

Nonlinear Continuous System Identification by Means of Multiple Integration II

J. John

This paper presents a new modification of the multiple integration method [1, 2, 3] for continuous nonlinear SISO system identification from measured input – output data. The model structure is changed compared with [1]. This change enables more sophisticated systems to be identified. The resulting MATLAB program is available in [4]. As was stated in [1], there is no need to reach a steady state of the identified system. The algorithm also automatically filters the measured data with respect to low frequency drifts and offsets, and offers the user a potent tool for selecting the frequency range of validity of the obtained model.

Keywords: continuous system identification, multiple integration.

1 Basic definitions

Let us take a nonlinear continuous time-invariant SISO (Single Input – Single Output) system, described by the state equations (see also Fig.1)

$$\dot{x}_i(t) = x_{i+1}(t) + f_i[\mathbf{u}(t), y(t)]; \quad i = 1, 2, \dots, n-1 \quad (1)$$

$$\dot{x}_n(t) = f_n[\mathbf{u}(t), y(t)] \quad ; \quad i = n$$

and an output equation

$$y(t) = g_y^{-1}[g_u(\mathbf{u}(t)) + x_1(t), \mathbf{u}(t)]. \quad (2)$$

(The state-space representation is substantially changed in

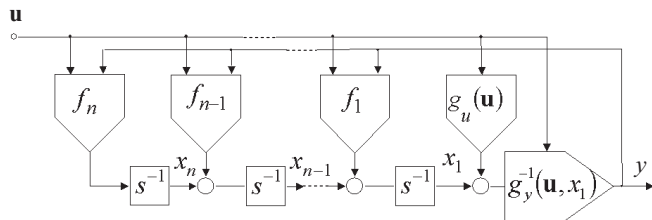


Fig. 1: State space diagram of the identified model

comparison with [1].) Let us suppose that \mathbf{u} and y are the only measurable quantities in the system. The state-space diagram must fulfill the following rules:

1. The basic string of the scheme consists of n integrators followed by an algebraic block $g_y^{-1}(\mathbf{u}, x_1)$.
2. There exists an inverse function $g_y(\mathbf{u}, y)$ of the output function $g_y^{-1}(\mathbf{u}, x_1)$ from Eq. (2).
3. All the direct paths of the scheme lead from the output source \mathbf{u} on the basic string.
4. All the feedback loops of the scheme lead via the output algebraic block $g_y^{-1}(\mathbf{u}, x_1)$ and none of them is algebraic, i.e., all of them contain at least one integrator.

Then we can defined

$$x_1(t) = -f_0(\mathbf{u}(t), y(t)) \quad (3)$$

where

$$f_0(\mathbf{u}(t), y(t)) = g_u(\mathbf{u}(t)) - g_y(\mathbf{u}(t), y(t)). \quad (4)$$

The functions f_i are supposed to be linear in parameters $a_{i,j}$, i.e., to have the form

$$f_i(\mathbf{u}(t), y(t)) = \sum_{j=1}^{m_i} a_{i,j} \cdot g_{i,j}(\mathbf{u}(t), y(t)); \quad i = 0, 1, \dots, n, \quad (5)$$

where $g_{i,j}$ are known (generally nonlinear) functions of the measured data and $a_{i,j}$ (generally unknown) constants.

Example:

$$f_3(\mathbf{u}, y) = a_{3,1} \cdot u_1 \cdot y - a_{3,2} \cdot y^3 + a_{3,3} \cdot \cos(u_1 + u_2 - y).$$

2 Identification algorithm

For the sake of simplicity, let us denote a multiple integral of a time-dependent variable $x(t)$ as

$$\mathcal{I}(x, i, t) = \int_0^t \int_0^{t_1} \dots \int_0^{t_{i-1}} x(t_i) dt_i dt_{i-1} \dots dt_1; \quad i > 0 \quad (6)$$

$$\mathcal{I}(x, 0, t) = x(t) \quad i = 0.$$

Note that (6) corresponds to a time response of a chain of i integrators in series, with zero initial conditions, excited by the input function $x(t)$. The first determined integral from 0 to t of the last state-space variable $x_1(t)$ in Fig.1 can be expressed as

$$\mathcal{I}(x, 1, t) = \sum_{i=1}^n \left[x_i(0) \frac{t^i}{i!} + \mathcal{I}(f_i(\mathbf{u}(t), y(t)), i+1, t) \right]. \quad (7)$$

From (3), (4), (7) follows

$$\sum_{i=1}^n x_i(0) \frac{t^i}{i!} + \sum_{i=0}^n \mathcal{I}[f_i(\mathbf{u}(t), y(t)), i+1, t] = 0. \quad (8)$$

Now let us integrate (8) for kT , $k = 1, 2, \dots, n+2$ and multiply both sides of the k -th equation by a binomial coefficient $(-1)^k \binom{k}{n+2}$. Let us make a sum of the resulting $(n+2)$ equations to obtain

$$\sum_{k=1}^{n+2} (-1)^k \binom{k}{n+2} \cdot \left[\sum_{i=1}^n x_i(0) \frac{(kT)^i}{i!} + \sum_{i=0}^n \mathcal{I}[f_i(\mathbf{u}(t), y(t)), i+1, t] \right] = 0. \quad (9)$$

Because

$$\sum_{k=1}^n (-1)^k \binom{k}{m} k^i = 0; \quad i = 1, 2, \dots, m-1, \quad (10)$$

the generally non-measurable initial conditions $x_i(0)$ disappear from (9) and we obtain

$$\sum_{k=1}^{n+2} (-1)^k \binom{k}{n+2} \sum_{i=0}^n \mathcal{I}[f_i(\mathbf{u}(t), y(t)), i+1, t] = 0. \quad (11)$$

Let us denote the weighted sum of multiple determined integrals (6) (after enumeration of this expression we obtain a scalar constant) as

$$\mathcal{C}(x, i, n, T) = \sum_{k=1}^{n+2} (-1)^k \binom{k}{n+2} \mathcal{I}(x, i+1, kT). \quad (12)$$

From (5), (11), (12) follows the linear algebraic equation for the unknown coefficients $a_{i,j}$

$$\sum_{i=0}^n \sum_{j=1}^{m_i} a_{i,j} \mathcal{C}[g_{i,j}(\mathbf{u}(t), y(t)), i, n, T] = 0 \quad (13)$$

The weighted sums \mathcal{C} of determined integrals (12) of the (generally nonlinear) functions $g_{i,j}(\mathbf{u}, y)$ represent in (13) the known linear coefficients of the algebraic linear equation for the unknown constants $a_{i,j}$. By shifting the time origin of the input – output data and repeating algorithm (13) we obtain another algebraic equation. After repeating this process enough times we obtain a system of algebraic equations that can be solved for the unknown constants $a_{i,j}$, for example by the least squares method. To obtain a unique solution, one or more constants $a_{i,j}$ must be known. These constants are then put on the right side of system (13). The MATLAB program MI, described in part 4, was designed to enable such computation.

3 Examples

A. A linear system, described by a transfer function

$$\frac{Y(s)}{U(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \dots + b_n}{a_0 s^n + a_1 s^{n-1} + \dots + a_n} \quad (14)$$

can be described by the state-space diagram in Fig.2. From the canonical forms this is the unique scheme corresponding to the structure shown in Fig. 1. State-space equations of the system are

$$\begin{aligned} \dot{x}_i &= x_{i+1} + b_i u - a_i y; \quad i = 1, 2, \dots, n-1 \\ \dot{x}_n &= b_n u - a_n y \end{aligned} \quad (15)$$

and the output equation is

$$y = (x_1 + b_0 u) \cdot a_0^{-1}. \quad (16)$$

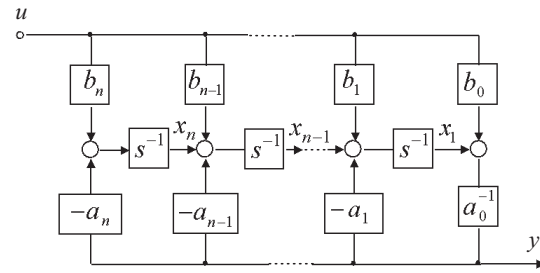


Fig. 2: State-space diagram of the linear system

Linear algebraic equations for calculating the unknown system parameters are

$$\sum_{i=0}^n [b_i \mathcal{C}(u, i, n, T) - a_i \mathcal{C}(y, i, n, T)] = 0. \quad (17)$$

An example of an identification algorithm for a linear system of the fourth order can be found in [1].

B. A parallel dynamo (Fig. 3) is a nonlinear system with two input variables: resistance of the excitation winding R and

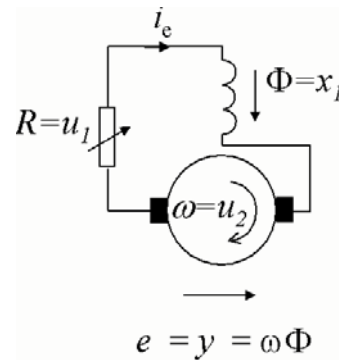


Fig. 3: Parallel dynamo

angular velocity of rotation ω . The output is dynamo voltage e . (The dynamo is supposed without any external load). The differential equation for the system dynamics is

$$e = \omega \Phi = R i_e + N \frac{d\Phi}{dt}, \quad (18)$$

where $e = \omega \Phi$ is the induced voltage, Φ is the machine flux, $N \frac{d\Phi}{dt}$ is the voltage induced by the change of the machine flux in the excitation winding, and $R i_e$ is the voltage drop on the resistance in the excitation circuit.

The inverse magnetization curve $i_e(\Phi)$ is supposed in the form

$$i_e(\Phi) = a\Phi + b\Phi^3. \quad (19)$$

Let us define the variables and constants to correspond to the symbols introduced in part 1.

$$\begin{aligned} u_1 &= R; \quad u_2 = \omega; & x_1 &= \Phi; & y &= e = u_2 \cdot x_1; \\ a_{1,1} &= \frac{1}{N}; & a_{1,2} &= \frac{a}{N}; & a_{1,3} &= \frac{b}{N}. \end{aligned} \quad (20)$$

The state-space diagram in Fig. 4 corresponds to (18). It does not fulfill the condition of closing the feedback loops via

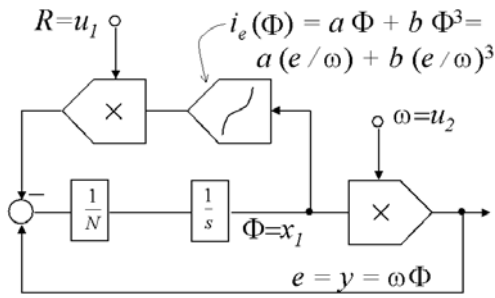


Fig. 4. State-space diagram of the parallel dynamo

the measurable output variable y (item 4, page 64). The diagram must therefore be restructured, as in the Fig. 5. Then it

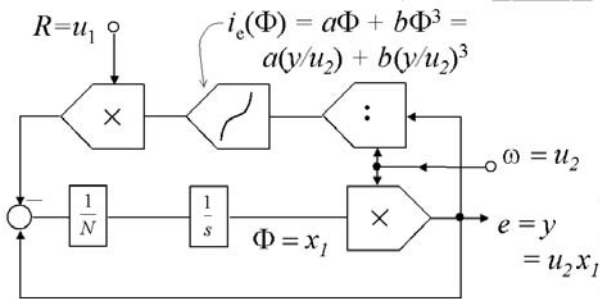


Fig. 5: Restructured state-space diagram of the parallel dynamo

corresponds to a state-space equation

$$\dot{x}_1 = a_{1,1}y - a_{1,2}u_1 \frac{y}{u_2} - a_{1,3}u_1 \left(\frac{y}{u_2}\right)^3 \quad (21)$$

The negative inverse output function is then

$$f_0(u, y) = -\frac{y}{u_2} \quad (22)$$

The algebraic equation (13) for computing the unknown parameters is

$$\begin{aligned} &C\left(-\frac{y}{u_2}, 0, 1, T\right) + a_{1,1}C(y, 1, 1, T) + a_{1,2}C\left(-u_1 \frac{y}{u_2}, 1, 1, T\right) + \\ &+ a_{1,3}C\left(-u_1 \left(\frac{y}{u_2}\right)^3, 1, 1, T\right) = 0 \end{aligned} \quad (23)$$

In original notation:

$$\begin{aligned} &\left[-2 \int_0^T \frac{e(t)}{\omega(t)} dt + \int_0^{2T} \frac{e(t)}{\omega(t)} dt \right] + \\ &+ \frac{1}{N} \left[-2 \int_0^T \int_0^{t_2} e(h_1) dh_1 dt_2 + \int_0^{2T} \int_0^{t_2} e(h_1) dh_1 dt_2 \right] + \\ &- \frac{a}{N} \left[-2 \int_0^T \int_0^{t_2} R(h_1) \frac{e(h_1)}{\omega(h_1)} dh_1 dt_2 + \int_0^{2T} \int_0^{t_2} R(h_1) \frac{e(h_1)}{\omega(h_1)} dh_1 dt_2 \right] + \\ &- \frac{b}{N} \left[-2 \int_0^T \int_0^{t_2} R(h_1) \left(\frac{e(h_1)}{\omega(h_1)}\right)^3 dh_1 dt_2 + \int_0^{2T} \int_0^{t_2} R(h_1) \left(\frac{e(h_1)}{\omega(h_1)}\right)^3 dh_1 dt_2 \right] = 0. \end{aligned} \quad (24)$$

4 Identification program MI and its use

The identification program (available in [4]) is written for one-shot identification (identification *off-line*). The basic condition for using the program is to find a proper structure of the model, corresponding to the state-space diagram in Fig. 1. This is not always a simple task and sometimes we need to transform the physical reality, as we did it in the case of the parallel dynamo in item B of the previous part. The other necessity is the existence of relevant data \mathbf{u}, \mathbf{y} in the form of time vectors. The data must contain a considerable content of frequencies interesting from the point of view of the future model. (These frequencies often lie around the critical frequency ω_k , i.e., during the measurement, at least for a short time, the identified system should oscillate in phase opposition.) The shortest integration period T , which is one of the items of the program input data, has to be (see [1]) $T_{\min} \leq \pi/\omega_k$. By Shannon's theorem, the data ought to have at least ten samples in the critical period. This implies for the sample period $h \leq 0.2 T_{\min}$. To obtain one algebraic equation (13), a time sequence $(n+2)T$ of data is necessary. In the program, the time origin of the data for a new equation (13) is shifted by $T/2$. This implies that from the time period t_{\max} , approximately

$$n_e(T) = 2(t_{\max} - (n+2)T)/T + 1 = 2t_{\max}/T - 2n - 3$$

equations will be obtained (an approximation is given by rounding). More than one integration period can be put for one program call, and for each of them its own system of $n_e(T)$ algebraic equations (13) will be obtained. Before the final computing of the parameters, all these systems are joined together and the parameters are calculated from this joint system by the method of least squares. By a proper choice of the integration period vector T we can freely select the filtration properties of the kernels (see [1] or [4] for further details). It is recommended to choose the integration periods in a geometric series beginning by $T_{\min} = \pi/\omega_k$ and with quotient $q \approx 1.5 \div 4$. The program displays on the screen the numbers of equations used for the given integration period T , the sum of these numbers, the root mean square error of the complete equation system, and the singular numbers of its matrix. The singular numbers can serve as a measure of the system conditionality number and, consequently, the error of the calculated parameters. Let us suppose in the following that the model contains m independent nonlinear blocks $g_{i,j}$ from the equation (5). These blocks must be programmed as vector functions of the measured data vectors (by the period convention of MATLAB).

The program is called by a statement

$$\mathbf{parameters} = \mathbf{mi}(\mathbf{h}, \mathbf{T}, \mathbf{staircase}, \mathbf{data}, \mathbf{c})$$

where \mathbf{h} is the sampling periods of the measured data

\mathbf{T} is the row vector of the sampling periods

$\mathbf{staircase}$ is a row m -vector of data indicators;

for $\mathbf{staircase}(i) = 0$ the corresponding data will be integrated as continuous, by Euler's method,

for $\mathbf{staircase}(i) = 1$ the corresponding data will be integrated as stepwise with a valid value at the end of the sampling period,

for $\mathbf{staircase}(i) = -1$ the corresponding data will be integrated as stepwise with a valid value at the beginning of the sampling period;

data is a matrix consisting of m column vectors of (generally) nonlinear functions of measured data, corresponding to the functions from equation (5);

c is a matrix describing the model structure. It consists of m columns, corresponding to the columns of the matrix **data**. Absolute values of the nonzero members in the columns correspond to the index $(i + 1)$, i.e., to the number of integrators between the block $g_{i,j}$ and the final state equation x_1 at the end of the integrator chain in Fig. 1, augmented by one. The sign of the member corresponds to the sign of the connection. The coefficient $c(1, 1)$ corresponds obligatorily to the known parameter (here one), and the corresponding sum of weighted integrals will be put with inverted sign on the right side of equation (13). We must complete matrix c by zero members to a rectangular shape. These members are of no significance for the computation;

parameters is the output row vector. It is ordered in accordance with the columns of matrix c , specifically according to the order of its nonzero elements. Unitary coefficient corresponding to $c(1, 1)$ does not appear in the output vector.

Examples

A. The linear system with a transfer function $\frac{b_1s + b_2}{s^2 + a_1s + a_2}$

corresponds to the state-space diagram in Fig. 2 for $n = 2$. The corresponding algebraic equations (13), built from the weighted integrals, will have the form of (17). For identification, column data vectors of measured data $\mathbf{u}(t)$ and $\mathbf{y}(t)$ will be necessary. Parameter a_0 is selected as unitary in our transfer function. In the state-space diagram, Fig. 2, this parameter is connected with the last state-space variable x_1 only by an algebraic connection (without integrators), and in the equations it appears with negative sign. Due to this reasons the matrix **data** will be ordered as $[\mathbf{y}, \mathbf{u}]$, and in the structure matrix c the first member will be $c(1, 1) = -1$. If the

structure matrix has the form $c = \begin{bmatrix} -1 & 2 \\ -2 & 3 \\ -3 & 0 \end{bmatrix}$, we obtain the result

(output vector) $\mathbf{parameters} = [a_1, a_2, b_1, b_2]$. Parameter a_1 corresponds to the member -2 in c . Consequently, $\frac{a_2}{2}$ corresponds to -3 , b_1 to 2 and b_2 to 3 . Parameter 1 by s^2 in the transfer function denominator corresponds to the member -1 in c .

For the same measured data we can obtain the resulting transfer function in the form $\frac{b_1s + 1}{a_0s^2 + a_1s + a_2}$ with the inputs

$$\mathbf{data} = [\mathbf{u}, \mathbf{y}], c = \begin{bmatrix} 3 & -1 \\ 2 & -2 \\ 0 & -3 \end{bmatrix}$$

and the result will be $\mathbf{parameters} = [b_1, a_0, a_1, a_2]$.

B. Parallel dynamo – see Fig. 3, 4, 5 and Eqs. (19–25). If we have at our disposal the time vectors of the measured data \mathbf{u}_1 , \mathbf{u}_2 and \mathbf{y} (resistance in the excitation circuit R , angular velocity ω , output voltage of the dynamo e), and if we want to obtain the program output vector

$$\mathbf{parameters} = [a_{1,1}, a_{1,2}, a_{1,3}],$$

the input data matrix must be (description in MATLAB)

$$\mathbf{data} = [-\mathbf{y}./\mathbf{u}_2, \mathbf{y}, \mathbf{u}_1.*\mathbf{y}./\mathbf{u}_2, \mathbf{u}_1.*((\mathbf{y}./\mathbf{u}_2).^3)]$$

and the matrix of description of the model structure must be $c = [-1, 2, -2, -2]$. In the original notation the input data matrix is

$$\mathbf{data} = [e./\omega, e, R.*e./\omega, R.*((e./\omega).^3)]$$

and the output vector of results corresponds to the expression

$$\mathbf{parameters} = [1/N, a/N, b/N], \text{ see (20).}$$

5 Conclusions

A new modification of the methods presented in [1, 2, 3] has been presented, including the corresponding program. In comparison to [1, 2, 3], more general nonlinearities are permitted. A complete description of the method, more examples and corresponding programs can be found in [4].

References

- [1] John, J., Rauch, V.: *Nonlinear continuous system identification by means of multiple integration*. Acta Polytechnica, Vol. 39 (1999), No. 1, pp. 55–62
- [2] Maletínský, V.: *I-i-p Identifikation kontinuierlicher dynamischer Prozesse*. Abhandlung zur Erlangung des Titels eines Doktors der Technischen Wissenschaften der ETH Zürich, (In German: *I-i-p Identification of Continuous Dynamic Systems*. PhD thesis, ETH Zürich.), SSS Zürich 1978
- [3] Skočdopole, J.: *Nelineární vícenásobná integrace*. Kandidátská disertační práce FEL ČVUT 1981, (In Czech: *Nonlinear multiple integration*.) PhD thesis, FEE CTU Prague 1981
- [4] See: <http://dce.felk.cvut.cz/sri2/index.htm>

Doc. Ing. Jan John, CSc.
Department of Control Engineering
phone: +420 2 2435 7597
e-mail: john@control.felk.cvut.cz
Czech Technical University in Prague
Faculty of Electrical Engineering
Technická 2, 166 27 Praha 6, Czech Republic