'88 KACC 1988.10.21~22

APPLICATION OF TREE SEARCH TO A CONTROL PROBLEM

Mitsutaka Miyashita* and Hikaru Inooka**

*Sankyo Seiki Mfg. Co.,Ltd
Nagano, 393 Japan
**Department of Mechanical Engineering
Faculty of Engneering
Tohoku University
Sendai, 980 Japan

Abstract: Tree search can be applied to a control problem if the system is a discrete-time one and if the control input takes only discrete values. This paper considers the application of heuristic tree search to a simple control problem. The results of simulation studies show the good possibility of this approach to a control problem.

1.Introduction

Dynamic behavior of a discrete-time system is described by a tree if the control input takes only discrete values(see Fig.1). Then the problem of finding a control sequence which moves the system state from an initial point to a final goal is reduced to that of tree search.

There have been reported many search procedures in the field of artificial intelligence[1],[2]. In those, breadth-first search (horizontal search) and depth-first search (vertical search) are simple and easily applicable; furthermore, the procedures make it possible to find an optimum solution. However, the time required to perform the search procedures increases with the levels of a tree. In control problems, the levels are large; thus, an efficient search procedure is necessary. A heuristic is a technique that improves the efficiency of a search process,possibly by sacrificing claims of optimum. Using a heuristic function, we choose the node of a tree for further search so that the smallest value of the heuristic function is attained.

This paper considers the application of heuristic search to simple control problems; the landing control of a space vehicle and the swinging-up control of a pendulum. Simulation studies are performed by the use of mathematical models and several heuristic functions.

2.Graph Traverser

There exist some procedures in heuristic search such as A algorithm, A^* algorithm, and Graph Traverser[1]. Here we use G.T.(Graph Traverser) because the algorithm is simple; furthermore, the number of nodes generated during search is comparatively small. We will use following symbols:

- S : Start states
- G : Goal states
- $h(\cdot)$: Heuristic function; $h(n)$ is the value of the function at the node n.
- D : Set of nodes which have been examined already
- $\bar{D}$: Set of nodes which will be examined at the next step; D is connected to D.
- $\Gamma(n)$: Set of nodes which are connected to the node n.

The algorithm of G.T. can be summarized as follows.

(1) Let D be equal to S; $\bar{D}$ is put to $\Gamma(S)$.

(2) Find $n \in \bar{D}$ such that $h(n) = \min_{k \in \bar{D}} h(k)$.

(3) Generate $\Gamma(n)$.

(4) If $G \in \Gamma(n)$, then, the search procedure results in success.

(5) If $G \notin \Gamma(n)$, then, move n from $\bar{D}$ to D. Add $\Gamma(n)$ to $\bar{D}$.

(6) If $\bar{D}$ becomes empty, the procedure falls in failure; otherweise go back to (2).

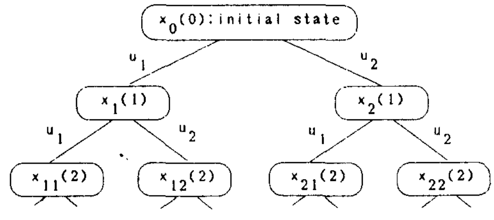

Fig.1 Tree representation of a control problem.

3. Example of Landing Control

Consider the landing control problem of a space vehicle. The altitude $x(t)$ of the vehicle is assumed to be governed by

$$\ddot{x}(t)=u(t) \qquad (1)$$

where $u(t)$ is a control input. Assume that $u(t)$ takes only two values; +1 and -1. The initial condition is

$$x(0)=1 \ , \ \dot{x}(0)=0 \qquad (2)$$

The goal will be

$$x(T)=0 \ , \ \dot{x}(T)=0 \qquad (3)$$

where T is a terminal time.

We will introduce a discrete-time model for (1). Then the condition (3) is not adequate in such a case; we will use the following condition for the goal.

$$x(T) + \dot{x}^2(T)/2 < 0.005 \qquad (4)$$

Assume the following constraints:

$$x(t)>0 \ , \ \dot{x}(t)<0 \qquad (5)$$

Consider the two heuristic functions:

$$h_1(x,\dot{x})=|\dot{x}^2(t)/2 + k_1x(t)| \qquad (6)$$

$$h_2(x,\dot{x})=\dot{x}^2(t) + k_2x^2(t) \qquad (7)$$

where k_1 and k_2 are positive constants. The function $h_1(x,\dot{x})$ corresponds to an energy function, and $h_2(x,\dot{x})$ is a quadratic form.

Now let us apply the G.T. algorithm to the problem; the equation(1) is changed to a discrete-time model with the sampling time of 0.05. Fig.2 and Fig.3 show the number of nodes which are generated during search and the length of final trajectories for various values of k_1 and k_2. In Fig.2 the number of nodes shows the minimum when k_1 is nearly equal to 14; on the other hand, the length of trajectories decreases with k_1. It is clear from Fig.3 that the length of the trajectory takes the minimum value of 39 when k_2 is large; the value is close to the theoretical one of 40 which is attained by time-optimal control.

Fig.4 shows the result of simulation using $h_1(x,\dot{x})$ where k_1=32. Fig.4-(a) is a contour map; note that there exists a contour line corresponding to zero. Fig.4-(b) shows all trajectories generated during search, and Fig.4-(c) is a final trajectory. These trajectories are expressed in the phase plane; Fig.4-(d) shows the time response of $x(t)$ and $u(t)$.

Fig.5 shows the result using $h_2(x,\dot{x})$ where k_2=30. Fig.5-(a) is a contour map; $h_2(x,\dot{x})=0$ only when $x(t)=\dot{x}(t)=0$. From Fig.4 and Fig.5, it can be seen that the pattern of a contour map affects the process of search. Fig.6 shows the result using $h_2(x,\dot{x})$ where k_2=80. Note that this is the time-optimal case(see Fig.3).

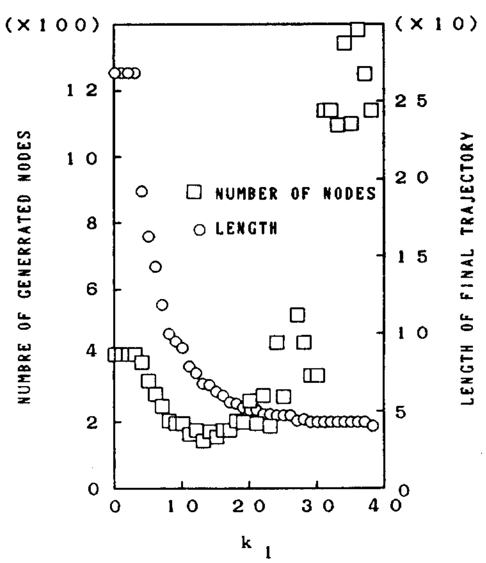

Fig.2 Characteristics of heuristic search using $h_1(x,\dot{x})$.

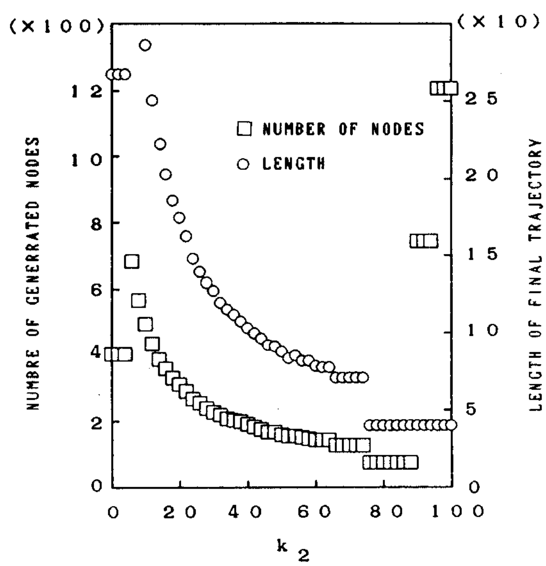

Fig.3 Characteristics of heuristic search using $h_2(x,\dot{x})$.

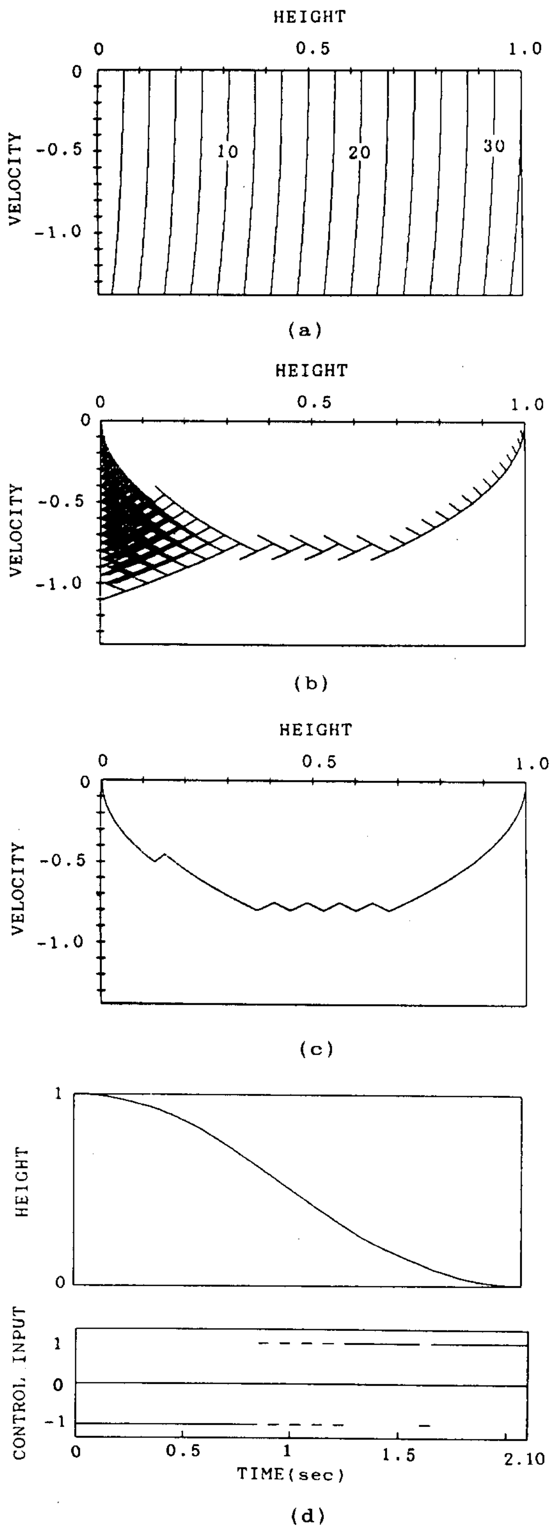

Fig.4 Simulation result using $h_1(x,\dot{x})$ $(k_1=32)$.

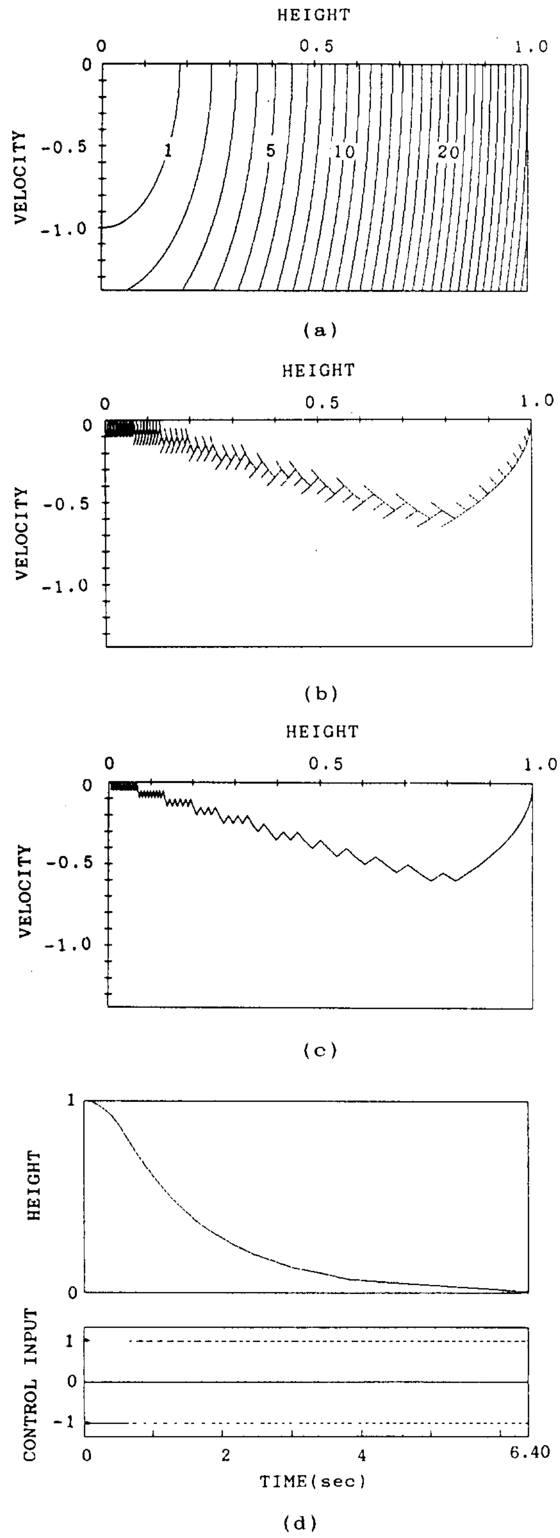

Fig.5 Simulation result using $h_2(x,\dot{x})$ $(k_2=40)$.

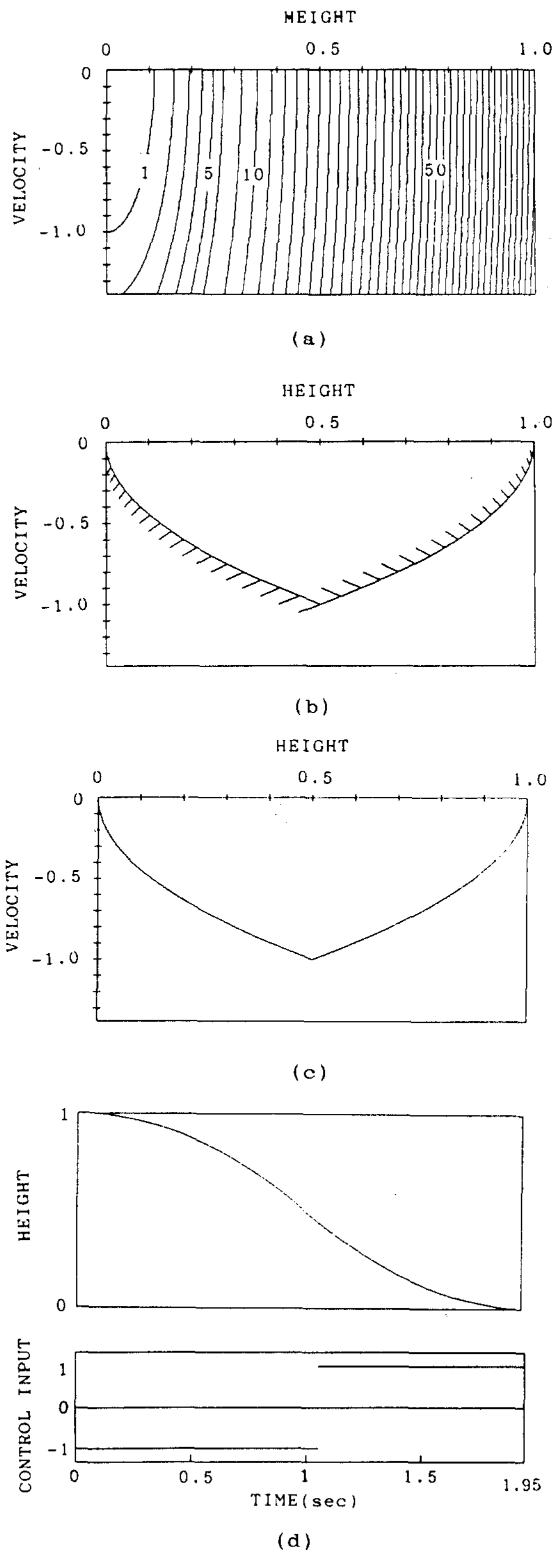

Fig.6 Simulation result using $h_2(x,\dot{x})$ $(k_2=80)$.

4.Example of Swinging-up Control

Consider the pendulum shown in Fig.7. The purpose of control is to move the pendulum from the stable equilibrium point of the bottom position to the top position under the condition that the available torque is limited. This is a highly nonlinear problem; it is difficult theoretically to determine such a control law[3].

Assume that the angle $\theta(t)$ is governed by

$$\ddot{\theta}(t)+2\zeta\dot{\theta}(t)+\sin\theta(t)=\alpha u(t)-\alpha_f\,\mathrm{sgn}(\dot{\theta}) \qquad (8)$$

where

ζ (damping factor)=0.042

α (nondimensional torque)=0.577

α_f (nondimensional friction torque)=0.017

$\mathrm{sgn}(\cdot)$: sign function

The control input u(t) is assumed to take three values;{-1,0,1}. The condition for the goal is given by

$$\pi-|\theta(T)|\leq 0.07,\ |\dot{\theta}(T)|\leq 0.052 \qquad (9)$$

Consider the following heuristic function of an energy function type :

$$h_3(\theta,\dot{\theta})=\left|\frac{1}{2}\dot{\theta}^2(t)-k_3\{1+\cos\theta(t)\}\right| \qquad (10)$$

where k_3 is a positive constant.

Eq.(8) is changed to a discrete-time model with the sampling time of 0.137; then G.T. algorithm is applied. The result is shown in Fig.8. Fig.8-(a) is a contour map; there exists a contour line corresponding to zero. Fig.8-(b) shows the all trajectories generated during search. Fig.8-(c) shows the time response $\theta(t)$ and u(t); note that the direction of the swinging-motion changes only once. The length of

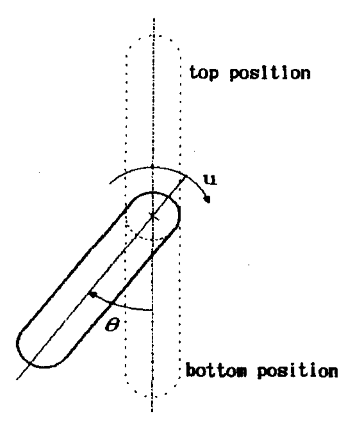

Fig.7 Swinging-up motion of a pendulum.

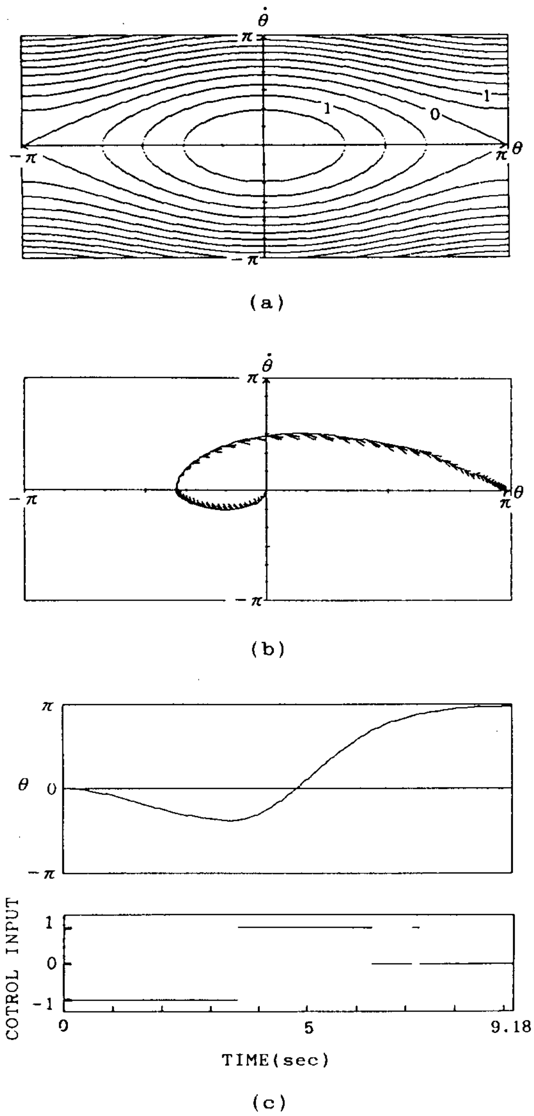

Fig.8 Simulation result using $h_3(x,\dot{x})$ $(k_3=1)$.

the trajectory is 67 steps; it is equivalent to 9.18 sec. This time is 24% longer than the time which is attained by time-optimal control[3]. In the time optimal case, u(t) takes only -1 and 1. On the other hand, u(t) takes three values including 0 as is shown in Fig.8-(c).

5.Concluding Remarks

Heuristic tree search is applied to control problems. The results of simulation studies show the good possibility of applying heuristic search to such problems; in particular, this approach will be useful for the control of a nonlinear systems. However, there remain many problems to be solved.

First, the guiding principle is necessary how to choose an appropriate heuristic function. As is shown in examples, some heuristic function gives a satisfactory result, and the other not. A contour map will be helpful in judging the performance of a heuristic function.

Second, we need to choose a suitable sampling time when we introduce a discrete-time model. If a sampling time is long, then the level of a tree becomes low; therefore, it is easy to find a goal. In this case, however, the accuracy of a result necessarily gets worse. The accuracy depends upon a sampling time. We have to make a compromise between the easiness of search and the accuracy of a result.

Third, there exists the problem of directions in which a search process will be conducted. Here we use only forward search starting from an initial state. Backward search starting from a goal state may be effective in some cases; furthermore, the combination of these two will be possible.

Finally, the use of some knowledge will be considered. In examples, we do not use any knowledge about a control sequence. If we use some knowledge such as the pattern of a control sequence, we will be able to improve the efficiency of search.

Acknowledgment

The authors would like to thank Mr.T.Miyauchi for his help in simulation and preparing the manuscript.

References

[1] N.J.Nilsson, Problem-Solution Methods in Artificial Intelligence, McGraw-Hill, New York, 1971.

[2] E.Rich, Artificial Intelligence, McGraw-Hill, New York, 1983.

[3] T.Takahashi and H.Inooka, "Control of a pendulum under torque limitation: 2nd Report, Energy equation approach to swinging-up motion and experiments",Trans. JSME(C), vol.53(1987), pp.107-112.