**Proceedings of the 13th
KACC, October 1998**

Time-Delay System Toolbox and its Application [1]

시간 지연 시스템에 대한 툴박스와 그 응용

Wook Hyun Kwon*,°Arkadii Kim* [2], Soo Hee Han *,
Vladimir Pimenov**, Andrew Lozhnikov ***, Olga Onegova***

* Electrical Eng. Seoul National University (Tel : +82-2-880-7314; Email : avkim@cisl.snu.ac.kr)
** Mathematical Department, Ural State University, Russia
*** Institute of Mathematics and Mechanics, Ural Branch Acad. Sci. of Russia

Abstract: The report presents basic functions of Time-delay System Toolbox (for MATLAB) - the general-purpose software package for Computer Aided Design of control systems with delays. The Toolbox is a collection of algorithms, expressed mostly in m-files for simulating and analysis of MIMO *linear* and *nonlinear* systems with *discrete* and *distributed* (time-varying) delays.

Keywords: Time-delay, numerical methods, control, software

1. Introduction

At present there are elaborated effective numerical methods and corresponding software for solving different classes of ordinary differential equations (ODE) and partial differential equations. The progress in this direction results in wide application of these types of equations in practice. Another class of differential equations is represented by delay differential equations (DDE), also called systems with delays, hereditary systems, functional differential equations. It is easy to understand that the delay occurs practically in many mathematical models. Though at present different *theoretical* aspects of DDE theory are developed with almost the same completeness as the corresponding parts of ODE theory, there were no *effective* numerical methods of solving *general classes* DDE (for example, for general DDE the classical Runge-Kutta method was suggested only in continuous form [4] which is difficult for software realization), as consequence, there are no general-purpose software packages for simulating systems with delays. This is, apparently, one of the main reasons why DDE are not so widely used in practice unlike ODE, though such type equations describe many phenomena more accurately than ODE. In the papers [1, 2, 3] there was elaborated new approach to constructing simple and effective numerical methods for systems with delays. These methods are the basis of numerical techniques of the suggested in this report Toolbox for simulating time-delay systems. The distinguishing feature of the Toolbox consists not only in new effective numerical algorithms, but also in new control design methods [2] such as linear quadratic regulator (LQR) for system with delays.

Time-delay System Toolbox provides support for:

- numerical *simulation* of general linear and nonlinear systems with delays with discrete and distributed delays,
- *time-domain analysis* of linear systems with delays,
- *stability* analysis including pole location algorithm, coefficient stability conditions and special numerical test method for stability,
- new *analytical* design techniques for linear quadratic regulator problem.

Remark. The numerical methods and control design algorithms, realized in Time-Delay System Toolbox, are a *direct* generalization of the corresponding methods of ODE case, i.e. if delays disappear then all algorithms coincide with the corresponding numerical and control algorithms for ODE.

2. What algorithms are used

2.1. General remarks

In the Toolbox we realized general algorithms which can be applied for modelling and analysis of:

- MIMO time-delay systems,
- systems with *distributed* and *time-varying* delays,

so we use only **state space** representation of systems with delays and the corresponding methods, because frequency domain approach cannot be applied for these problems.

2.2. Numerical algorithms

Simulation of systems with delays is realized on the basis of new numerical Runge-Kutta-like methods [1, 2, 3].

[1] This work was supported in part by Korean Federation of Science and Technology Societies and in part by Russian Foundation for Fundamental Investigations (Grant N 98-01-00363).

[2] On leave from the Institute of Mathematics & Mechanics Ural Branch Acad. Sci. of Russia.

The distinguishing features of numerical algorithms are the following:

1 the numerical methods for DDE are direct analogies of the corresponding classical numerical methods of ODE theory, i.e. if delays disappear, then the methods coincide with ODE methods (note, in the frameworks of other approaches to constructing numerical methods for general classes DDE there is no such kind of succession of algorithms);

2 in contrast to other methods the proposed numerical algorithms don't depend on specific form of systems with delays, so the corresponding numerical schemes are the same for different classes of DDE (this allows to elaborate sufficiently simple structure of software algorithms).

Remark. The numerical algorithms work for *piecewise continuous* initial functions.

2.3. Time-domain analysis

On the basis of the numerical algorithms the m-files for *time-domain analysis* (time-response) of general linear systems with delays are elaborated.

2.4. Stability

One of the most important characteristic of (linear) systems is the *stability* of the corresponding solutions. The Toolbox contains some algorithms of verification stability of linear time-invariant systems with delays.

For systems with *distributed* and *time-varying* delays there are no effective *algorithmic* methods of verification the stability property. So we propose some procedures of testing stability property for *general* linear systems with delays using numerical simulation.

2.5. Control algorithms

LQR technique is one of the most useful methods for designing closed-loop controller for ODE. Note, in the Control System Toolbox for MATLAB the effective procedure of designing linear quadratic regulator for linear finite-dimensional systems (on the basis of solving ARE equation) is realized.

However, from the early sixties, when LQR problem for systems with delays was stated, up to now there are no approximate methods of its solving that could be applicable for software realization [6] .

In the paper [2], new approach to constructing *exact* solutions of generalized LQR problems for systems with delays is developed. These *analytical methods* of designing liner quadratic regulator for systems with delays (which can be considered as direct generalization of the corresponding methods of solving LQR problem for ODE) are realized in the Toolbox.

[6]Though there were proposed a few approximate methods of solving LQR problem (see, for example, [5, 6], however these approaches are complicated for practical implementation and creating software packages.

3. Functions of the Toolbox

1. **Numerical algorithms**
   Explicit Runge-Kutta-like methods (**fde45**)
   Lottka-Volterra delay system (**lv45**)
   Linear time-delay system (**rk45lin, rk45lie**)

2. **Time-domain analysis**
   Initial condition response (**initiald**)
   Arbitrary input response (**inputd**)
   Step input response (**stepd**)
   Impulse response (**impulsed**)

3. **Stability**
   Test for stability (**test, teste**)
   Pole location (**pole**)
   Coefficient stability conditions (**coef1, coef2**)

4. **Control algorithms**
   Generalized Riccati equations (**gre**)
   Optimal gain (**lqdelay**)
   Generalized LQ regulator (**clso**)
   Arbitrary linear feedback (**clsim**)
   Positiveness of quadratic matrices (**isdef**)
   Positiveness of quadratic functionals (**costfun**)

4. Representation of time-delay systems in the Toolbox

4.1. Description of DDE by finite number of functions and integrals

Systems with delays

$$\dot{x}(t) = f(t, x(t), x(t+s)), -\tau \le s < 0, \qquad (1)$$

are *infinite dimensional* systems because of the presence of the functional component $x(t+s)$, $-\tau \le s < 0$, which characterizes delays. So the right part of system (1) is a mapping $f(t,x,y(\cdot)) : [t_0, t_0+\theta] \times \mathbf{R}^n \times Q[-\tau, 0) \to \mathbf{R}^n$ and the most problem of simulating of such systems consists in describing of $f(t,x,y(\cdot))$ by *finite number* of parameters [7] .

Analyzing the structure of systems with delays we can see that in concrete cases right parts of such systems are combinations of *finite dimensional* functions and *integrals* and can be presented in following forms (or their combinations):

$$f(t,x,y(\cdot)) = g(t,x) + \int_{-\tau}^{0} \beta(s, y(s))ds \qquad (2)$$

$g(\cdot,\cdot) : \mathbf{R} \times \mathbf{R}^n \to \mathbf{R}^n$, $\beta(\cdot,\cdot) : [-\tau, 0] \times \mathbf{R}^n \to \mathbf{R}$,

[7]Because for computer simulation usually only *finite* algorithms with *finite number* of *input parameters* are used.

$$f(t,x,y(\cdot)) = \int_{-\tau}^{0} \eta(t,s,x,y(s))ds \quad (3)$$

$$\eta(\cdot,\cdot,\cdot,\cdot) : \mathbf{R} \times [-\tau,0] \times \mathbf{R}^n \times \mathbf{R}^n \to \mathbf{R}^n,$$

$$f(t,x,y(\cdot)) = \int_{-\tau^*(t)}^{0} \mu(s,y(s))ds \quad (4)$$

$$\tau^*(\cdot) : \mathbf{R} \to (0,\tau], \mu(\cdot,\cdot) : [-\tau,0] \times \mathbf{R}^n \to \mathbf{R}^n,$$

$$f(t,x,y(\cdot)) = g(t,x) + P[y(-\tau)] \quad (5)$$

$$P[\cdot] : \mathbf{R}^n \to \mathbf{R}^n,$$

$$f(t,y(\cdot)) = P[y(-\tau^*(t))] \quad (6)$$

$$\tau^*(\cdot) : \mathbf{R} \to (0,\tau], P[\cdot] : \mathbf{R}^n \to \mathbf{R}^n.$$

So every of mappings (2) – (6) can be described just by finite number of functions. For example, to describe mapping (4) we should set two functions $\tau^*(\cdot)$ and $\mu(\cdot,\cdot)$. All these functions can be described as m-files .

4.2. Conditional representation of DDE

For *structural* presentation of time-delay systems it is convenient to use the following *conditional representation*

$$\dot{x} = f(t,x,y(\cdot)) \quad (7)$$

($h = \{x, y(\cdot)\} \in H$), i.e. we just write in the right-hand side of the equations the mapping $f(t,x,y(\cdot))$ (without any indication of solutions).

Remark. Let us remember that for ODE

$$\dot{x}(t) = g(t,x(t)) \qquad (g(t,x) : \mathbf{R} \times \mathbf{R}^n \to \mathbf{R}^n)$$

the conditional representation is

$$\dot{x} = g(t,x), \quad (8)$$

i.e. the argument t is not pointed out in state variable $x(t)$, and we just write the mapping $g(t,x)$ (without any indication of solutions) in the right-hand side of the equations.

4.3. Initial condition for DDE

We consider the initial conditions of systems with delays as the pair $h = \{x^0, y^0(\cdot)\} \in H$ of a vector $x^0 \in \mathbf{R}^n$ and a function $y^0(\cdot) \in Q[-\tau, 0)$, i.e.

$$x(t_0) = x_0, \quad (9)$$

$$x_{t_0}(s) = y^0(s), \ -\tau \le s < 0. \quad (10)$$

So in order to simulate systems with delays it is necessary to set into the m-files initial data as $\{x^0, y^0(\cdot)\}$.

Remark. The initial function $y^0(\cdot)$ can be described using m-file.

4.4. Linear control systems with delays

Let us consider linear system with delay

$$\dot{x}(t) = A_0\, x(t) + A_\tau\, x(t-\tau) + \int_{-\tau}^{0} G(s)\, x(t+s)\, ds + B\, u. \quad (11)$$

where A, A_τ, B are constant $n \times n$, $n \times n$, $n \times r$ matrices, $x \in \mathbf{R}^n$, $u \in \mathbf{R}^r$.

In order to simulate this system using Toolbox it is necessary to define the following *finite number* of parameters:

- matrices A, A_τ, B,
- the matrix-function $G(s)$,
- open-loop control $u(t)$ (closed-loop control $u(t,x,y(\cdot))$),
- an initial time-moment t_0,
- an initial point x^0,
- an initial prehistory $y^0(\cdot) = \{y^0(s), -\tau \le s < 0\}$.

Remark. The matrix-function $G(s)$ can be described using m-file.

5. Time-domain analysis (example)

The present version of Time-Delay System Toolbox provides four functions for time-domain analysis of linear system with delays. The figure1 and figure2 show the simulation results by **initiald** and **impulsed** for system

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} -0.3 & -0.1 \\ -0.2 & -0.4 \end{bmatrix} x(t-5) \\ &+ \begin{bmatrix} 0 \\ 0.333 \end{bmatrix} u(t) \\ y(t) &= \begin{bmatrix} 0.9 & 0.1 \end{bmatrix} x(t). \end{aligned} \quad (12)$$

System (12) is unstable (because this system has two roots with positive real parts), so trajectories are diverging.

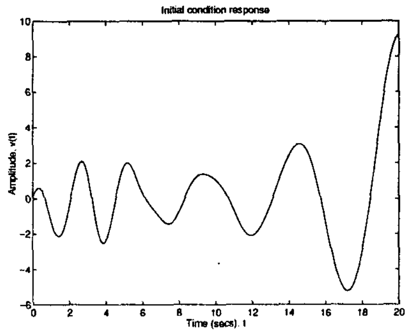

Figure 1: Response due to initial value

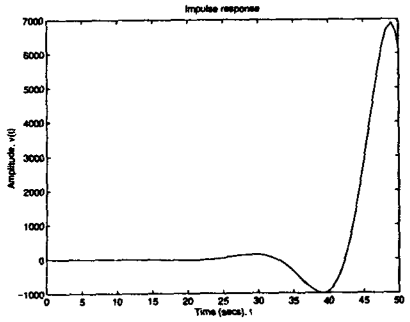

Figure 2: Response due to impulse input

6. LQR design technique (with an example)

LQR method (basing on solution of algebraic Riccati equation) is the universal tool for designing stabilizable controller for ODE. In case of DDE the most difficulties of realization of the corresponding LQR technique is to find solutions of the specific system of generalized Riccati equations (GRE), consisting of algebraic equations, ordinary differential equations and partial differential equations.

In the Toolbox we realized procedure of designing optimal LQR controller based on explicit form of GRE solutions obtained in [2].

The design procedure of optimal control consists of three steps.

1 Commands **gre** and **lqdelay** calculate the matrices C, D_0, D_1, D_2, related with the optimal control

$$u^0(x, y(\cdot)) = C\,x + D_0 \int_{-\tau}^{0} e^{D_1 s} D_2\, y(s)\, ds\,.$$

2 Function **clso** forms the corresponding closed-loop system and allows to simulate it.

3 Command **test** allows to simulate the optimal closed-loop system with respect to the specific set of initial base functions (polynomials, sin, cosine, etc). Using this function it is possible to check stability of the closed-loop system with respect to initial function space spanned on corresponding base functions.

Besides that, Toolbox offers some additional functions. For example, **costfun** calculates the approximate value of the cost functional of LQR problem. Using **isdef** one can check the definetness of a matrix which is the finite dimensional approximation of the optimal value quadratic functional.

Applying described LQR algorithms to system (12) we find the form of optimal control

$$u^0(x, y(\cdot)) = \left[\begin{array}{cc} -1 & -2.6469 \end{array}\right] x + \int_{-5}^{0} \{ \left[\begin{array}{cc} 0 & -0.3330 \end{array}\right]$$

$$exp(\left[\begin{array}{cc} 0 & -0.3330 \\ 1 & -0.8814 \end{array}\right] \times s) \left[\begin{array}{cc} 0.1053 & 0.1921 \\ -0.0052 & 0.1297 \end{array}\right] y(s)\} ds$$

Figure 3 shows that the designed control stabilizes the system (12).

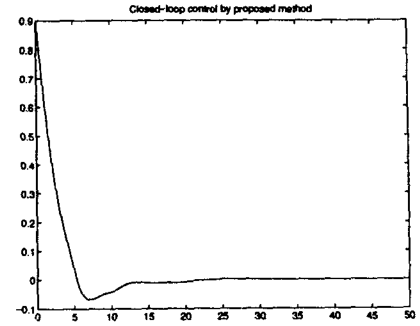

Figure 3: Response due to optimal control

7. Conclusion

Time-Delay System Toolbox is, to authors' knowledge, the first general purpose software for simulating and analysis of systems with delays. We would appreciate comments and suggestions for the next version of the Toolbox.

Reference

[1] A.V. Kim and V.G. Pimenov, "Numerical methods for time-delay systems on the basis of i-smooth analysis" *Proceedings of the 15th World Congress on Scientific Computation, Modelling and Applied Mathematics*, V. 1: Computational Mathematics, pp. 193–196, 1997'

[2] A.V. Kim, S.H. Han, W.H. Kwon and V.G. Pimenov, "Explicit numerical methods and LQR control algorithms for time-delay systems", *Proceedings of the International Conference on Electrical Engineering*, Kyungju, Korea, July 21-25, 1998.

[3] A.V. Kim and V.G, "On application of i–smooth analysis to elaboration of numerical methods for functional differential equations", *Proceedings of the Institute of Mathematics and Mechanics*, Vol. 5, pp. 104–126, 1998.

[4] L. Tavernini, "One-step methods for the numerical solution of Volterra functional differential equations", *SIAM J. Numer. Anal.*, Vol. 8, pp. 786–795, 1971.

[5] D.W. Ross, "Controller design for time lag systems via quadratic criterion", *IEEE Trans. Automat. Control*, Vol. 16, pp.664–672, 1971.

[6] K. Uchida, E. Shimemura, T. Kubo and N. Abe, "The linear-quadratic optimal control approach to feedback control design for systems with delay", *Automatica*, Vol. 24, No. 6, pp. 773–780, 1988.