelligence algorithm which does not need prior knowledge. Under the premise of a given reward function, it is brought into the environment for many iterations to optimize the strategy. The Active Inference algorithm proposed in this paper is an artificial intelligence algorithm similar to reinforcement learning, which optimizes the steps of artificially constructing reward function compared with traditional reinforcement learning algorithm, so as to eliminate the potential influence of reward function with personal preference on the results. By constructing a two-dimensional raster map to simulate the construction site and importing the map into the Frozen-lake environment, the algorithm proposed in this paper is verified in the improved Frozen-lake environment.

2 Environment Construction

The raster method is used to establish two-dimensional raster map to construct environmental information. When the two-dimensional plan of the building is known, using raster map for path planning has the following advantages. First of all, the building plan can be easily converted into a raster map by means of image processing, such as binarizing the pixels of the building plan to get a black-and-white pixel map. Black and white pixels represent obstacles (walls, materials, etc.) and open spaces (free movable space, Free Space). Grayscale image processing can also be carried out to indicate the mobility of the space referred to by the pixel and whether it can be passed by the robot according to the gray value; secondly, the grid map can represent the complex and unstructured environment in a relatively simple way, which can effectively reduce the complexity of environmental modeling, and it is more convenient to judge the relative position of obstacles and robots. Therefore, the grid method has been widely used in path planning [4]. Taking a typical building construction site as an example, as shown in figure 1, areas 1 to 5 on the left are different material storage sites, and areas An and B are construction areas, build a grid map with 12 grids in length and width, white color block as a free movable space area, black color block as a wall or obstacle, collision should be avoided and a two-dimensional grid map is generated as shown in figure 2.

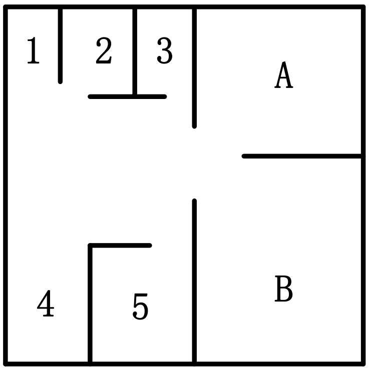

Figure 1 Typical building construction site

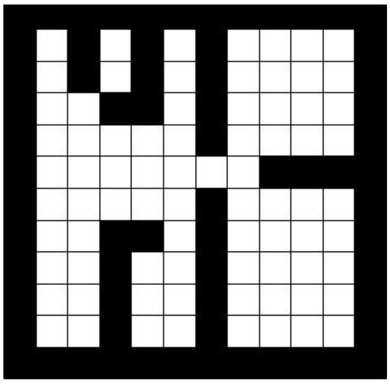

Figure 2 2 Two-dimensional raster map

The mobile mode of the robot is defined as the step-by-step mode of adjacent grids, and the movable distance is one grid at a time, as shown in figure 3. When the robot is located at O point, there are eight directions from A to H. combined with the actual situation, only four directions are selected, that is, A, B, C and D are shown in the picture.

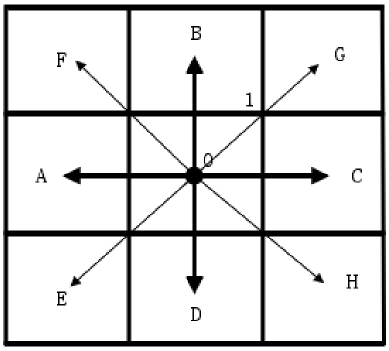

Figure 3 Step direction of robot

The raster map is represented in the program in the form of an array matrix. G (i ,j) = 0 indicates that the grid area is a movable space region, that is, a white color block that can be passed by the robot, and G (i ,j) = 1 means that the grid area is a wall obstacle, that is, a black color block that should avoid collision.

3 Active Inference Algorithm

Active Inference is a theory proposed by the theoretical neuroscientist Karl Friston to speculate, understand and describe the perception, plan and behavior of the brain according to probability, which provides a comprehensive perspective to explain the coordinated relationship among brain, cognition and behavior [5]. The core of Active Inference is to transform behavioral action into perception based on a single command (Imperative) that minimizes free energy. The theory is considered to be the "first rule" for understanding the brain and behavior. It also explains how biology or artificial intelligence (Artificial) carries on cognition and reasoning in unsteady (Non-stationary) environment [6].

As an indispensable part of reinforcement learning, reward signal is not indispensable in Active Inference algorithm. Reward is regarded as some kind of Observation result of our Preference, and even when there is no reward at all, the behavior of Agent is learned through preference. Compared with the Q-Learning algorithm in the same test, the Active Inference agent can infer the behavior in the non-reward environment, and learn similar considerable preference (Prior Preferences) to the algorithms that contain reward signals, such as Q-Learning, by putting zero preference above reward. In the Active Inference algorithm, the interaction between the agent and the environment is determined by the action sequence that minimizes the expected free energy, rather than the expectation of the reward signal in reinforcement learning. In standard reinforcement learning, the reward function defines the agent's subject goal and allows the agent to learn how to best act or get the optimal solution in the environment [7]. Reward in reinforcement learning can be seen as a scalar signal, and any Goal or Purpose is achieved by maximizing the sum of the cumulative expected values of the scalar signal [8]. The reward result of a single study is output to the action and behavior of the study. The repeated learning cycle for many times, gradually approaches the maximum value of the sum of reward expectations, which is the final output of the reinforcement learning. The difference of reinforcement learning is that the difference of scalar signals will not affect the final output of Active Inference algorithm, that is, different reward functions in the same environment will not lead to differences in learning results. In other words, any implicit reward related to results in reinforcement learning comes from the considerable features observed by the agent, rather than the features obtained from the environment through learning (reasoning), including the non-objective parts.

The variational free energy formula is derived from a simple result generation model, which is the starting point of the derivation process of Active Inference algorithm. For example, in figure 4, the random variable $s \in S$ is introduced to represent the specific hidden state of the environment, where S is a finite set of all possible hidden states, and the random variable $o \in O$ is introduced to represent the specific results received by the agent. The O is a finite set of all possible results. The generation model abstractly believes that there is a real hidden state $s^* \in S$ in the world. The o is generated by the generation process, and the agent deduces s from the generation model. Hidden state is a combination of agent-related features (such as location, etc.), and the result is information from the environment (such as feedback, speed, reward, etc.). Through the reverse process of mapping the hidden state to the result, the agent explains how the result is caused by the hidden state, that is, the inversion Bayesian model for reasoning.

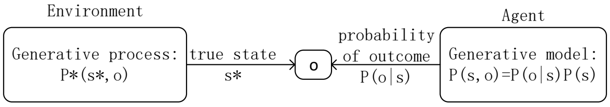
Environment
Generative process: P*(s*, o)
true state s*
o
probability of outcome P(o|s)
Agent
Generative model: P(s, o)=P(o|s)P(s)

Figure 4 the relationship between the generation model and the process

The generation model is defined as a partially observable Markov decision process, which contains the joint probability P(o|s) consisting of the result $o \in O$ and the state $s \in S$. The joint probability can be decomposed into likelihood function P(o|s) and internal state P(s).

$$P(o，s)=P(o|s)P(s) \tag{1}$$

In order for the agent to converge stably, we need to edge all the states that may lead to a given result, decomposed by equation 2.

$$P(o)=\sum_{s \in S} P(o \mid s)P(s) \tag{2}$$

Because the dimensions of hidden state and action sequence space may be very large, and the calculation of the above formula may be extremely complex, we consider using variational method to approximate P(o), which is more feasible and easier to deal with. Therefore, we introduce an important concept, difference[9] (Surprise or Surprisal, which can also be called Shannon information quantity Shannon Information Content), which is used to measure the uncertainty of information. In order to distinguish it from state S, we use C to represent difference. Define the probability that the difference is the negative logarithm of outcome, such as formula 3. The P represents the probability distribution.

$$C(o)=-\log P(o) \tag{3}$$

The variational free energy F is defined as the upper limit of the difference, which is derived from the Jensen inequality. In the literature of variational reasoning, it is usually called the (negative) evidence lower bound(ELBO)[10].

$$C(o)=-\log \sum_{s \in S} P(o,s) \tag{4}$$

$$\leq -\sum_{s\in S} Q(s)\log\frac{P(o,s)}{Q(s)} \tag{5}$$

Q(s) is a variational distribution. The concrete expression of variational free energy F is obtained by further derivation.

$$F = \sum_{s\in S} Q(s)\log\frac{Q(s)}{P(o,s)} \tag{6}$$

$$= D_{KL}\left[Q(s)\,||\,P(s\,|\,o)\right] - \log P(o) \tag{7}$$

The relationship between the difference C and the variational free energy F is shown below, and the KL divergence is always greater than zero.

$$C(o)=F - D_{KL}\left[Q(s)\,||\,P(s\,|\,o)\right] \tag{8}$$

Variational free energy provides us with a way to perceive the environment, that is, to determine the environment state s from the output o, which is also the process of Inference to the environment. But at present, there is still a lack of action in the above equation, so we introduce a new variable and define the action strategy as π ∈ ﬂ, where π is a finite set of policies that may be allowed, so that the agent can transfer the state directly in the hidden state. On the condition that formula 6 and formula 7 bring in strategy π, we further derive the available formula as follows:

$$F = D_{KL}\left[Q(s\,|\,\pi)\,||\,P(s\,|\,o,\pi)\right] - \log P(o) \tag{9}$$

$$F(\tau,\pi) = \sum_{s_\tau^\pi} Q(s_\tau|\pi)Q(s_{\tau-1}|\pi)\log\frac{Q(s_\tau|\pi)}{P(o_\tau,s_\tau|s_{\tau-1},\pi)} \tag{10}$$

These strategies are the actions that the agent will take. In the process of interacting with the environment (that is, the agent learns in the environment), the action sequence affects the result to get the reasoning cognition of the environment, while the environment and the result affect the next action sequence. Through repeated iterations with predictability, the Active Inference algorithm obtains an action sequence composed of strategy Pi to minimize the free energy on the premise of keeping the difference minimized to maintain the intelligent body steady state[11]. The learning result output under this condition is the optimal result we expect.

4 Simulation and results

Frozen-lake is an environment for training and testing agents launched by OpenAI in 2016. The previous initial version of Frozen-lake, as shown in figure 5, contains a grid map of 3x3 and three states, including a frozen lake (gray grid), a hole (grid with irregular blue shapes), and a destination (grid with pentagram). The starting point is fixed to the upper left corner of the map, through the frozen lake and avoid falling into the ice hole, reaching the destination is victory.

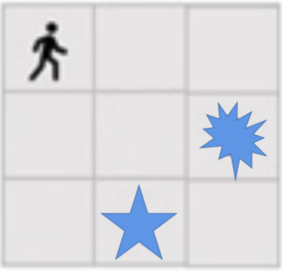

Figure 5 The previous initial version of Frozen-lake

The agent runs in an improved Frozen-lake environment, in which four states are defined: starting point, ice, wall, and end point. Of all the states, only the "wall" is the unsafe state, corresponding to the "hole" in the initial version. Each training (Episode) starts from the "starting point" , and "ice" is a grid through which the agent can safely pass. When the agent moves to the grid where the "wall" is located, the task fails and the training ends and returns to the initial position "starting point" to start the next round of training. The agent starts from the "starting point" and moves safely to the "end point" through the "ice". The task completes the end of the training and gets 100 points. This scoring standard is not the reward function of the agent, but the score set to facilitate comparison with other reinforcement learning algorithms. After the training, the Frozen-lake environment gives a score according to the learning results of the agent, and the agent with faster convergence speed can get a higher score. This paper chooses Q-Learning algorithm citation to compare with Active Inference algorithm. In order to compare the effects of the two algorithms intuitively and efficiently, the Frozen-lake environment is improved and the raster map of figure 2 is optimized to create a 12-12 grid map as shown in figure 6.

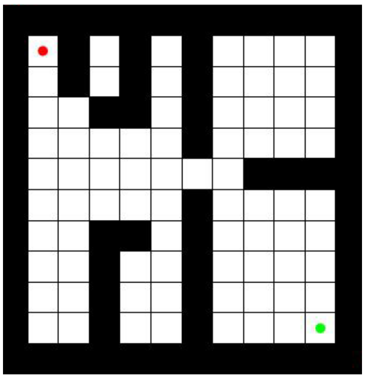

Figure 6 12*12 raster map

In this environment, the generation model of Active Inference agent is defined as follows, as shown in figure 7, there are four action states: upper, lower, left and right, and are not allowed to stay in place, and each motion state will move the agent to a different location. The starting point is defined at position 14, the end point is defined at position 131, the dark grid is a dangerous grid wall (the task fails when the agent moves to the "wall" grid), and the light grid is safe grid ice; we limit the maximum number of training movements to 50. The action state controls the agent to switch between position states, for example, at the starting position 39, the agent can move to position 27 (up), position 51 (down), position 38 (to the left) or position 40 (to the right). Position 27 and position 40 are walls, so the agent is judged to be the end of the round after moving to the above two positions, and the task fails. Both position 38 and position 51 are safe areas and can be moved again. The agent infers and observes two types of results at each point in time: the current grid state and the score.

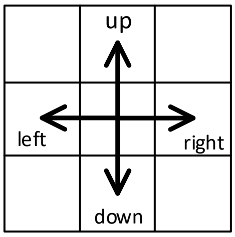

Figure 7 Motion direction of robot

This paper compares the performance of Q-Learning algorithm in the same Frozen-lake environment. Perhaps the more advanced and complex reinforcement learning algorithm will perform better under the same conditions. In order to compare with the Naive reinforcement learning algorithm under the standard framework, the Q-Learning algorithm is chosen. Q-Learning algorithm uses ε-Greed strategy, define greed = 0.1, learning rate A = 0.5, discount factor γ = 0.99. At the same time, in order to compare the influence of artificially defined reward function on agent learning results, two different reward functions are defined under this condition. The first is the reward function with cost, that is, one point is deducted for each move; the other is cost-free, and no points are deducted for a single move and moving to a dangerous area. The Q-Learning algorithm and Active Inference algorithm with two different reward functions both reward 100 points when moving to the end point, and the number of moves in a single round is limited to 50 steps, and is not allowed to stay in place.

The above three algorithms are run in the improved Frozen-lake environment, and each agent undergoes no more than 200 rounds of training. Within this framework, all three agents can begin to converge within 120 rounds of training and successfully reach the end point. After training, all agents can output several equivalent shortest paths.

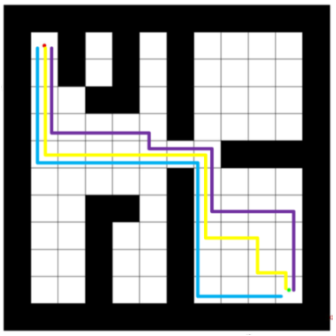

Figure 8 Partial equivalent shortest path output

In all the results of the output, taking the three paths shown in figure 8 as an example, the three paths all pass through 18 grids, purple and yellow paths all have six turns, while the blue path has only three turns. Combined with the actual situation, the less the number of turns, the shorter the working time of the robot, and the higher the transportation stability. At present, none of the three algorithms can distinguish the advantages and disadvantages of the three paths, so they can only select the shortest path from the number of steps, and the 18-grid path is the shortest path under the map. Among them, the average score of Active Inference agent in training is 81, the average score of Q-Learning agent with cost is 77, while the performance of non-cost Q-Learning agent is the worst, with an average score of 72. The convergence speed of Active Inference agent is faster than that of Q-Learning agent. Because Q-Learning algorithm will lead to serious dimension disaster in large map environment[12], and Active Inference algorithm can avoid dimension disaster very well, it can be inferred that the latter will perform better when realizing path planning function in a larger range of maps.

5 Concluding remarks

Active Inference algorithm is a relatively new algorithm that has been applied in practice in recent years, and it is also the first time to apply it to the robot raster map path planning task. There are similarities between the reinforcement learning algorithm based on Bayesian model and the Active Inference algorithm, which are different from the reinforcement learning algorithm in some key points. In reinforcement learning, the goal is defined by the reward function, that is, the scalar signal from the environment. On the contrary, the target in the Active Inference algorithm is defined by the agent's a priori preference for the result. In contrast, the Q-Learning algorithm seems to be more sensitive to the shaping of the reward function, and the Q-Learning agents under different reward functions show different performance, which also shows that the artificial reward function is subjective and directly affects the output of the agent. It is not ruled out that through reasonable algorithm optimization or setting a more reasonable reward function, or using a better algorithm, reinforcement learning agents may perform similar to or better than Active Inference.

Through simple experiments, we make a preliminary comparison between Active Inference algorithm and naive reinforcement learning algorithm. The research content of this paper is relatively limited, for example, the time consumption of turning has not been distinguished and only the length of the path has been considered. Under the background of this application, we can explore the potential of Active Inference algorithm more deeply. For example, more complex coding methods can be used to give more meaning to the environment, path planning can be carried out in semantic maps with more complete environmental content, or more complex tasks can be learned in non-stationary or even unstructured environments. This is still a more prominent problem at present.

6 References

[1] Ma Zhiliang M.: Interpretation of the Development report of Informatization in China's Construction Industry (2021) Application and Development of Intelligent Construction, China's construction informationization. Vol. 24, 2021, pp.12-15

[2] Qi Jialin D.: Research on obstacle Detection and obstacle avoidance Strategy of Building handling Robot, Xi'an University of Architectural Science and Technology. 2020

[3] Shen Mingxin D.: Research and implementation of Autonomous Positioning and Navigation system for Indoor Construction Robot, University of Electronic Science and Technology. 2020

[4] Sun Shangjie, Jiang Shuhal, Cui Haohe J.: Cui Haohe. Path Planning of Forest Fire fighting Robot based on Deep Learning, Forest engineering. Vol. 36, 2020, pp. 51-57

[5] Friston K. J.: Active inference: A process theory, MIT PRESS.2017(1)

[6] Friston K. J.: Deep temporal models and active inference, Neuroscience and Biobehavioral Reviews. Vol. 90,2018

[7] Paper D M.: An Introduction to Reinforcement Learning. 2022

[8] Sajid N J.: Active Inference: Demystified and Compared, Neural Computation. Vol. 90, 2021, pp. 1-39

[9] Zénon Alexandre J.: An information-theoretic perspective on the costs of cognition, Neuropsychologia. 2018, pp. 123

[10] David M.: Variational Inference: A Review for Statisticians, Journal of the American Statistical Association. Vol. 518, 2017, pp. 112

[11] Rafal Bogacz J.: A tutorial on the free-energy framework for modelling perception and learning, Journal of Mathematical Psychology. Vol. B, 2017, pp. 76

[12] Song Qisong D.: Research on Optimization of path Planning algorithm for Mobile Robot, Guizhou University, 2021