IDENTIFICATION OF DISCRETE TIME SYSTEMS WITH USING OF MATLAB

Stanislav Kocúr

Department of Control and Information Systems,
Faculty of Electrical Engineering, University of Žilina

Abstract

Identification of unknown continuous or discrete time systems is very wide and important part of system control theory. Qualitative indicators of system control directly depends on system identification because system control is only as good as good is created model of system. So this article deals with system identification problem. At first identification problem is described, then there is simple classification of identification methods. Rest of paper is devoted to required theory and one practical example of discrete time FIR system parameters identification with using of recursive least mean squares method in Matlab.

1 INTRODUCTION

Continuous-time or discrete-time system control assumes accomplishment of two preconditions. First of all is needed to know controlled system so it is needful to make his mathematical model. The next step is selection of adequate control algorithm, or setting of control law parameters of fixed structure, which is predetermined. The synthesis depends on attributes and structure of mathematical model both of these cases.

There are two ways how to make mathematical model. The first approach requires using of analytic methods. This means model making based on physical, chemical or other characteristic processes. These processes are running in system. This kind of models has some disadvantages. These models cannot include all real factors of system function and they can give non-linear functionalities. The second method is application of experimental identification methods. These methods can process data measured on concrete system [1].

2 IDENTIFICATION PROBLEM

System identification is possible to understand like sequence for making abstract system from his real paragon. This abstract system gives a true picture of real system cardinal attributes, what are needful for working and optimal system control. So this abstract system is model of real system. In system control domain are used mathematical models of systems what can have form of differential equation for continuous-time systems or difference equation for discrete-time systems. System models is possible to sort into many groups like static and dynamic, continuous-time and discrete-time, deterministic and stochastic, linear and non-linear, etc. In this article is thinking dynamic discrete-time model of unknown discrete-time system.

Principle of unknown system identification is visible on the figure 1. Unknown system is impacted by input signal *u(n)*. This signal is generated by control system (regulator). Reaction of system for input signals is system response *y(n)*. Usually is impossible or unsuitable to interfere in real system if we want to prevent damage or correct function change. Then remains only one technique, how to acquire some information about system. We must make model (abstract system), which we will periodically compare with real system. Input signal *u(n)* is collective for both of systems. Adjustable system task is self-acting change of his operation. This change must be special-purpose. It is desirable convergence of adjustable system function to unknown system function. Variable, which represents correspondence dimension between system and his model, is identification error:

$$e_i(n) = y(n) - y'(n) . \tag{1}$$

If identification error is minimal or zero, model of unknown system is made by parameters of adjustable system *h(n)*.

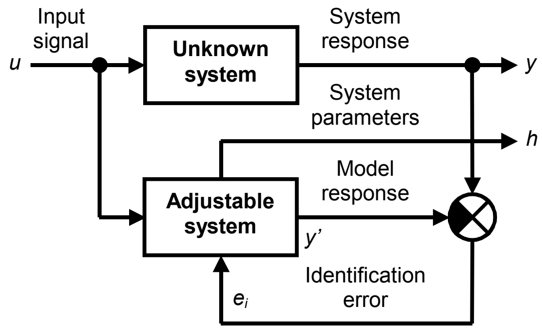

Figure 1: Direct system identification

Control system needs to know attributes of controlled system for calculation of control signal. Control signal is also input signal *u(n)* of controlled system. If unknown system function is unchangeable in time, identification process is easier and what is most important, it is not necessary to repeat it. This sort of identification is called static identification and this identification method is called one-off processing method. But in area of automatic regulation is system function time-inconstant very often. There is previous method insufficient. Identification must be repeated. In extreme cases is inevitable to repeat this process in every sample period. This kind of model making is called dynamic modelling. Methods used for this model making are named running identification methods [2].

3 IDENTIFICATION METHODS EVOLUTION AND SORTING

Evolution of identification methods is visible on the figure 2. Identification of continuous-time dynamic systems was concentrated to two goals to the first half of sixties. The first goal was evolution of methods, what could evaluate transfer characteristics, or some other responses for normalized signals. The second goal was evolution of system identification, when input signal of system was stochastic. This identification was based on Wiener-Hopf equation solving [1].

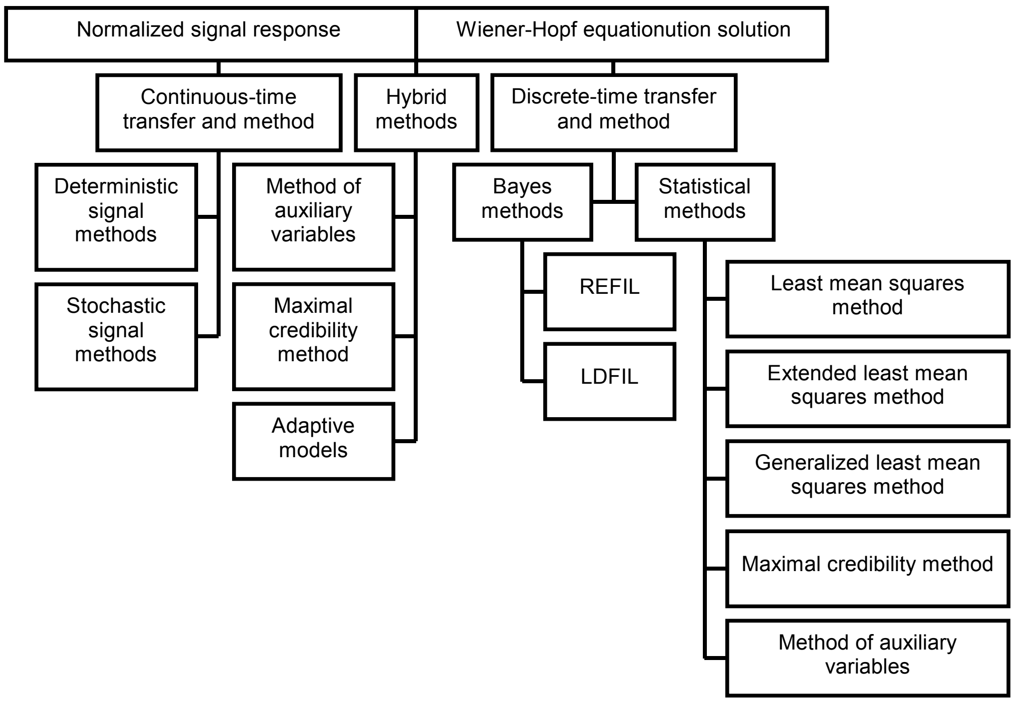

Figure 2: Identification methods evolution and sorting

Next evolution of identification methods was connected with computer technology progress. The base of digital methods is one-off processing method for determining of identification parameter's vector *h*. Least mean squares method or maximal credibility method is usable for it. Then running identification methods are extrapolated from one-off processing methods. Centre of a running identification problem is based on iteration of identification parameter's vector's calculation. That means repeating of calculation with every another sampling rate period, when some new input and output data was acquired from the system (new identification parameter's vector's calculation).

4 RECURSIVE LEAST SQUARES METHOD

Recursive methods based on least mean squares method calculate parameter estimation repeatedly in every sample rate period. If we know parameter estimation $\Theta'(n)$ which was calculated from information acquired in time n, we can compute parameter $\Theta'(n+1)$ by using some simple modification of previous estimation $\Theta'(n)$ [3].

There are three typical features of recursive methods:

a) they are important part of adaptive systems, where system control is based on topical model,
b) they have very undemanding control system memory requirements,
c) they are simply modifiable for real-time data processing and for changed parameters.

Deduction of recursive methods is based on matrix inversion theorem. Let is given equation:

$$M = A + BC^{-1}D\,, \tag{2}$$

where A, B, C, D, M are matrixes. If A and C are regularly matrixes, then following equation is valid:

$$M^{-1} = A^{-1} - A^{-1}B\left(DA^{-1}B + C\right)^{-1}DA^{-1}\,. \tag{3}$$

Let is system described by his difference equation [4]:

$$y(n) + a_1 y(n-1) + \ldots + a_{na} y(n-na) = b_1 u(n-1) + \ldots + b_{nb} u(n-nb) + e(n)\,. \tag{4}$$

If we set up terminology:

$$\begin{aligned} \Theta^T &= \left(a_1, \ldots, a_{na}, b_1, \ldots, b_{bn}\right), \\ z^T(n) &= \left(-y(n-1), \ldots, -y(n-na), u(n-1), \ldots, u(n-nb)\right), \end{aligned} \tag{5}$$

then we can write:

$$y(n) = \Theta^T z(n) + e(n)\,. \tag{6}$$

Let is available complex of measured values in matrix form:

$$\begin{gathered} Z = \begin{pmatrix} -y(0) & \ldots & -y(1-na) & u(0) & \ldots & u(1-nb) \\ -y(1) & \ldots & -y(2-na) & u(1) & \ldots & u(2-nb) \\ \vdots & & \vdots & \vdots & & \vdots \\ -y(K-1) & \ldots & -y(K-na) & u(K-1) & \ldots & u(K-nb) \end{pmatrix}, \\ Y = \begin{pmatrix} y(1) \\ y(2) \\ \vdots \\ y(K) \end{pmatrix},\ e = \begin{pmatrix} e(1) \\ e(2) \\ \vdots \\ e(K) \end{pmatrix}, \end{gathered} \tag{7}$$

where $K > na + nb$. Then is possible to rewrite equation (6):

$$Y = \Theta Z + e\,, \tag{8}$$

and we can determine parameter estimation Θ':

$$\Theta' = \left(Z^T Z\right)^{-1} Z^T Y\,. \tag{9}$$

For RLS method deduction is required knowledge of three elements. The first is parameter estimation *Θ'(n)*. The second parameter is covariance matrix *P(n)*, which is defined like:

$$P(n) = \left(Z^T(n)Z(n)\right)^{-1} . \tag{10}$$

The third parameter is data file acquired by gauging:

$$Y(n+1) = \begin{pmatrix} Y(n) \\ y(n+1) \end{pmatrix},\ Z(n+1) = \begin{pmatrix} Z(n) \\ z^T(n+1) \end{pmatrix},\ Z^T(n+1) = \begin{pmatrix} Z^T(n) & z(n+1) \end{pmatrix}. \tag{11}$$

After extrapolation based on equation (9) with using of information (11) is possible following formal notation of recursive least mean squares method algorithm [3]:

$$\begin{gathered} \gamma\,(n+1) = \left(1 + z^T(n+1)P(n)z(n+1)\right)^{-1}, \\ L(n+1) = \gamma\,(n+1)P(n)z(n+1), \\ P(n+1) = P(n) - \gamma\,(n+1)P(n)z(n+1)z^T(n+1)P(n), \\ \Theta'(n+1) = \Theta'(n) + L(n+1)\left(y(n+1) - z^T(n+1)\Theta'(n)\right). \end{gathered} \tag{12}$$

It is possible to write matrix equation for correction of an old matrix *Θ'(n)* with recursive mode:

$$\Theta'(n+1) = \Theta'(n) + \frac{\Theta'(n)u(n+1)u^T(n+1)\Theta'(n)}{1 + u^T(n+1)\Theta'(n)u(n+1)} . \tag{13}$$

Every recursive algorithm need initial conditions. Usually we can vote *Θ'(0)=cE* where *c* is constant suitably voted from interval $0{,}01 \le c \le 1$, and *E* is identity matrix.

5 RLS METHOD APPLICATION

Typical application of RLS algorithms is their using in adaptive models [2]. Adaptive model of unknown discrete-time system is usually based on adaptive filter with finite impulse response. Digital filter is easy for using and parameter changing. His coefficients are changed in every sample rate period for maximal model output signal convergence to unknown system output with using of following equation:

$$h(n+1) = h(n) + \Theta'(n+1)u(n+1)\left(y(n+1) - y'(n+1)\right), \tag{14}$$

where *h(n)* is an old and *h(n+1)* is a new vector of filter parameters, and the rest of variables correspond with equation (13) and figure 1.

In following last chapter is offered simply function for model adaptation based on derived theory. But it is needful to warn of this function program integration. Degree of filter is good to vote *FD=10* or *11*. This number was selected after some off-line simulations like the most suitable. Function was primary wrote for simulation. His using for on-line identification adjures removing of initial conditions. Filter parameters vector *h* must be set up to zero only one time at the start of modeling process and out of function body, somewhere in main program by need. The same is valid for *theta* matrix initialization. Also main cycle instruction (FOR) must be removed from function and last condition (IF) is not needful. Body of cycle will stay of course and condition will be reduced to her result only. These modifications are entrusted for user requirements.

6 FUNCTION FOR MATLAB

```
%Function for RLS modelling
function [h,esti,er]=lmsr(u,y,FD,c)                     % function definition, inputs and outputs
N=length(u);                                            % input signal vector length finding
theta=inv(0.01*eye(FD));                                % matrix initial value
for m=1:1:FD                                            % cycle for filter parameter vector clearing
    h(m,1:FD)=0;                                        % initial filter coefficients = 0
end
for n=FD:1:N                                            % cycle for adaptive model making
    U=u(n:-1:n-FD+1)';
    esti(n)=u(n:-1:n-FD+1)*h(n-1,1:FD)';                % estimated output signal computation
    er(n)= y(n)-esti(n);                                % identification error calculation
    theta=theta+((theta*U*U'*theta)/(1+Y'*theta*Y));    % recursive correction of matrix
    if n<N                                              % if this sample is not last,
      h(n,1:FD)=(h(n-1,1:FD)'+theta*U*er(n))';          % then new coefficients are calculated
    end
end
```

7 CONCLUSION

Matlab includes more modifications of least squares method for identification processes. There is function RLS for construction of a recursive least squares adaptive algorithm object too. But when :

a) user needs to find optimal filter degree for his/her model,
b) user needs on-line object identification, when data come from PC port or additional meter reading PC card,
c) user needs to implement identification process into control algorithm with continuous monitoring of all regulation and identification circuit quantities,

functions included in Matlab are difficult for use in these kinds of programs. There is not possibility to control all signals and mathematical operations accomplished with them in every sample rate period in function body. And transparency is very important function character during program tuning. This was reason for own identification function making.

This article was created with support of culture and education agency under the terms of project KEGA number K-057-06-00 "Innovation of laboratory teaching methods based on Matlab modelling and simulation in combination with teaching models via e-learning."

REFERENCES

[1] Chmúrny, D., Prokop, R., Bakošová, M.: Automatizované riadenie chemickotechnologických procesov. ALFA, Bratislava 1988.

[2] Jan, J.: Číslicová filtrace, analýza a restaurace signálu. VUTIUM, Brno 2002. ISBN 80-214-1558-4

[3] Fikar, M., Mikleš, J.: Identifikácia systémov. STU Bratislava 1998. ISBN 80-227-1177-2

[4] Balátě, J.: Automatické řízení. BEN, Praha 2003. ISBN 80-7300-020-2