UDC 681.3.07

THE METHODS OF SEARCH OPTIMAL CONTROL

A. I. Bronnikov, Kharkiv National University of Radio Electronics

In order to plan the object moving in the program should be laid the algorithm by which the decision on the planning movement is acceptedt. Based on dynamic programming approach solved the problem of finding of the optimal control, which is implemented in software.

Keywords: *dynamic programming, optimization of movement, optimal control, methods of optimal control*

Introduction

The sufficient number of mathematic methods and algorithms for solving the tasks of searching the optimal route of passing the moving object has been developed.

One of the most perspective methods is the method of dynamic programming.

Dynamic programming is the part of optimal programming (optimal control), in which process of making decisions and control may be divided into separate parts (steps).

Dynamic programming allows leading one difficult task with many variables to many tasks with small number of variables. This significantly reduces the amount of calculations and accelerates the process of making management decisions.

The control – complex of decisions, made in every step for influence on course of development process.

The operation – controlled process, means it is possible to choose some parameters that influence on process course and to control operation steps, provide the gain in every step and during the whole operation.

The decision in every step is called “step control”.

The complex of all step controls is the control of whole operation [1].

Statement of the problem dynamic programming

It is necessary to find the control that has minimum costs of made moves:

$$F(x) = \sum_{i=1}^{m} F_i(x_i) \longrightarrow \min \qquad (1)$$

where F – operation cost;

$F_i(x_i)$ – cost of step i;

x – the whole operation control;

x_i – the control in step i (i = 1,2, ..., m).

In general step control ($x_1, x_2 \ldots x_m$) can be number, vectors, functions.

The control (x^*) that has minimum is called optimal control. Control optimality consists of the complex of optimal step controls $x^* = x_1^*, x_2^*, ..., x_m^*$.

$F^* = \min\{F^*(x^*)\}$ – minimal cost, that the optimal control x^* has.

Based on conditions of each specific task, step length has to be chosen in that way so in every step it is necessary to get simple optimization task and provide required accuracy of calculations.

The main method of dynamic programming is the method of recurrence relations that is based on using optimality principle, developed by American mathematician R. Bellman.

The essence of the principle is in the following. Regardless of the initial state in any step and control, the choosing in this step, next controls must be chosen optimal according to state that the system has in the end of every step.

The use of this principle guarantees that control in any step is better not only locally but in general.

Conditional optimization:

$$S_0 \frac{x_1}{step} S_1 \frac{\tilde{o}_2}{step} S_2 \frac{x_3}{step} S_3 ... \frac{\tilde{o}_m}{step} S_m \qquad (2)$$

Unconditional optimization:

s_i – state of system in step i. The main recurrence formula of dynamic programming in case of solving the maximization task is:

$$f_m(i) = \min \left\{ \begin{matrix} f_m(step\_\cos t) + \\ + f_{m+1}(new\_state\_before\_step\_m+1) \end{matrix} \right\}, \qquad (3)$$

where minimum in this formula is found from the all possible decisions in case when system is in step m has the state i.

The variable $f_m(i)$ – minimal cost of completion task of moving from state i supposing that in step m system has the state i.

The maximal efficiency of control can be received from the minimization of sum of cost the step m with minimal cost the step ($m+1$) and till the end of route.

Planning the multistep operation, it is necessary to choose control in every step taking into account all his future consequences in future steps.

The control in step i has to be chosen not only according to the cost of given step is minimal, but the minimal sum of costs in all steps, that left, plus given step.

Among all steps, the last step is being planned without looking in future, so this step brings the greatest benefit and the least cost.

The task of dynamic programming is being solved starting from the end, from the last step. It is being solved in 2 steps:

1 step (from the end to the start step by step).

The conditional optimization is being carried out, in result conditional optimal controls and conditional costs are being determined in all steps of process.

2 step (from the start to the end step by step).

Already prepared recommendations are being chosen (read) from the first step to the last and the unconditional optimization x^* is being determined, $x^* = x_1^*, x_2^*, ..., x_m^*$.

During the formulation of dynamic programming tasks there are some principles such as:

a) to chose parameters (phase coordinates) that characterize state S of controlled system before every step;

b) to divide operation into stages (steps);

c) to find out the set of step controls x_i for each step and impose restrictions on them;

d) to determine the cost of step i control x_i, if the system was in state S before, i. e. write “cost function”:

$$W_i = f_i(S, x_i) \tag{4}$$

e) to determine change of state S system S under the influence of control x_i in step i: it moves to a new state

$$S' = \varphi_i(S, x_i) \tag{5}$$

f) to write the main recurrence relation of dynamic programming that shows conditional optimal cost $W_i(s)$ (starting with step i and till the end) via already known function $W_{i+1}(s)$:

$$W_i(S) = \min_{x_i}\{f_i(S, x_i) + W_{i+1}(\varphi_i(S, x_i))\} \tag{6}$$

The conditional optimal control in step i $x_i(s)$ corresponds to this cost (what is more it is necessary to substitute changed state $S' = \varphi_i(S, x_i)$ to already known function $W_{i+1}(s)$ instead of S;

g) to conduct conditional optimization of the last (m) step using the gamma of states S from which the final state can be reached, calculating for each of them the conditional optimal gain by formula $W_m(S) = \min_{x_{mi}}\{f_m(S, x_m)\}$;

h) to conduct the conditional optimization of (m-1)’s, (m-2)’s etc. steps by formula (6), considering in it i=(m-1), (m-2), …, and indicate for every step conditional optimal control $x_i(s)$ that gets the minimum;

i) to conduct the unconditional optimization of control, “reading” appropriate recommendation on every step. To get optimal control defined in the first step; to change state of system by formula (5); to search optimal control for defined state in the second state x_2* etc. till the end.

Automation of search optimal control

For development the algorithms software it’s been chosen MatLab for mathematical calculations.

M-file GridWorld.m for realization the algorithm based on dynamic programming has been designed.

Each program has next components:

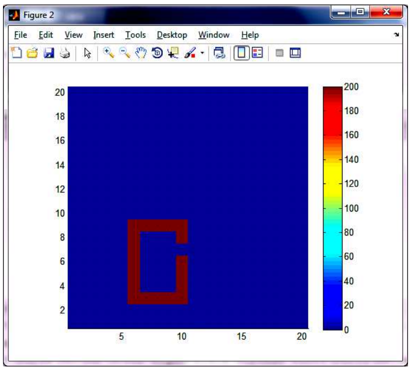
- the determination of variables;
- the introduction of the initial values and limits;
- the realization of algorithm;
- the visualization of results.

The input information that is necessary for program to work:

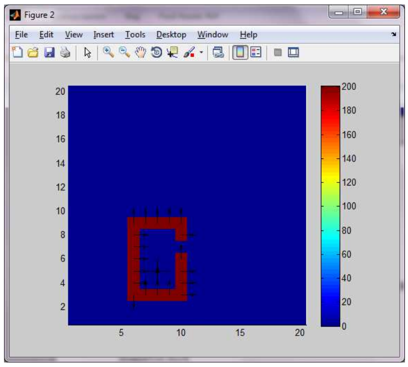
- area size (variable “board”);
- the initial coordinates of object (variable “start”);
- coordinates of movement goal (variable “goal”);
- the information about barriers (variable P).

Consider stepping execution of development algorithm based on the method of dynamic programming. The first step is shown in picture 1 and displays only the barrier and area of movement. In this algorithm the initial position of object doesn’t matter – algorithm calculates optimal trajectories for each area point, starting with the coordinates of goal (picture 2). Interesting is the process of search the exit from limited area in which the goal is.

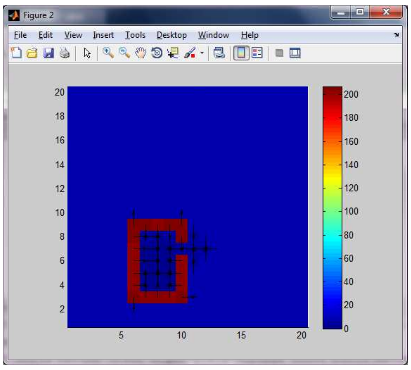
In picture 3 the result of the fifth iteration of calculation is shown, as it is seen, the algorithm is already out of barrier area.



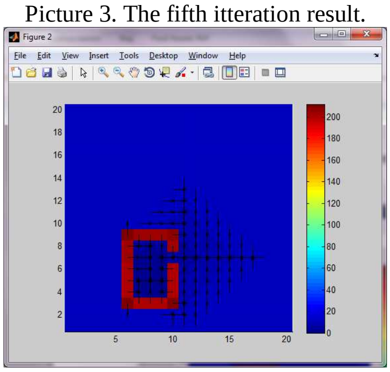
Picture 1. Display of introduced initial variables



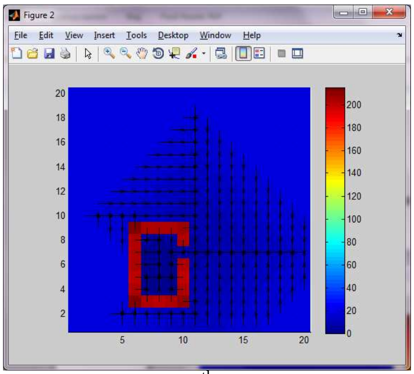
Picture 2. The first itteration result.



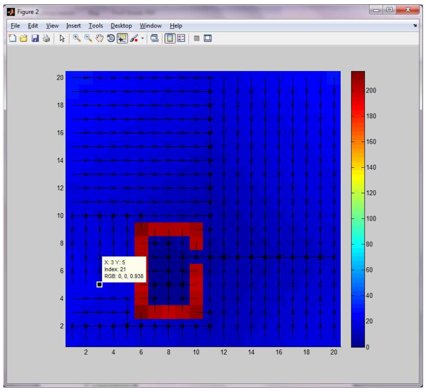
Picture 3. The fifth itteration result.



Picture 4. The tenth itteration result.



Picture 5. The 15th itteration result.



Picture 6. The final result.

It can be seen that cost of movement from the initial object position consists of 21 units (21 steps and 4 turns).

Conclusion

Thus in proposed article the main features of realization method of dynamic programming for designing the optimal route of mobile robot movement have been considered. Conducting experiments of designing the optimal route in program environment MatLab (in different steps of software execution) has been shown.

Bibliography:

1. *MATLAB 7[Текст] / И.Е. Ануфриев, А.Б. Смирнов, Е.Н. Смирнова – Санкт-Петербург, «БХВ-Петербург», 2005р. – 1104с.*
2. *Методы оптимизации в теории управления: Учебное пособие / И. Г. Черноруцкий. — СПб.: Питер, 2004. — 256 с.*
3. *Webinar on topic "Optimization in MATLAB: An Introduction to Quadratic Programming" [Electronic resource] / The Mathworks – Access mode: www/URL:http://www.mathworks.com/company/events/webinars/wbnr62151.html?id=62151&p1=961663319&p2=961663337 .*
4. *Russ Tedrake. UNDERACTUATEDROBOTICS: Algorithms for Walking, Running, Swimming, Flying, and Manipulation CHAPTER 9. Dynamic Programming [Электронный ресурс] / Режим доступа: http://people.csail.mit.edu/russt/underactuated/underactuated.html?chapter6.*
5. *Алгоритмы обхода препятствий - [Электронный источник] – режим доступа http://pmg.org.ru/ai/navigato.htm.*