URC97029

FEEDBACK IMPLEMENTATION OF ZERMELO'S OPTIMAL CONTROL BY SUGENO APPROXIMATION

C. Clifton, A. Homaifar, and M. Bikdash
The NASA-Autonomous Control Engineering Center
Department of Electrical Engineering
North Carolina A&T State University
Greensboro, NC 27411

Abstract

This paper proposes an approach to implement optimal control laws of nonlinear systems in real time. Our methodology does not require solving two-point boundary value problems online and may not require it off-line either. The optimal control law is learned using the original Sugeno controller (OSC) from a family of optimal trajectories. We compare the trajectories generated by the OSC and the trajectories yielded by the optimal feedback control law when applied to Zermelo's ship steering problem.

1. Introduction

Optimal control [Bryson, 1996; Kirk, 1970] is one of oldest approaches to control engineering. It has many advantages: (1) State and control constraints can be include explicitly. (2) The cost function to be minimized can be often given a simple intuitively appealing interpretation. (3) Optimal control is a very general methodology applicable to multi-input-multi-output, nonlinear, stochastic, or infinite-dimensional systems. Hence, optimal control theory provides a unified approach to stating and solving very general control problems that are at the same time physically intuitive. Unfortunately, optimal control theory suffers from a major disadvantage; namely, solving optimal control problems is in general compuationally difficult except in very special cases where a closed-form expression of the control law can be obtained. These cases include many nonlinear second-order systems and the celebrated linear quadratic regulator. In general however the necessary conditions have no closed-form solution and are at least as difficult to obtain as to solve a nonlinear two-instant boundary value problem (for the control of a system described by deterministic nonlinear ordinary differential equations. When the plant is stochastic or infinite-dimensional, the numerical difficulties are compounded.)

The absence of simple closed-form solutions and online numerical solutions of the general open-loop control problem means that there is no general feedback implementation (except in the neighborhood of an optimal reference trajectory using the well-known neighboring optimal control [Bryson and Ho; 1975].) The lack of feedback implementation is in our opinion the main reason why interest and research conducted in optimal control has greatly diminished.

On the other band, fuzzy-logic controllers (FLCs) are essentially feedback control laws. While theses controllers can be easily made to incorporate the heuristic knowledge of the control engineer, and this can be an advantage in cases where this is about the only knowledge available, designing a FLC using detailed, mathematical, and exact descriptions of the plant is not very well-understood or practiced.

Clearly, using all available knowledge about the system should in principle yield control laws with superior performance. Hence, we investigate in this paper the possibility of designing fuzzy logic controllers that approximate optimal control laws; from another point of view, we investigate feedback implementation of optimal control laws using fuzzy-logic controlled

To illustrate this approach, we consider the Zermelo's problem; that is, the problem of docking a ship going at constant water speed in minimum time in a region of strong water currents using the heading angle as the control i nput. We obtain a family of open-loop solutions of this problem and use it to train the OSC. The resulting trained engine will then be a feedback implementation of (a least-squares approximation of) Zermelo's optimal control. The Sugeno controllers [Buckley, 1993] are capable of approximating any continuous map within an arbitrary accuracy.

This paper is organized as follows. Section 2 provides the necessary background information on the optimal control of the ship steering problem. Section 3 discusses the training procedure used in designing the Sugeno-type controller from the data obtained from the optimal trajectories. Section 4 discusses the generation of training data and the elimination of angle discontinuity. Finally, section 5 summarizes the used procedure and shows simulation results.

2. Zermelo's Optimal Control Problem

The objective of Zermelo's problem is to find a minimum-time path through a region of position-dependent vector velocity [Bryson and Ho, 1975]. In this problem, a ship must travel in minimum time through a region of strong currents denoted by the two component vector $v(x)$

$$v_1 = v_1(x_1, x_2) \quad \text{and} \quad v_2 = v_2(x_1, x_2) \tag{1}$$

where (x_1, x_2) represent the position of the ship in rectangular coordinates and (v_1, v_2) are the velocity components in the same coordinate system, and the control u is the steering angle O, or $u = \theta$. The magnitude of the ship's velocity relative to the water, V, *is* a constant. The problem is to steer the ship in such a way as to minimize the time necessary to go from an arbitrary position x_0 to a specified docking position x_f.

The purpose of the generalized Sugeno controller is to approximate the steering angles needed to generate these minimum time paths as a function of x.

The equations of motion are as follows:

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} \equiv \begin{bmatrix} v_1 + V\cos\theta \\ v_2 + V\sin\theta \end{bmatrix} = v(x) + V \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \tag{2}$$

where $u = \theta$ is the heading angle of the ship's axis relative to the coordinate axes and is the control signal. The Hamiltonian of the system is:

$$H = \lambda_1(V\cos\theta + v_1) + \lambda_2(V\sin\theta + v_2) + 1 \tag{3}$$

and the Euler-Lagrange equations are $\dot{\lambda}_1 = -\frac{\partial H}{\partial x_1}$, $\dot{\lambda}_2 = -\frac{\partial H}{\partial x_2}$, and O $= \frac{\partial H}{\partial \theta}$ whose solution is $\tan\theta = \frac{\lambda_2}{\lambda_1}$. The optimal trajectories satisfy the boundary conditions $x(t_0)$ and $x(t_f)$ specified. Since the Hamiltonian is not an explicit function of time, H = constant is an integral of the system. Furthermore, since the objective is to minimize time, this constant is O. We have five equations to solve for the unknowns $x(t), \lambda(t)$ fort $\in [0, t_f]$, and for t_f.

Following [Bryson and Ho, 1975], we can simplify the two-point boundary problem by solving for $\dot{\theta}$ to obtain

$$\dot{\theta} = \sin^2\theta \frac{\partial v_2}{\partial x_1} + \sin\theta\cos\theta\left(\frac{\partial v_1}{\partial x_1} - \frac{\partial v_2}{\partial x_2}\right) - \cos^2\theta \frac{\partial v_1}{\partial x_2} \tag{4}$$

Equations (4) and (2) are the necessary conditions satisficd by this new and reduced-ordertwo-point boundary value problem. The four boundary conditions are: $x(t_0)$, and $x(t_f)$ are specified. They are used to solve for $\{x(t), \theta(t)\}$ from t_0 to t_f and for t_f itself. Note that if $v(x)$ were constant, then θ would be a constant. In other words, the minimum time paths are straight lines. If $v(x)$ vanes, it is possible for some of the optimal trajectories to intersect at conjugate points x_c. For these trajectories to be considcred optimal solutions, the control, u^* (t) = θ^* (t), must satisfy the following sufficient conditions

$$\frac{\partial^2 H\ (z''(t),\ \theta^*(t),\ \lambda^*(t), t)}{\partial \theta^2} = \frac{v}{V + v_1\cos\theta + v_2\sin\theta} \tag{5}$$

which is clearly is positive &finite if

$$V > v_1 \cos\ \theta + v_2 \sin\ \theta \ \text{ or if } \ V > \|v\| = \sqrt{v_1^2 + v_2^2} \tag{6}$$

For further discussion of Zermelo's problem, see [Bryson and Ho, 1975].

3. Approximation using the Sugeno Controller

A generalized Sugeno-type controller [Buckley, 1993] is a fuzzy engine mapping a vector $x = [x_1, x_2, \cdot\text{-}., x_n]^T \in \Re^n$ into u $\in \Re$ where x is interpreted as being a state vector or a measurement and u as a control action. The inference is of the form:

$$R^k\text{: IF } x_1 \text{ is } A_1^k \text{ and } x_2 \text{ is } A_2^k \text{ and...and } x_n \text{ is } A_n^k \text{ THEN } u = y^k = P_k(x) \tag{7}$$

where x_i is the i^{th} component of the input vector x and is a crisp value, A_i^k specifies which among the fuzzy attributes of x_i is tested by rule k, and P_k is a polynomial in $x_1, x_2, \ldots. x_n$ assigned to u by the k^{th} rule. The rulcs of the original Sugeno controller (OSC) have the following form

$$R^k : IF\ x_1 \text{ is } A_1^i \text{and } x_2 \text{ is } A_2^i \text{ and...} x_n \text{ is } A_n^i\ THEN\ u = y^k = c_0^k + c_1^k x_1 + + c_n^k x_n \tag{8}$$

where $c_0^k, c_1^k, \cdot\ldots, c_n^k$ are the consequence cocfficients of the k^{th} fuzzy rule. For further discussion of Sugeno-type controlled see [Buckley, 1993]. Buckley proved that a Sugeno-type controller can approximate any continuous real-valued function in the output space to any degree of accuracy if: (a) the input fuzzy sets have continuous membership functions and (b) a continuous T-norm is being used in the rule evaluation process. This is the universal approximating property of the Sugeno-type controller

Next, we consider approximating the trajectories of the optimal feedback control law by the original Sugeno controller. To do so, we need to determine the coefficients c_0^k, c_1^k, etc. In this paper, we use subscripts to index vectors and superscripts to identify components within a vector. In general, the output $\hat{u}$ for the inputs $x_1,...,x_n$ is obtained by the centroid method of defuzzification.

Let (x^j, u^j) be the j^{th} training input/output pair out of a total of J pairs. In this paper, these training data are obtained from the generated optimal trajectories. Then the consequence parameters can be obtained by solving a recursive least squares parameter identification problem [Takagi and Sugeno, 1985] where we determine the unknown coefficients by minimizing the error index

$$\min_{\mathbf{c}^1, \mathbf{c}^2, \cdots} \quad J = \sum_{j=1}^{J} (u^j - \hat{u}^j)^2 \tag{9}$$

where u^j is the output of the optimal feedback control law and $\hat{u}^j$ is the output of the Sugeno controller. The necessary conditions satisfied by the solution is $\mathbf{ZC}$ = U where

$$\mathbf{C} = \begin{bmatrix} c^1 \\ \vdots \\ \mathbf{c}^K \end{bmatrix}, \mathbf{c}^j = \begin{bmatrix} c_0^j \\ \vdots \\ c_n^j \end{bmatrix} \quad \mathbf{U} = \begin{bmatrix} u^1 \\ \vdots \\ u^J \end{bmatrix}, \quad Z = \begin{bmatrix} \beta^1 \otimes X^1 \\ \beta^2 \otimes X^2 \\ \cdots \\ \beta^J \otimes X^J \end{bmatrix} \tag{10}$$

where $\mathbf{Z}$ is a $J \times K(n+1)$ matrix, where $X^j = [1, x_1^j, \cdots, x_n^j]$ and

$$\beta_k^j = \beta_k(x^j), \text{ and } \beta^j = \beta(x^j) \tag{11}$$

represent the truth values of the rules evaluated at the vector x^j. The least squares solution for C can be calculated recursively by using the following procedure [Takagi and Sugeno, 1985 and Ljung and Soderstrom, 1986]. Denote the j^{th} row vector of matrix Z defined in (??) by z_j and the j^{th} row of U by u^j. Then, $\mathbf{C}$ can be then computed using the iteration.

$$\mathbf{C}^{(j+1)} = \mathbf{C}^{(j)} + \mathbf{S}^{(j+1)} \cdot z_{j+1}^T \cdot \left(u^{j+1} - z_{j+1} \cdot \mathbf{C}^{(j)}\right) \tag{12}$$

$$\mathbf{S}^{(j+1)} = \mathbf{S}^{(j)} - \frac{\mathbf{S}^{(j)} \cdot z_{j+1}^T \cdot z_{j+1} \cdot \mathbf{S}^{(j)}}{1 + z_{j+1} \cdot \mathbf{S}^{(j)} \cdot z_{j+1}^T} \tag{13}$$

where $\mathbf{S}^{(j)}$ is a square $(n(k+1) \times n(k+1))$ covariance matrix at the j^{th} iteration (i.e., after the j^{th} training pair has been acquired and used), and $\mathbf{C}^{(j)}$ the corresponding coefficient vector Then $\mathbf{C}^{(J)}$ at the final iteration is the least squares solution. The initial estimates, $\mathbf{C}^{(0)}$ and $\mathbf{S}^{(0)}$, are chosen as $\mathbf{C}^{(0)} = 0$ and $S^{(0)} = \alpha I$ where α is a large number and I is the identity matrix

Note that if rule 1 never fires (i.e. $\beta_1^j = 0$ for all j), Then Z is not full rank and $\mathbf{ZC} =$ U has no unique least-squares solution. Hence, if a rule never fires for the training data given, this rule should be eliminated to make the solution of the least squares problem unique. Also, this rule will not be applicable or relevant in all trajectories similar to the training data.

When the generalized Sugeno controller is used, the above procedure remains largely unchanged except that X now becomes for the case of n = 2

$$X = [1, x_1, x_2, x_1^2, x_1x_2, x_2^2, x_1^3, x_1^2x_2, x_1x_2^2, x_2^3, \cdots] \tag{14}$$

and the definition of $\mathbf{c}^k$ is to updated accordingly so that y^k defined in Eq. (7) can be expressed in the form $y^k = X\mathbf{c}^k$.

4. Methodology and Procedure

This section proposes a technique to approximate a feedback implementation of optimal controls. It uses the data generated from the optimal control law to identify the coefficients of the generalized Sugeno controller Here, we do not need to solve the two-point boundary-value problem for a arbitrary but given x_0; we only need to generate a family of optimal solutions of $x(t)$ in which an optimal trajectory reaches the final docking position x_f at some final steering angle Of, To generate one such trajectory, we use integrate Equations (4) and (2) backwards in time from $x(t_f) = x_f$ (the docking position) and $\theta(t_f) = \theta_f$ for any desired t_f until t = O. Optimal solutions are generated for two cases of $v_1(x)$ and $v_2(x)$.

We consider the simple case where the current velocity varies linearly. The objective is to find the minimum-time path from a certain point x_0 to a docking position at the origin. The velocity components of the currents are the following:

$$v_1(x) = -\frac{V}{h}x_2, \qquad v_2(x) = 0 \tag{15}$$

We generated 18 trajectories for t_f = 0.6 seconds. The trajectories along with their time and O contours are shown in Figures 1-2. The magnitude of the ship's velocity relative to water, V, is chosen to be 10 and h, a constant parameter, is chosen to equal 2. The optimal solutions for this case can be obtained in closed-form [Bryson and Ho, 1975], but our figures are generated by backward integration.

The map $\theta = \theta(x)$ modulo 2π has a discontinuity due to the modulo operation. For example, a training trajectory can start with an initial heading angle of 330°, and the angle increases gradually until it reaches 360° (at which θ becomes O") and end at the final heading angle of 10°. The discontinuity occurs at the transition point of 360°/00. Sugeno Engines encounter difficulties in approximating discontinuous maps.

To eliminate this problem, we use two Sugeno engines to approximate the sine and cosine of the heading angle as a function of the state and then to combine them after the approximation. Hence, we approximate $u_1(x) = \cos\theta$, and $u_2(x) = \sin\theta$ using two Sugeno engines and then we combine the two approximates using the formula $\theta = \arctan\frac{u_2}{u_1}$ for use in Eq. (2).

5. Simulation and Results

Now we summarize the procedure followed in this paper and show and discuss the results.

1. Generate the training data
   - Starting from the docking position x_f and a large t_f, integrate Eqs. (2) and (4) backwards in time from final conditions $x(t_f) = x_f$ and $\theta(t_f) = \theta_f$ till $t = O$. *This* will generate one extremum (actually optimum) trajectory for every choice of θ_f.
   - During an integration record the state $x(t)$ and the control $\theta(t)$ as the input and the output training data. The optimal time-to-go for that state is $t_f - t$.
   - Generate a set of trajectories by choosing a set of final values θ_f that is fine enough as to cover the regions of interest in the state space with enough optimum trajectories. Figures 1 and 2 show the generated optimum trajectories for Case 1 along with the associated time and control contours.
2. Perform the least-squares recursion to obtain the consequence coefficients C. There are two sets of coefficients, one for the. sine, one for the cosine.
3. Generate the testing data set. This is achieved by choosing a set of initial conditions z(O). We considered two testing sets:
   - One set was generated by taking the values of the state at the end of the backward integration conducted in step 1 above. We refer to this as Testing Set 1. If the approximation was perfect, the feedback control law will regenerate the optimum trajectories of step 1.
   - Another set was generated more or less randomly near the edge of the region of interest.
4. Simulate the feedback-controlled ship motion.
5. Consider the two performance measures .
   (a) The approximation error index J defined in Eq. (9).
   (b) How close the trajectories of the feedback-controlled ship matched the trajectories of the optimally-controlled ship.

We used five attributes for each input variable. Therefore, there are 25 possible rides, but six of these rides could be eliminated. The original Sugeno Controller yielded a very good approximation. The optimum trajectories and those generated from the approximate feedback law (using Testing Set 1) are shown in Figure 3. The average error index for the OSC is 7.2901.

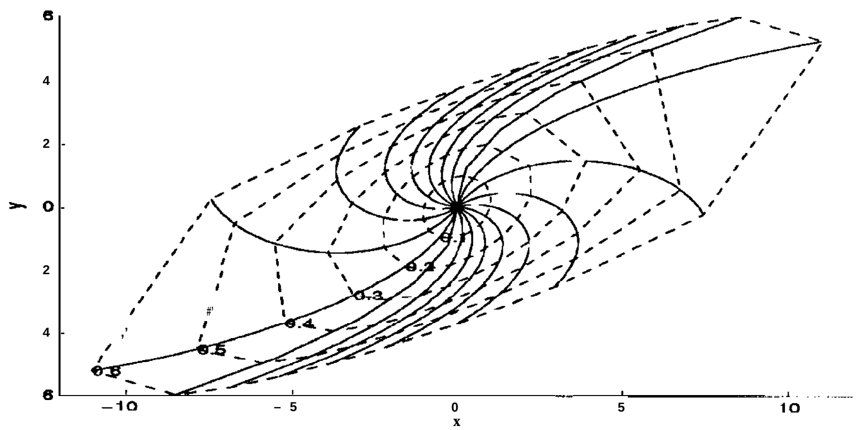

Figure 1: Optimal trajectories with equal-time contours.

6. Conclusions

Sugeno approximation and learning–from–example were shown to yield a powerful and easy-to-use method to implement optimal control in feedback. Since the lack of readily-available feedback implementation is a major limitation of optimal control, this new method is promising and encouraging.

In our approach, optimal control theory is used to generate a set of optimal state and control trajectories usually by backward integration, thus alleviating if not eliminating the need to solve two-point boundary value problems. Next, Sugeno Fuzzy Engines are taught to abstract and approximate the state-to-control mapping from these example trajectories. The original Sugeno engine was used to implement in feedback Zermelo's optimal control.

Acknowledgment

This work is funded by NASA-Autonomous Control Engineering Center under grant number ACE-48146. The authors wish to thank them for their financial support.

References

1. Bryson, A.E., "Optimal Control-1950 to 1985." *IEEE Control Systems*, Vol.16, No. 3, pp. 26-33, 1996.
2. Bryson, A.E. and Ho, Y.C., Applied optimal Control: *Optimization Estimation and Control,* Washington: Hemp-Shire, 1975.
3. Buckley, J, J., "Sugeno-type Controllers Are Universal Controllers. " *Fuzzy Sets & Systems*, Vol. 53, pp.299-303, 1993.
4. Kirk, D.E., *Optimal Control Theory*, Englewood Cliffs, NJ: Prentice Hall, 1970.
5. Lastman, G.J. and Sinha, N.K., *Microcomputer-Based Numerical Methods for Science and Engineering, New York* Saunders, 1989.
6. Ljung, L and Soderstrom, T., *Theory and Practice of Recursive Identification,* Cambridge, MA: MIT, 1986.
7. Sugeno, M. and Kang, G.T., "Fuzzy Modeling and Control of Multilayer Incinerator." *Journal of Fuzzy Sets & Systems,* Vol. 19, pp. 329-346, 1986.
8. Takagi, T. and Sugeno, M., "Fuzzy Identification of Systems and Its Applications to Modeling and Control." *IEEE Trans. on Systems, Man, and Cybernetics,* SMC-Vol. 15, No.. 1, pp. 116-132, 1985.

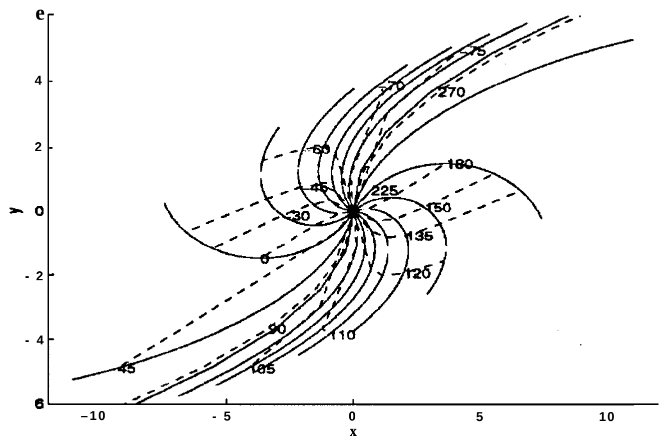

Figure 2: Optimal trajectories with equal-angle contours.

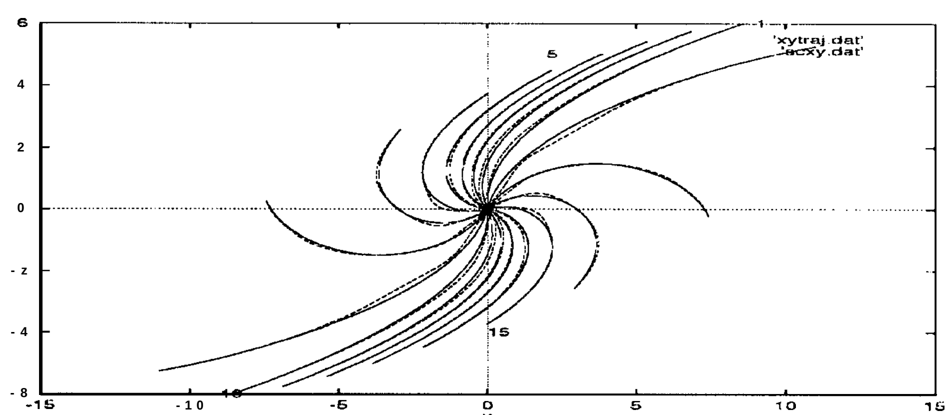

Figure 3: Trajectories generated by the optimal and approximate control laws.