Symbolic Computation for Integrator Backstepping Control Laws

Reza Ahangar and Rongdong Wang
Department of Mathematics
Texas A&M University-Kingsville, Kingsville, TX 78363, USA

Abstract

Chain integrator backstepping is a recursive design tool that has been used in nonlinear control systems. The complexity of the computation of the chain integrator backstepping control law makes inevitable the use of a computer algebra system. A recursive algorithm is designed to compute the integrator backstepping control process. A computer algebra program (Maple procedure) is developed for symbolic computation of the control function using a newly developed recursive algorithm. We will present some demonstrative examples to show the stability of the control systems using Lyapunov functions.

1 Introduction

Backstepping is a recursive design for systems with nonlinearities not constrained by linear bounds. With backstepping the construction of both feedback control laws and associated Lyapunov functions is systematic. This methodology is important in both theory and applications of nonlinear control systems.

Although the idea of integrator backstepping may be implicit in many earlier researches, it flourished during the past decades [1], [5], [9]. However, the complexity of the computation of the backstepping control law makes the use of a computer algebra system inevitable. The use of computer algebra systems not only helps the applications with numerical and graphical representation, but also helps the learning of the methodology.

Backstepping of chain of integrators has been studied since 1990 [2], [4], [5]. A lemma of chain of integrators was introduced [5]. Feedback control laws constructed with backstepping of a chain of integrators were shown

JCMCC 74 (2010), pp. 33-42

in [2] (page 86), [5] (page 36), and other publications in various forms. Detailed constructions for the case considered in this paper can be found in [6]. Computer algebra systems have been used in the study of backstepping [7]. We studied the chain of integrator backstepping and have derived the control law. We use symbolic computation of Maple to find the control law of the nonlinear system with chain of integrators.

This paper is structured as follows. In the first part we introduce the backstepping control laws of nonlinear systems with chain of integrators. The recursive algorithm is presented using Maple procedure to compute the control laws. In the following section several examples demonstrate the validity of the algorithm. Finally the conclusion and plan for further development of this algorithm is presented in the last section.

2 Integrator Backstepping

Consider the following integrator nonlinear output-feedback system

$$\begin{aligned} \dot{x} &= f(x) + g(x)\xi_1 \\ \dot{\xi}_1 &= \xi_2 \\ \dot{\xi}_2 &= \xi_3 \\ &\vdots \\ \dot{\xi}_{n-1} &= \xi_n \\ \dot{\xi}_n &= u. \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^k$, $\xi_i \in \mathbb{R}$ $(i = 1, ..., n)$, $u \in \mathbb{R}$ is the control input, f and g are locally Lipschitzian.

Lyapunov Stable Control Design: The Lyapunov-based control design is a methodology that generates an input function that satisfies Lyapunov stability theorems [3],[8]. Assume a dynamical system

$$\begin{aligned} \dot{x}(t) &= f(t, x(t), u(t)) \quad \text{for } t > t_0 \\ x(t) &= \phi(t) \end{aligned} \tag{2}$$

Where x is the state and u is the initial input. Let system (2) have an equilibrium point x_e in the solution domain. Choose $V(x)$ to be a scalar valued positive definite function of the state vector x (this idea is inspired from the formula of the total energy of the dynamical system), that is $V(x) = x^T x$. The time derivative can be described by the following

$$\dot{V}(x) = (\Delta_x V)\dot{x}(t).$$

System (2) is said to be stable if there is an input control function u so that the time derivative of the Lyapunov function is negative definite in a region of the equilibrium point x_e.

Throughout this paper, the following is assumed. For the system

$$\dot{x} = f(x) + g(x)u, f(0) = 0, \tag{3}$$

where $x \in \mathbb{R}^k$ is the state, $u \in \mathbb{R}$ is the control input, there exist a continuously differentiable feedback control law $u = \alpha_0(x)$ and a smooth, positive definite, radially unbounded function V: $\mathbb{R}^k \rightarrow \mathbb{R}$ such that $\frac{\partial V}{\partial x}(f(x) + g(x)\alpha_0(x))$ is negative definite.

Under the above assumption, system (3) is globally asymptotically stable with the initial control law $u = \alpha_0(x)$.

Chain Integrator Backstepping Lemma. Under the above assumption, for system (1) there is a Lyapunov function

$$\begin{aligned} V_a(x, \xi_1, ..., \xi_n) &= V(x) + \frac{1}{2}(\xi_1 - \alpha_0(x))^2 + \\ & \frac{1}{2}\sum_{i=2}^{n}(\xi_i - \alpha_{i-1}(x.\xi_1, ..., \xi_{i-1}))^2, \end{aligned}$$

which is smooth, positive definite, radially unbounded, and $\dot{V}_a$ is negative definite [5].

One choice for α_i $(i = 1, .., n)$ and the control law $u = \alpha_n$, which could be viewed as a special case of that presented in [6], is

$$\begin{aligned} \alpha_1(x, \xi_1) &= \alpha_0(x) - \xi_1 - \frac{\partial V}{\partial x}g(x) + \\ & \frac{\partial \alpha_0(x)}{\partial x}(f(x) + g(x)\xi_1), \end{aligned}$$

$$\begin{aligned} \alpha_2(x, \xi_1, \xi_2) &= \alpha_0(x) - \xi_1 + \alpha_1(x, \xi_1) - \xi_2 + \\ & \frac{\partial \alpha_1(x, \xi_1)}{\partial x}(f(x) + g(x)\xi_1) + \\ & \frac{\partial \alpha_1(x, \xi_1)}{\partial \xi_1}\xi_2, \end{aligned}$$

$$\begin{aligned}\alpha_i(x,\xi_1,...,\xi_i) \quad = \quad & \alpha_{i-2}(x,\xi_1,...,\xi_{i-2}) - \xi_{i-1} + \\ & \alpha_{i-1}(x,\xi_1,...,\xi_{i-1}) - \xi_i + \\ & \frac{\partial\alpha_{i-1}(x,\xi_1,...,\xi_{i-1})}{\partial x}(f(x)+g(x)\xi_1) \\ & +\sum_{j=1}^{i-1}\frac{\partial\alpha_{i-1}(x,\xi_1,...,\xi_{i-1})}{\partial\xi_j}\xi_{j+1}. \\ & (i=3,...,n)\end{aligned}$$

The final control law $u = \alpha_n(x,\xi_1,...,\xi_n)$ renders the zero solution of the chain integrator system, $x = 0$, $\xi_1 = \xi_2 = \cdots = \xi_n = 0$, globally asymptotically stable.

With the above choice for α_i $(i = 1,..,n)$, it can be verified that the function $V_a(x,\xi_1,...,\xi_n)$ is positive definite and its derivative is negative definite.

3 A Maple Procedure

The following is a Maple Procedure for computing the control law u of nonlinear system (1). For simplicity, we assume $k = 1$, that is, the equation $\dot{x} = f(x) + g(x)\xi_1$ is a scalar equation. In the case $k > 1$, the procedure can be modified easily.

The first argument of the procedure is $f(x)$. The second argument of it is $g(x)$. The third and the fourth arguments are the known Lyapunov function, $V(x)$, and the initial control law, $u(x)$, for system (3), $\dot{x} = f(x) + g(x)u$. The last argument is the number of integrators, n, of the system. The output of the procedure is the final control law of the system.

```
bst := proc (f, g, V, a, n)
  local i, u, xi, alpha;
  alpha[1] := a-xi[1]+
    (diff(a, x))*(f+g*xi[1])
    -(diff(V, x))*g;
  alpha[2] := a-xi[1]+alpha[1]-xi[2]
    +(diff(alpha[1], x))*(f+g*xi[1])
    +(diff(alpha[1], xi[1]))*xi[2];
  alpha[2] := simplify(alpha[2]);
  for i from 3 to n do
    alpha[i] := alpha[i-2]-xi[i-1]
```

```
      +alpha[i-1]-xi[i]
      +(diff(alpha[i-1], x))*(f+g*xi[1])
      +sum((diff(alpha[k], xi[k]))
      *xi[k+1], k = 1 ..  i-1);
    alpha[i] := simplify(alpha[i])
  end do;
  u := simplify(alpha[n])
end proc
```

4 Examples

To demonstrate the validity, we would like to test this computer algebra algorithm. The goal is to select and demonstrate examples whose analytical solutions are already known. We give two examples to show how the above procedure works.

Example 1. Our first example is borrowed from reference [5], page 35. Consider the system

$$\begin{aligned} \dot{x} &= x\xi_1 \\ \dot{\xi}_1 &= u. \end{aligned}$$

$V(x) = \frac{1}{2}x^2$ and $u(x) = -x^2$ are the Lyapunov function and a control law of the system

$$\dot{x} = xu$$

respectively.

The arguments of the Maple procedure are 0, x, $\frac{1}{2}x^2$, $-x^2$, and 1.

The following is the maple command to execute the program and its result. The result agrees with the control law shown in [5] (page 36).

$$u := bst\left(0, x, 1/2\,x^2, -x^2, 1\right)$$

$$u := -2\,x^2 - \xi_1 - 2\,x^2\xi_1$$

Stability of the Solution. We have written a Maple procedure, called `portrait`, to plot the phase portraits of initial problems with the variable initial conditions. The first argument of the procedure is the system; the second argument of the procedure is the number of initial value problems it will solve. We use that procedure to test the stability of the above systems. The following test shows the system with the computed control law globally stable.

We can run the Maple procedure for $n = 20$ randomly with generated initial conditions in Example 1, a system with one integrator. First, name the system as sys1, then call the procedure. The phase portrait demonstrates the stability of the dynamic control system.

$$sys1 := \left\{ \frac{d}{dt}x(t) = x(t)\,xi1(t), \frac{d}{dt}xi1(t) = -2\,(x(t))^2 - xi1(t) - 2\,(x(t))^2\,xi1(t) \right\}$$

portrait(*sys1*, *20*)

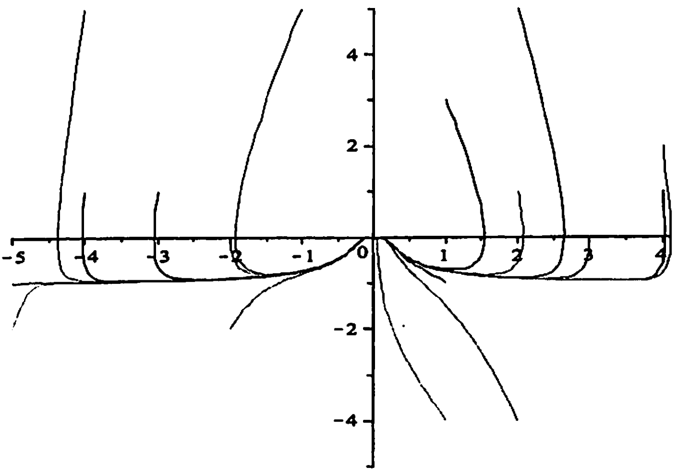

Fig 1; The phase portrait with random initial conditions

We call the procedure one more time:

portrait (*sys1*, 20)

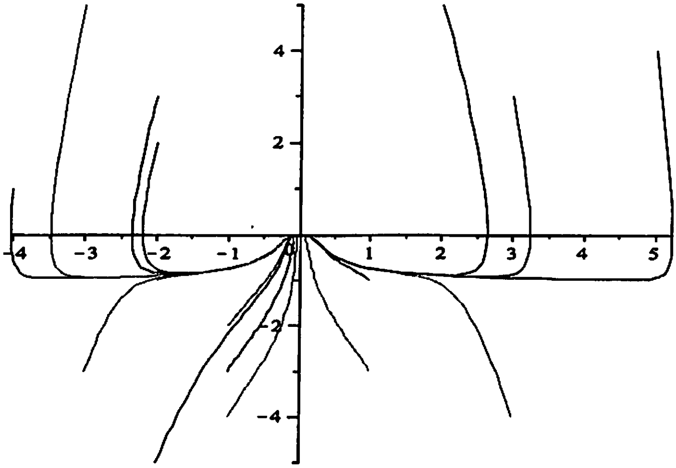

Fig 2; The phase portrait with random initial conditions

Now Consider the system with two integrators:

$$\begin{aligned}\dot{x} &= x\xi_1 \\ \dot{\xi}_1 &= \xi_2 \\ \dot{\xi}_2 &= u.\end{aligned}$$

The arguments of the Maple procedure are 0, x, $\frac{1}{2}x^2$, $-x^2$, and 2. The following are the commands for execution and the results of the computer algebra.

$u := bst\left(0, x, 1/2\,x^2, -x^2, 2\right)$

$$u := -3\,x^2 - 2\,\xi_1 - 6\,x^2\xi_1 - 2\,\xi_2 - 4\,x^2{\xi_1}^2 - 2\,\xi_2 x^2$$

Now consider the system with three integrators:

$$\begin{aligned}\dot{x} &= x\xi_1 \\ \dot{\xi}_1 &= \xi_2 \\ \dot{\xi}_2 &= \xi_3 \\ \dot{\xi}_3 &= u.\end{aligned}$$

The arguments of the Maple procedure are 0, x, $\frac{1}{2}x^2$, $-x^2$, and 3.

The following is the command and result.
$u := bst\left(0, x, 1/2\, x^2, -x^2, 3\right)$

$$u := -5\,x^2 - 3\,\xi_1 - 14\,x^2\xi_1 - 3\,\xi_2 - 16\,x^2{\xi_1}^2 - 2\,\xi_2 x^2 - \xi_3 - 8\,x^2{\xi_1}^3 - 4\,x^2\xi_1\xi_2$$

Example 2. Our next demonstrative example is selected from the reference [2], page 86. Consider the system

$$\dot{x} = x^2 + (1+x)\xi_1$$
$$\dot{\xi_1} = u.$$

where $V(x) = \frac{1}{2}x^2$ and $u(x) = -x$ are the Lyapunov function and a control law of the the following system respectively.

$$\dot{x} = x^2 + (1+x)u$$

The arguments of the Maple procedure are x^2, $1+x$, $\frac{1}{2}x^2$, $-x$, and 1. With the maple code the following nonlinear feed-back control result will be produced. The result of the algorithm produced by computer algebra agrees with the control law shown in [2] (page 86).
$u := bst\left(x^2, 1+x, 1/2\, x^2, -x, 1\right)$

$$u := -2\,x - 2\,\xi_1 - 2\,x^2 - \xi_1 x$$

Now consider the system with two integrators:

$$\dot{x} = x^2 + (1+x)\xi_1$$
$$\dot{\xi_1} = \xi_2$$
$$\dot{\xi_2} = u.$$

The arguments of the Maple procedure are x^2, $1+x$, $\frac{1}{2}x^2$, $-x$, and 2. The following is the command and result of the maple computation.
$u := bst\left(x^2, 1+x, 1/2\, x^2, -x, 2\right)$

$$u := -3x - 5\xi_1 - 4x^2 - 7\xi_1 x - 3\xi_2 - 4x^3 - 5\xi_1 x^2$$
$$-{\xi_1}^2 - {\xi_1}^2 x - \xi_2 x$$

5 Conclusions

The paper gives the feedback control laws of nonlinear systems of chain of integrators with backstepping and a Maple Procedure to compute the control laws. Several classical and well-known examples are used for demonstration purposes and to check the validity of the algorithm. These examples demonstrate that the symbolic computation approach is a powerful method in determining the nonlinear control solution in the backstepping method.

This algorithm and maple procedure can be developed to find the feedback control solution for some other nonlinear dynamical systems. The stability and behavior of the system can be studied. Developing the program to demonstrate the numerical and geometrical solution using computer algebra, like Mathematica or Matlab, will be interesting educational tools that can be used in industry or the engineering environment. Converting the algorithm into a Java program and using it on the Web will disseminate a world-wide application of backstepping method in nonlinear control systems.

Applying this algorithm to optimal control problems will help to discover more interesting links between backstepping procedure and dynamic programming.

References

[1] C. I. Byrnes and A. Isidori, "New Results and Examples in Nonlinear Feedback Stabilization." Systems and Control Letters, 12:437-442, 1989.

[2] I Kanellakopoulos, P. Kokotovic, and A. Morse, A toolkit for nonlinear feedback design, *Systems & Control Letters*, vol. 18, 1992, pp 83-92.

[3] H. K. Khalil, Nonlinear Systems, 2nd Ed. Upper Saddle River, New Jersey, 07458: Prentice Hall, 1996, vol. 1.

[4] P. Kokotovic and H. Sussnann, A positive real condition for global stabilization of nonlinear systems, *Systems & Control Letters*, vol. 13, 1989, pp 125-133,

[5] M. Krstic, I Kanellakopoulos, and P. Kokotovic, *Nonlinear and Adaptive Control Design*, John Wiley and Sons, 1995.

[6] Li, Z.-H. and M. Krstic, "Maximizing regions of attraction via backstepping and CLFs with singularities," Systems and Control Letters 30 (1997) 195-207

[7] F. Pozo and F. Ikhouane, "Computing adaptive backstepping control law using computer algebra", *Proceedings of the 2006 IEEE Conference on Computer Aided Control System Design*, Munich, Germany, 2006, pp. 403-407.

[8] J.J. E. Slotine and W. Li, "Applied Nonlinear Control", J. Wenzel, Ed. Upper Saddle River, New Jersey: Prentice Hall, 1991, vol. 1.

[9] J. Tsimias, "Observer design for nonlinear systems", Systems and Control Letters, 13:135-142, 1989.