KYBERNETIKA — VOLUME *50* (2014), NUMBER 4, PAGES 473–490

STATE ELIMINATION FOR NONLINEAR NEUTRAL STATE-SPACE SYSTEMS

Miroslav Halás and Pavol Bisták

The problem of finding an input-output representation of a nonlinear state space system, usually referred to as the state elimination, plays an important role in certain control problems. Though, it has been shown that such a representation, at least locally, always exists for both the systems with and without delays, it might be a neutral input-output differential equation in the former case, even when one starts with a retarded system. In this paper the state elimination is therefore extended further to nonlinear neutral state-space systems, and it is shown that also in such a case an input-output representation, at least locally, always exists. In general, it represents a neutral system again. Computational aspects related to the state elimination problem are discussed as well.

Keywords: nonlinear time-delay systems, neutral systems, input-output representation, linear algebraic methods, Gröbner bases

Classification: 93C10, 34K40, 93B25

1. INTRODUCTION

The majority of techniques and methods for the analysis and synthesis of nonlinear control systems, taking into account the systems with delays as well, consider as a starting point a state-space representation. However, knowing an input-output representation is sometimes essential for finding a solution to a specific control problem. Typical representatives are various system equivalence and controllability problems [18, 19, 22, 26, 30], model matching problems [8, 15, 16], observer design [9, 14, 21], observability and identifiability of parameters [1, 29], and others. Eventually, any transfer function approach to a control problem deals, in general, with input-output properties of a system. See for instance [17, 25] for the applications to the systems with delays, and for instance [10, 11] for the extension of the transfer function formalism to the nonlinear case.

In the linear case any system described by state equations can equivalently be described by an input-output differential equation. From that point of view Laplace transforms play a key role. In the nonlinear case the situation is more complicated, and several techniques have been developed to find the corresponding input-output equations, see for instance [7] or [5]. The latter shows that for a given state-space representation a corresponding set of input-output equations can be, at least locally, always constructed by

DOI: 10.14736/kyb-2014-4-0473

applying a suitable change of coordinates. The problem is usually referred to as the state elimination. The idea of the state elimination has recently been carried over by [1] to nonlinear systems with delays, and it has been shown that even for a state-space system with delays there always exists, at least locally, a set of input-output differential-delay equations. However, such a state elimination might result in a set of input-output equations representing a system of neutral type, even when one starts with the state-space equations being of retarded type. Note that by retarded one means a classical (non-neutral) system and by neutral a system having delays in the highest derivative. As an example, one can consider a first order system plus dead time which when combined with an (ideal) PID controller generate a closed loop system being of neutral type.

That the nonlinear time-delay systems can have an input-output representations of a neutral type was discovered in [16] and then, in details, studied in [13]. Though, this can also be suspected from the inversion algorithm of [20]. It represents a strictly nonlinear phenomenon, for this cannot happen in the linear time-delay case where retarded systems always admit an input-output representation of retarded type. Since the state elimination can for a nonlinear retarded state-space system yield a neutral input-output equation, the natural question can be asked whether it is possible to extend the ideas of [1, 13] further to neutral state-space systems, and, if so, what type of an input-output representation one gets. The aim of this work is to give an answer to that question and to show that even for a nonlinear neutral state-space system there always exists, at least locally, an input-output differential-delay representation which is, in general, of neutral type again.

The preliminary discussion on this topic has been given in the conference paper [12], and there are contact points to the results of [13] as well. With respect to these works, in this paper the results are extended, in general, to multi-input multi-output nonlinear neutral systems, while in both [12] and [13] the attention was restricted to the single-output case only (the latter discussing retarded state-space systems only). Additionally, the problem solution is reformulated here such that it does not rely on applying the Poincaré lemma. Therefore, one does not need to inspect the integrability of the respective differential one-forms, as the Frobenius theorem is not available for infinite dimensional systems. Finally, the computational aspects of the problem of finding an input-output representation for a nonlinear neutral system are discussed as well. In that respect the use of Gröbner bases technique plays a key role for the polynomial and rational systems.

2. PRELIMINARIES

In this paper we use the mathematical setting of [1, 13, 21, 28] extended hereinafter to the nonlinear neutral state-space systems.

The nonlinear neutral state-space systems considered in this paper are objects of the form

$$\begin{aligned} \dot{x}(t) &= f\left(\{\dot{x}(t-i), x(t-j), u(t-k); i>0; j,k \geq 0\}\right) \\ y(t) &= h\left(\{x(t-i); i \geq 0\}\right) \end{aligned}$$

where the entries of f and h are real meromorphic functions, and $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, and $y(t) \in \mathbb{R}^p$ denote state, input, and output to the system respectively.

Remark 2.1. We assume the system has commensurable delays. Hence, it is not restrictive to consider $i, j, k \in \mathbb{Z}_+$, for all commensurable delays can be interpreted as multiples of some elementary delay τ.

Let us denote by $\mathcal{K}$ the field of real meromorphic functions of real variables $\{\dot{x}(t-i), x(t-j), u^{(l)}(t-k); i > 0; j, k, l \geq 0\}$ and by

$$\mathcal{E} = \mathrm{span}_{\mathcal{K}}\{\mathrm{d}\xi(t); \xi(t) \in \mathcal{K}\}$$

the formal vector space of differential one-forms. That is, the standard differentials of the functions in $\mathcal{K}$ are elements of $\mathcal{E}$.

Let σ denote now the delay operator defined as

$$\sigma(\xi(t)) = \xi(t-1)$$

for any $\xi(t) \in \mathcal{K}$. The operator σ induces the delay operator, which is by abuse of notation denoted by the same symbol σ, that acts on $\mathcal{E}$ as follows. Let $v(t) = \sum_i \alpha_i(t)\mathrm{d}\xi_i(t)$ be in $\mathcal{E}$, then

$$\sigma(v(t)) = \sum_i \alpha_i(t-1)\mathrm{d}\xi_i(t-1).$$

The delay operator σ also induces the (non-commutative) skew polynomial ring[1] $\mathcal{K}(\delta]$ with the usual addition and the non-commutative multiplication given by the commutation rule

$$\delta\,\xi(t) = \xi(t-1)\delta$$

for any $\xi(t) \in \mathcal{K}$. Hence, the elements of $\mathcal{K}(\delta]$ represent the linear ordinary time-delay operators that act on any $v(t) \in \mathcal{E}$ as follows. Let $p(\delta] = \sum_{i=0}^{k} a_i(t)\delta^i \in \mathcal{K}(\delta]$, then

$$p(\delta]v = \left(\sum_{i=0}^{k} a_i(t)\delta^i\right) v(t) = \sum_{i=0}^{k} a_i(t)\sigma^i(v(t)) = \sum_{i=0}^{k} a_i(t)v(t-i).$$

In plain words, δ is interpreted as a time-delay operator σ and we thus have $\delta\,\xi(t) = \xi(t-1)\delta$ for any $\xi(t) \in \mathcal{K}$, but $\delta\,v(t) = v(t-1)$ for any $v \in \mathcal{E}$.

Example 2.2. To any function in $\mathcal{K}$, and therefore also to any one-form in $\mathcal{E}$, one can now associate polynomials from the ring $\mathcal{K}(\delta]$. For example, let $\varphi(t) = x(t-1)+x(t-2)^2$. Then

$$\begin{aligned}
\mathrm{d}\varphi(t) &= \mathrm{d}x(t-1) + 2x(t-2)\mathrm{d}x(t-2)\\
\mathrm{d}\varphi(t) &= \delta\,\mathrm{d}x(t) + 2x(t-2)\delta^2\,\mathrm{d}x(t)\\
\mathrm{d}\varphi(t) &= (\delta + 2x(t-2)\delta^2)\,\mathrm{d}x(t).
\end{aligned}$$

The ring $\mathcal{K}(\delta]$ is Noetherian and a left Ore domain.

Lemma 2.3. (Ore condition) For all non-zero $a(\delta], b(\delta] \in \mathcal{K}(\delta]$ there exist non-zero $a_1(\delta], b_1(\delta] \in \mathcal{K}(\delta]$ such that $a_1(\delta]b(\delta] = b_1(\delta]a(\delta]$.

[1]The notation “( ]“ is used only to highlight the non-commutativity of a ring or a polynomial. It is used here in order to be consistent with the notation introduced in the earlier works from this area.

See [23, 24], and for instance [28] for the case of nonlinear systems with delays.

The properties of the nonlinear neutral state-space systems can now be analyzed by introducing the formal module

$$\mathcal{M} = \operatorname{span}_{\mathcal{K}(\delta]}\{\mathrm{d}\xi(t); \xi(t) \in \mathcal{K}\}.$$

The rank of a module over the left Ore domain $\mathcal{K}(\delta]$ is well-defined [4].

Remark 2.4. In plain words, one can now work with polynomials from the non-commutative polynomial ring $\mathcal{K}(\delta]$ rather than with the delays of the respective variables, see Example 2.2.

To simplify the notation we often denote $\xi(t) \in \mathcal{K}$ by ξ only and a meromorphic function $\varphi(\xi_1(t), \ldots, \xi_1(t-i_1), \ldots, \xi_k(t), \ldots, \xi_k(t-i_k))$, where $i_1, \ldots, i_k$ are nonnegative, by $\varphi(\delta, \xi_1, \ldots, \xi_k)$ only. Using this notation, the nonlinear neutral state-space systems studied in this paper are objects of the form

$$\begin{array}{rcl} \dot{x} & = & f(\delta, \dot{x}, x, u) \\ y & = & h(\delta, x). \end{array} \tag{1}$$

For the sake of simplicity, we also introduce the following notation. Let φ be an r-dimensional vector with entries $\varphi_j(\delta, \xi_1, \ldots, \xi_k)$. Then, let $\partial\varphi/\partial\xi$ denote the $r \times k$ matrix with entries

$$\left(\frac{\partial\varphi}{\partial\xi}\right)_{j,i} = \sum_{\ell\geq 0} \frac{\partial\varphi_j}{\partial\xi_i(t-\ell)}\delta^\ell \in \mathcal{K}(\delta]$$

for $i = 1, \ldots, k$ and $j = 1, \ldots, r$.

Remark 2.5. Note that this notation allows us to simply write for instance $\mathrm{d}\varphi = \partial\varphi/\partial x \, \mathrm{d}x$ in Example 2.2, with $\partial\varphi/\partial x = \delta + 2x(t-2)\delta^2$.

Finally, for the system (1) we assume that

$$\operatorname{rank}_{\mathcal{K}(\delta]} \frac{\partial(\dot{x} - f(\cdot))}{\partial\dot{x}} = n. \tag{2}$$

Remark 2.6. The above assumption is not restrictive as it only means we, indeed, have a dynamical system here. In the linear counterpart it means that the system is of the form $E[\delta]\dot{x} = A[\delta]x + B[\delta]u$ with $E[\delta]$ full rank, where $A[\delta]$, $B[\delta]$, and $E[\delta]$ are matrices with entries in the (commutative) polynomial ring $\mathbb{R}[\delta]$.

Remark 2.7. Strictly speaking, the term $\dot{x} - f(\cdot)$ in (2) is zero by (1). However, for checking the rank condition we understand it formally as a nonzero function.

2.1. Notation and symbols

To help the reader get through the paper we summarize the notation and symbols used in the paper.

x or $x(t)$	state of the system
u or $u(t)$	input of the system
y or $y(t)$	output of the system
$\mathbb{Z}_+$	set of nonnegative integers
$\mathbb{R}$	field of real numbers
$\mathcal{K}$	field of real meromorphic functions (of real variables)
$\mathcal{E}$	formal vector space of differential one-forms
$\dot{\xi}$	derivative of ξ with respect to t
σ	delay operator that takes t to $t-1$
δ	indeterminate of a polynomial ring representing the action of the operator σ
$\mathbb{R}[\delta]$	commutative ring of polynomials in δ over $\mathbb{R}$
$\mathcal{K}(\delta]$	non-commutative ring of polynomials in δ over $\mathcal{K}$
$\mathcal{M}$	formal module of differential one-forms
$\varphi(\delta, \xi_1, \ldots, \xi_k)$	stands for $\varphi(\xi_1(t), \ldots, \xi_1(t-i_1), \ldots, \xi_k(t), \ldots, \xi_k(t-i_k))$
$\mathrm{span}_{\mathcal{K}}\{\nu_1, \ldots, \nu_n\}$	set of all linear combinations of $\nu_1, \ldots, \nu_n$ over a field $\mathcal{K}$
$\mathrm{rank}_{\mathcal{K}(\delta]} M$	rank of a matrix M over the non-commutative ring $\mathcal{K}(\delta]$
$\deg p$	degree of a polynomial p
$k[z_1, \ldots, z_n]$	commutative ring of polynomials in $z_1, \ldots, z_n$ over a field k
$\langle f_1, \ldots, f_m \rangle$	ideal generated by polynomials $f_1, \ldots, f_m \in k[z_1, \ldots, z_n]$
$>_{lex}$	lexicographical ordering of monomials

3. STATE ELIMINATION

In this section we extend the state elimination of [1, 5, 13] further to the nonlinear neutral systems.

First, we explain intuitively certain aspects of the problem by the following introductory example.

Example 3.1. The following system was considered in [13]

$$\begin{aligned} \dot{x} &= u \\ y &= x^2 + x(t-1). \end{aligned}$$

Following the ideas of the state elimination for the systems without delays [5] one can compute

$$y = x^2 + x(t-1) \tag{3}$$
$$\dot{y} = 2xu + u(t-1). \tag{4}$$

However, one cannot go any further, since from (3) one cannot express x and substitute to (4). In other words, the equations (3) and (4) are independent over $\mathcal{K}$. However, they are dependent over $\mathcal{K}(\delta]$, for $\partial y/\partial x = \delta + 2x$, $\partial \dot{y}/\partial x = 2u$, and thus $\mathrm{rank}_{\mathcal{K}(\delta]}[\partial y/\partial x] = \mathrm{rank}_{\mathcal{K}(\delta]}[\partial(y, \dot{y})/\partial x] = 1$.

The latter implies that one can consider the equations (3) and (4), and a sufficient number of their time delays to get a set of dependent equations over $\mathcal{K}$. In concrete terms

$$\begin{aligned} y &= x^2 + x(t-1) \\ \dot{y} &= 2xu + u(t-1) \\ \dot{y}(t-1) &= 2x(t-1)u(t-1) + u(t-2). \end{aligned}$$

One can, therefore, eliminate $x(t-1)$ from the last equation which yields the input-output equation

$$4u^2u(t-1)y - u(t-1)(\dot{y} - u(t-1))^2 - 2u^2(\dot{y}(t-1) - u(t-2)) = 0. \tag{5}$$

However, (5) represents a neutral system. Note that the delay is present in the highest derivative of the system output.

Remark 3.2. The algebraic methods are very useful mainly for characterizing system structural properties. Hence, the neutrality of equation (5) needs to be understood from such a point of view (i. e. the equation is neutral in the sense that the delay is present in the highest derivative of y, and there is no other first-order input-output differential equation for the system not being neutral in this sense). However, this does not guarantee the system exhibits behaviour we expect of neutral systems (as for instance infinitely many oscillating frequencies). Such aspects can possibly be verified using rather analytical than algebraic tools.

Example 3.3. Naturally, one can expect similar problems (as in Example 3.1) appear when working with the neutral state-space systems. Consider, for instance, the system

$$\dot{x} = \dot{x}(t-1) \tag{6}$$

$$y = x^2. \tag{7}$$

Then

$$\dot{y} = 2x\dot{x} \tag{8}$$

and we observe that we need to eliminate $\dot{x}$ first. Unfortunately, $\dot{x}$ cannot be expressed from (6) directly, as the system is neutral. Technically speaking, the equations (6) and (8) are independent over $\mathcal{K}$. However, they are dependent over $\mathcal{K}(\delta]$, for $\partial(\dot{x} - \dot{x}(t-1))/\partial\dot{x} = 1 - \delta$, $\partial\dot{y}/\partial\dot{x} = 2x$, and thus $\operatorname{rank}_{\mathcal{K}(\delta]}[\partial(\dot{x} - \dot{x}(t-1))/\partial\dot{x}] = \operatorname{rank}_{\mathcal{K}(\delta]}[\partial(\dot{x} - \dot{x}(t-1), \dot{y})/\partial\dot{x}] = 1$. This suggests us to apply a similar idea as in Example 3.1. Here, one can take the equations (6) and (8), and a sufficient number of their time delays, namely

$$\begin{aligned} \dot{x} &= \dot{x}(t-1) \\ \dot{y} &= 2x\dot{x} \\ \dot{y}(t-1) &= 2x(t-1)\dot{x}(t-1) \end{aligned}$$

which constitute now a set of dependent equations over $\mathcal{K}$. Therefore, $\dot{x}(t-1)$ can be eliminated from the last equation

$$\dot{y}x(t-1) - x\dot{y}(t-1) = 0. \tag{9}$$

Finally, we can continue with the elimination of x. This time, the equations (7) and (9) are considered. Again, the equations are independent over $\mathcal{K}$ but dependent over $\mathcal{K}(\delta]$. By considering a sufficient number of their time delays, in this case

$$\begin{aligned} y &= x^2 \\ y(t-1) &= x^2(t-1) \\ x\dot{y}(t-1) - \dot{y}x(t-1) &= 0 \end{aligned}$$

we can eliminate x and $x(t-1)$ from the last equation which gives us the input-output representation

$$\dot{y}^2(t-1)y - \dot{y}^2 y(t-1) = 0$$

for the system.

The ideas presented in the example can technically be generalized as follows.

3.1. State elimination for neutral systems

It will be shown now that for the system (1) there always exists, at least locally, a set of input-output differential-delay equations of the form

$$F_i(\delta, y, \dot{y}, \ldots, y^{(k)}, u, \dot{u}, \ldots, u^{(l)}) = 0 \tag{10}$$

with $i = 1, \ldots, p,\ k, l \geq 0$, and F_i being meromorphic, such that any pair (y, u) that solves the system (1) also satisfies (10).

Remark 3.4. The notion "locally" refers, in general, to the local nature of a problem one studies. That is, the solution found is valid in the neighborhood of any point from some suitable set, while there might not exist a global solution valid for all the points from such a set at the same time.

Theorem 3.5. Given a system of the form (1), there exist an integer l and an open and dense subset V of $\mathbb{R}^{n+ml}$ such that in the neighborhood of any point of V there exists an input-output representation of the system of the form (10).

Proof. The proof is constructive and is presented as an algorithm.

Step 0

For the system (1) set $s_i := 0$ and $h_i^{s_i} := y_i - h_i(\delta, x)$ for all $i = 1, \ldots, p$. Remark that s_i in $h_i^{s_i}$ is understood only as a superscript.

Set $i := 1$.

Step 1

If $i > p$ go to End.
If $\partial h_i^{s_i}/\partial x = 0$ then $i := i + 1$, and go to Step 1.
Compute $\dot{h}_i^{s_i}$. Under the assumption (2), one has

$$\mathrm{rank}_{\mathcal{K}(\delta]} \frac{\partial(\dot{x} - f(\cdot))}{\partial \dot{x}} = \mathrm{rank}_{\mathcal{K}(\delta]} \frac{\partial(\dot{x} - f(\cdot), \dot{h}_i^{s_i})}{\partial \dot{x}}.$$

Thus, there exists a nonzero polynomial $p(\delta] \in \mathcal{K}(\delta]$ such that

$$p(\delta]\frac{\partial \dot{h}_i^{s_i}}{\partial \dot{x}} \in \operatorname{span}_{\mathcal{K}(\delta]}\left\{\frac{\partial(\dot{x}-f(\cdot))}{\partial \dot{x}}\right\}.$$

That is

$$p(\delta]\frac{\partial \dot{h}_i^{s_i}}{\partial \dot{x}} = \sum_{j=1}^{n} p_j(\delta]\frac{\partial(\dot{x}_j - f_j(\cdot))}{\partial \dot{x}} \tag{11}$$

for some $p_j(\delta] \in \mathcal{K}(\delta]$, $j = 1, \dots, n$. The polynomials $p(\delta]$ and $p_j(\delta]$, $j = 1, \dots, n$, can be found by Ore condition, and one can, without loss of generality, assume that they have no common (left) factors other than 1.

Therefore, there exists a finite set of equations

$$\begin{array}{rcl} \dot{x}_1 - f_1 & = & 0 \\ & \vdots & \\ \dot{x}_n - f_n & = & 0 \\ \dot{h}_i^{s_i} & = & 0 \end{array} \quad \cdots \quad \begin{array}{rcl} \dot{x}_1(t-\tau_1) - f_1(t-\tau_1) & = & 0 \\ & \vdots & \\ \dot{x}_n(t-\tau_n) - f_n(t-\tau_n) & = & 0 \\ \dot{h}_i^{s_i}(t-\tau) & = & 0 \end{array} \tag{12}$$

where the highest delay considered for the respective equation is given by the degree of the respective polynomial in (11). That is, $\tau_j = \deg p_j(\delta]$, and $\tau = \deg p(\delta]$. Since (11) holds, the set of equations is (algebraically) dependent, and the variables $\dot{x}_j, \dots, \dot{x}_j(t-\tau_j)$, $j = 1, \dots, n$, can be eliminated from the equation $\dot{h}_i^{s_i}(t-\tau) = 0$. Proceed with the elimination, set $s_i := s_i + 1$, and denote by $h_i^{s_i}$ the left hand side of the equation $\dot{h}_i^{s_i-1}(t-\tau) = 0$ where $\dot{x}_j, \dots, \dot{x}_j(t-\tau_j)$, $j = 1, \dots, n$, have been eliminated. (Note that since s_i has been incremented the eqaution $\dot{h}_i^{s_i-1}(t-\tau) = 0$ is now identical with the last equation in (12)).

Step 2

If

$$\begin{aligned} &\operatorname{rank}_{\mathcal{K}(\delta]}\frac{\partial(h_1^0, \dots, h_1^{s_1-1}, \dots, h_i^0, \dots, h_i^{s_i-1})}{\partial x} \\ &< \operatorname{rank}_{\mathcal{K}(\delta]}\frac{\partial(h_1^0, \dots, h_1^{s_1-1}, \dots, h_i^0, \dots, h_i^{s_i})}{\partial x} \end{aligned}$$

go to Step 1.

If

$$\begin{aligned} &\operatorname{rank}_{\mathcal{K}(\delta]}\frac{\partial(h_1^0, \dots, h_1^{s_1-1}, \dots, h_i^0, \dots, h_i^{s_i-1})}{\partial x} \\ &= \operatorname{rank}_{\mathcal{K}(\delta]}\frac{\partial(h_1^0, \dots, h_1^{s_1-1}, \dots, h_i^0, \dots, h_i^{s_i})}{\partial x} \end{aligned} \tag{13}$$

and $i = p$, then go to End. Otherwise, $i := i + 1$, and go to Step 1.

End

The definition of s_i (13) implies[2] that for each $i = 1, \dots, p$ there exists a nonzero polynomial $b_i(\delta] \in \mathcal{K}(\delta]$ such that

$$b_i(\delta] \frac{\partial h_i^{s_i}}{\partial x} \in \operatorname{span}_{\mathcal{K}(\delta]} \left\{ \frac{\partial (h_1^0, \dots, h_1^{s_1-1}, \dots, h_i^0, \dots, h_i^{s_i-1})}{\partial x} \right\}.$$

That is

$$b_i(\delta] \frac{\partial h_i^{s_i}}{\partial x} = \sum_{r=1}^{i} \sum_{j=1}^{s_r} b_{r,j}(\delta] \frac{\partial h_r^{j-1}}{\partial x} \tag{14}$$

for some $b_{r,j}(\delta] \in \mathcal{K}(\delta]$. Therefore, there exists a finite set of equations

$$\begin{array}{rcl c rcl}
h_1^0 & = & 0 & & h_1^0(t-\tau_{1,1}) & = & 0 \\
 & \vdots & & & & \vdots & \\
h_1^{s_1-1} & = & 0 & & h_1^{s_1-1}(t-\tau_{1,s_1}) & = & 0 \\
 & \vdots & & & & \vdots & \\
h_i^0 & = & 0 & \cdots & h_i^0(t-\tau_{i,1}) & = & 0 \\
 & \vdots & & & & \vdots & \\
h_i^{s_i-1} & = & 0 & & h_i^{s_i-1}(t-\tau_{i,s_i}) & = & 0 \\
\hline
h_i^{s_i} & = & 0 & & h_i^{s_i}(t-\tau_i) & = & 0
\end{array} \tag{15}$$

where the highest delay considered for the respective equation is given by the degree of the respective polynomial in (14). That is, $\tau_{r,j} = \deg b_{r,j}(\delta]$, and $\tau_i = \deg b_i(\delta]$. Since (14) holds, the set of equations is (algebraically) dependent, and the variables $x_j, \dots, x_j(t-\tau_{r,j})$, for $j = 1, \dots, n$, and all $\tau_{r,j}$, can be eliminated from the equation $h_i^{s_i}(t-\tau_i) = 0$.

Finally, proceed with the elimination to get

$$F_i(\delta, y_1, \dots, y_1^{(s_1-1)}, \dots, y_i, \dots, y_i^{(s_i-1)}, y_i^{(s_i)}, u, \dot{u}, \dots, u^{(l)}) = 0$$

with $F_i \in \mathcal{K}$ and $l \geq 0$.

As a result, we have obtained an input-output representation for the system (1)

$$F_i(\delta, y, \dot{y}, \dots, y^{(k)}, u, \dot{u}, \dots, u^{(l)}) = 0$$

for some $k, l \geq 0$, and $i = 1, \dots, p$. □

Example 3.6. Consider the system

$$\begin{aligned}
\dot{x}_1 &= -\dot{x}_1(t-1) + x_2 \\
\dot{x}_2 &= \dot{x}_2(t-1)u \\
y &= x_1 - x_1(t-1).
\end{aligned}$$

[2]Note that it also implies the integer $s = s_1 + \cdots + s_p$ can be called an observability index.

Assumption (2) is satisfied

$$\operatorname{rank}_{\mathcal{K}(\delta]} \frac{\partial(\dot{x} - f(\cdot))}{\partial \dot{x}} = \operatorname{rank}_{\mathcal{K}(\delta]} \begin{pmatrix} \delta + 1 & 0 \\ 0 & -u\delta + 1 \end{pmatrix} = 2.$$

Step 0 First, we set $s_1 := 0$,

$$h_1^0 := y - x_1 + x_1(t-1)\ ,$$

and $i := 1$.

Step 1 We have $\partial h_1^0 / \partial x = (\delta - 1 \quad 0) \neq 0$.
Therefore, compute

$$\dot{h}_1^0 := \dot{y} - \dot{x}_1 + \dot{x}_1(t-1).$$

Note that $\partial \dot{h}_1^0 / \partial \dot{x} = (\delta - 1 \quad 0)$, and we have

$$p(\delta] \frac{\partial \dot{h}_1^0}{\partial \dot{x}} = p_1(\delta] \frac{\partial(\dot{x}_1 - f_1(\cdot))}{\partial \dot{x}} + p_2(\delta] \frac{\partial(\dot{x}_2 - f_2(\cdot))}{\partial \dot{x}}$$

where $p(\delta] = \delta + 1$, $p_1(\delta] = \delta - 1$, and $p_2(\delta] = 0$ can be found by Ore condition. Hence, we consider

$$\begin{aligned} \dot{x}_1 + \dot{x}_1(t-1) - x_2 &= 0 \\ \dot{x}_1(t-1) + \dot{x}_1(t-2) - x_2(t-1) &= 0 \\ \dot{y} - \dot{x}_1 + \dot{x}_1(t-1) &= 0 \\ \dot{y}(t-1) - \dot{x}_1(t-1) + \dot{x}_1(t-2) &= 0 \end{aligned}$$

which constitutes a set of (algebraically) dependent equations. The first three equations are used to eliminate $\dot{x}_1(t-1)$ and $\dot{x}_1(t-2)$ from the last equation, which yields

$$\dot{y}(t-1) + \dot{y} + x_2(t-1) - x_2 = 0.$$

Therefore $s_1 := 1$, and denote

$$h_1^1 := \dot{y}(t-1) + \dot{y} + x_2(t-1) - x_2.$$

Step 2 Since $\partial h_1^1 / \partial x = (0 \quad \delta - 1)$, we have

$$\operatorname{rank}_{\mathcal{K}(\delta]} \frac{\partial h_1^0}{\partial x} = 1 < \operatorname{rank}_{\mathcal{K}(\delta]} \frac{\partial (h_1^0, h_1^1)}{\partial x} = 2$$

Therefore go to Step 1, the condition (13) is not yet satisfied.

Step 1 Compute

$$\dot{h}_1^1 := \ddot{y}(t-1) + \ddot{y} + \dot{x}_2(t-1) - \dot{x}_2.$$

Now, $\partial \dot{h}_1^1 / \partial \dot{x} = (0 \quad \delta - 1)$, and we have

$$p(\delta] \frac{\partial \dot{h}_1^1}{\partial \dot{x}} = p_1(\delta] \frac{\partial(\dot{x}_1 - f_1(\cdot))}{\partial \dot{x}} + p_2(\delta] \frac{\partial(\dot{x}_2 - f_2(\cdot))}{\partial \dot{x}}$$

where the polynomials $p(\delta] = -u(t-1)\delta + \frac{1-u(t-1)}{1-u}$, $p_1(\delta] = 0$, and $p_2(\delta] = \delta + \frac{u(t-1)-1}{u-1}$ have been found by Ore condition.
Hence, we consider

$$\begin{array}{rcl} \dot{x}_2 - \dot{x}_2(t-1)u & = & 0 \\ \dot{x}_2(t-1) - \dot{x}_2(t-2)u(t-1) & = & 0 \\ \ddot{y}(t-1) + \ddot{y} + \dot{x}_2(t-1) - \dot{x}_2 & = & 0 \\ \ddot{y}(t-2) + \ddot{y}(t-1) + \dot{x}_2(t-2) - \dot{x}_2(t-1) & = & 0 \end{array}$$

which constitutes a set of (algebraically) dependent equations. The first three equations are used to eliminate $\dot{x}_2(t-1)$ and $\dot{x}_2(t-2)$ from the last equation, which yields

$$(\ddot{y}(t-1) + \ddot{y}(t-2))u(t-1)(u-1) - (u(t-1)-1)(\ddot{y} + \ddot{y}(t-1)) = 0.$$

Therefore $s_1 := 2$, and denote

$$h_1^2 := (\ddot{y}(t-1) + \ddot{y}(t-2))u(t-1)(u-1) - (u(t-1)-1)(\ddot{y} + \ddot{y}(t-1)).$$

Step 2 Now, the condition (13) is satisfied

$$\operatorname{rank}_{\mathcal{K}(\delta]} \frac{\partial(h_1^0, h_1^1)}{\partial x} = \operatorname{rank}_{\mathcal{K}(\delta]} \frac{\partial(h_1^0, h_1^1, h_1^2)}{\partial x} = 2$$

and since $i = p$, go to End.

End Note that $\partial h_1^2/\partial x = (0 \quad 0)$, and thus (14) holds for any nonzero polynomial $b_1(\delta] \in \mathcal{K}(\delta]$ (of degree 0), and $b_{1,1}(\delta] = b_{1,2}(\delta] = 0$. Therefore, we have found the input-output representation for the system

$$(\ddot{y}(t-1) + \ddot{y}(t-2))u(t-1)(u-1) - (u(t-1)-1)(\ddot{y} + \ddot{y}(t-1)) = 0$$

without necessity to proceed further with the elimination of states.

4. COMPUTATIONAL ASPECTS

As one can notice the problem of finding an input-output representation for the system of the form (1) consists of solving a set of equations of the form

$$g_i(\delta, x, \dot{x}, y, \dot{y}, \ldots, y^{(k)}, u, \dot{u}, \ldots, u^{(l)}) = 0 \tag{16}$$

for some $g_i \in \mathcal{K}$, and $i, k, l > 0$, where it is necessary to eliminate states, their derivatives and time delays. Depending on the type of the given system this process has to be repeated several times according to the number of steps in the algorithm introduced above. However, in order to find the input-output representation (10) it is not necessary to express the exact solution of each individual variable included in the set of equations (12) and, respectively, (15). In general, it suffices only to eliminate the respective variables from these sets of equations. In case the functions f and h in (1) are polynomial (or rational[3]), such a process of the elimination can be carried out by the Gröbner bases technique. Using this technique it is possible to transform one set of polynomial equations

$$\begin{aligned} g_1(z_1,\ldots,z_n) &= 0 \\ &\vdots \\ g_m(z_1,\ldots,z_n) &= 0 \end{aligned}$$

to another set

$$\begin{aligned} q_1(z_1,z_2,\ldots,z_n) &= 0 \\ q_2(z_2,\ldots,z_n) &= 0 \\ &\vdots \\ q_k(z_n) &= 0 \end{aligned}$$

for some $m,n,k>0$, and g_i,q_i being polynomial for all i, where the new set can be solved more easily. This technique has many applications in control theory (see e. g. [27]), and can be applied here to the elimination process of the system of polynomial equations (12) and (15) in order to get the input-output representation (10). That is, to find a representation with the variables corresponding only to the inputs, outputs, their derivatives, and time delays.

Remark 4.1. Note that Gröbner bases, due to the fact the transformed set of equations has an upper triangular form, generalize the usual Gauss reduction from linear algebra, the Euclidean algorithm for computation of univariate greatest common divisors and the simplex algorithm from linear programming [2, 6]. For instance, in the latter one aims to minimize a function of the form $c\cdot x$, subject to the condition $Ax=b$, where A,b,c are matrices with entries in $\mathbb{R}$ of appropriate dimensions. To do so, one needs to bring the following matrix

$$\begin{pmatrix} 1 & -c^T & 0 \\ 0 & A & b \end{pmatrix}$$

in to an upper triangular form.

B. Buchberger was the one who defined Gröbner bases and developed an algorithm for computing them, see e. g. [3]. The algorithm is a finite algorithm that takes in a finite set of generators for the ideal I in $k[z_1,\ldots,z_n]$ and returns a Gröbner basis [27]. It is a division type algorithm characterized by the aim to cancel leading terms of the respective polynomials and replace them with smaller ones. First, it is necessary to specify a total ordering of monomials in the polynomial ring $k[z_1,\ldots,z_n]$, where k is a field. For instance, the ordering defined by the ordinary lexicographical order $>_{lex}$, that is

$$z_1 > \ldots > z_n \tag{17}$$

can be used.

Gröbner bases can then be defined introducing the notion of a leading term. To simplify the notation we define $z^\alpha := z_1^{\alpha_1}z_2^{\alpha_2}\ldots z_n^{\alpha_n}$, where $\alpha=(\alpha_1,\ldots,\alpha_n)$ is in $\mathbb{Z}_+^n$.

[3]Note that any rational equation of the form (16) can easily be replaced by a polynomial one if we multiply it by the least common multiple of all the denominators.

Using this notation we can write a polynomial f in $k[z_1,\dots,z_n]$ as $f=\sum_\alpha a_\alpha z^\alpha$. Then the multidegree of f is defined as follows

$$\operatorname{multideg}(f)=\max\{\alpha\in\mathbb{Z}^n_+; a_\alpha\neq 0\}.$$

Note that according to the lexicographical ordering (17) one has $\alpha>_{lex}\beta$, where $\alpha=(\alpha_1,\dots,\alpha_n)$ and $\beta=(\beta_1,\dots,\beta_n)$ are in $\mathbb{Z}^n_+$, if in the vector $\alpha-\beta$ the leftmost nonzero element is possitive.

Let us denote the leading term of f by $LT(f)$. Then we have

$$LT(f)=a_{\operatorname{multideg}(f)}z^{\operatorname{multideg}(f)}.$$

Now a Gröbner basis can be defined as a finite subset $G=\{g_1,\dots,g_k\}$ of polynomials from an ideal $I\subset k[z_1,\dots,z_n]$ if the ideal generated by $LT(g_i)$, for $i=1,\dots,k$, equals the ideal generated by the leading terms of all the elements of I

$$\langle LT(g_1),\dots,LT(g_k)\rangle=\langle LT(I)\rangle.$$

It is important to emphasize that every nontrivial ideal I has a Gröbner basis G and any Gröbner basis G of I is a generating set of I.

In our case, the set of generators will be given by the left-hand side of equations (12) and (15), respectively.

A nice property of the Buchberger's algorithm is that it assures the variables are consequently eliminated, and the resulted Gröbner basis will include just one univariate polynomial. This follows from the elimination theory of Gröbner bases.

Let us define the jth elimination ideal of I as

$$I_j=I\cap k[z_{j+1},\dots,z_n]$$

where $j=0,\dots,n-1$, and we set $I_0=I$. If G is a Gröbner basis of I then

$$G_j=G\cap k[z_{j+1},\dots,z_n]$$

is a Gröbner basis of I_j, $j=0,\dots,n-1$. That is, G_j is given by the elements of G not involving $z_1,\dots,z_j$. Hence, G_{n-1} will depend only on z_n and, therefore, will finally determine the univariate polynomial that can be identified with the input-output representation (10).

Note that z_n, as the variable with the lowest order according to (17), is the only one included in the resulting univariate polynomial. Therefore, z_n has to be identified with a system variable that we want to appear in the polynomial. For that reason, we choose the ordering

$$\dot{x}_j(t)>\ldots>\dot{x}_j(t-\tau_j)>\ldots>\dot{x}_k(t)>\ldots>\dot{x}_k(t-\tau_k)>y_i^{(s_i+1)}(t-\tau) \tag{18}$$

where $j,k=1,\dots,n$, for the elimination process of the set of equations (12).

And similarly, we choose the ordering

$$x_j(t)>\ldots>x_j(t-\tau_{r,j})>\ldots>x_k(t)>\ldots>x_k(t-\tau_{r,k})>y_i^{(s_i)}(t-\tau_i) \tag{19}$$

where $j,k=1,\dots,n$, and $r=1,\dots,i$, to carry out the elimination process for the set of equations (15). Then the resulting univariate polynomial represents the input-output representation (10) we looked for.

For additional technicalities about Gröbner bases the reader is referred for instance to [2, 3, 6] .

Example 4.2. Consider the system

$$\begin{aligned}\dot{x}_1 &= \dot{x}_1(t-1)x_2\\ \dot{x}_2 &= \dot{x}_2(t-1)u_1\\ \dot{x}_3 &= \dot{x}_1(t-1)+u_2\\ y_1 &= x_1\\ y_2 &= x_3\end{aligned}$$

Step 0 First, we set

$$h_1^0 := y_1 - x_1 \ ,$$

Step 1 Its time derivative is

$$\dot{h}_1^0 := \dot{y}_1 - \dot{x}_1.$$

According to (11) we constitute the following set of equations of the form (12)

$$\begin{aligned}\dot{x}_1 - \dot{x}_1(t-1)x_2 &= 0\\ \dot{y}_1 - \dot{x}_1 &= 0\\ \dot{y}_1(t-1) - \dot{x}_1(t-1) &= 0.\end{aligned}$$

To eliminate $\dot{x}_1$ and $\dot{x}_1(t-1)$ we apply the lexicographical ordering $\dot{x}_1 > \dot{x}_1(t-1) > \dot{y}_1(t-1)$, and employing the Gröbner bases technique we get the polynomial[4]

$$x_2\dot{y}_1(t-1) - \dot{y}_1$$

which we denote by

$$h_1^1 := x_2\dot{y}_1(t-1) - \dot{y}_1.$$

Step 2 The condition (13) is not yet satisfied.

Step 1 We compute

$$\dot{h}_1^1 := \dot{x}_2\dot{y}_1(t-1) + x_2\ddot{y}_1(t-1) - \ddot{y}_1.$$

Again, we constitute the set of equations according to (11) and (12)

$$\begin{aligned}\dot{x}_2 - \dot{x}_2(t-1)u_1 &= 0\\ \dot{x}_2\dot{y}_1(t-1) + x_2\ddot{y}_1(t-1) - \ddot{y}_1 &= 0\\ \dot{x}_2(t-1)\dot{y}_1(t-2) + x_2(t-1)\ddot{y}_1(t-2) - \ddot{y}_1(t-1) &= 0.\end{aligned}$$

[4] In the examples the Maple computer algebra system has been used to carry out the computations. First, one needs to load the Gröbner base package by calling the command *with(Groebner)*. Then for instance for the list of polynomials $P := [\dot{x}_1 - \dot{x}_1(t-1)x_2, \dot{y}_1 - \dot{x}_1, \dot{y}_1(t-1) - \dot{x}_1(t-1)]$ and the list of ordered variables $O := [\dot{x}_1, \dot{x}_1(t-1), \dot{y}_1(t-1)]$ the command *Solve(P,O)* returns the Gröbner basis of which the first element is the polynomial, $x_2\dot{y}_1(t-1) - \dot{y}_1$, we looked for.

After application of the lexicographical ordering $\dot{x}_2 > \dot{x}_2(t-1) > \ddot{y}_1(t-2)$ we, this time, get by employing the Gröbner bases technique the polynomial

$$\dot{y}_1(t-1)u_1x_2(t-1)\ddot{y}_1(t-2)-\dot{y}_1(t-1)u_1\ddot{y}_1(t-1)-x_2\ddot{y}_1(t-1)\dot{y}_1(t-2)+\ddot{y}_1\dot{y}_1(t-2)$$

which we denote as

$$h_1^2 := \dot{y}_1(t-1)u_1x_2(t-1)\ddot{y}_1(t-2)-\dot{y}_1(t-1)u_1\ddot{y}_1(t-1)-x_2\ddot{y}_1(t-1)\dot{y}_1(t-2)+\ddot{y}_1\dot{y}_1(t-2).$$

Step 2 Since $i \neq p$ we set $i := 2$.

Step 1 We denote

$$h_2^0 := y_2 - x_3.$$

Then

$$\dot{h}_2^0 := \dot{y}_2 - \dot{x}_3$$

and by (11) and (12) we constitute the set of equations

$$\begin{aligned}
\dot{y}_2 - \dot{x}_3 &= 0\\
\dot{x}_3 - \dot{x}_1(t-1) - u_2 &= 0\\
\dot{x}_1 - \dot{x}_1(t-1)x_2 &= 0\\
\dot{y}_2(t-1) - \dot{x}_3(t-1) &= 0\\
\dot{x}_3(t-1) - \dot{x}_1(t-2) - u_2(t-1) &= 0\\
\dot{x}_1(t-1) - \dot{x}_1(t-2)x_2(t-1) &= 0.
\end{aligned}$$

After choosing the lexicographical ordering $\dot{x}_3 > \dot{x}_3(t-1) > \dot{x}_1 > \dot{x}_1(t-1) > \dot{x}_1(t-2) > \dot{y}_2(t-1)$ one gets the resulting polynomial

$$x_2(t-1)\dot{y}_2(t-1) - x_2(t-1)u_2(t-1) - \dot{y}_2 + u_2$$

which we denote by

$$h_2^1 := x_2(t-1)\dot{y}_2(t-1) - x_2(t-1)u_2(t-1) - \dot{y}_2 + u_2.$$

End Finally, we compute the input-output representation for the system. According to (14) and (15) we, for y_1, constitute the set of equations

$$\begin{aligned}
x_2\dot{y}_1(t-1) - \dot{y}_1 &= 0\\
\dot{y}_1(t-1)u_1x_2(t-1)\ddot{y}_1(t-2) - \dot{y}_1(t-1)u_1\ddot{y}_1(t-1)- &\\
-x_2\ddot{y}_1(t-1)\dot{y}_1(t-2) + \ddot{y}_1\dot{y}_1(t-2) &= 0\\
x_2(t-1)\dot{y}_1(t-2) - \dot{y}_1(t-1) &= 0.
\end{aligned}$$

The polynomial resulting from the Gröbner bases technique, employing the lexicographical ordering $x_2 > x_2(t-1) > \ddot{y}_1(t-2)$, yields then the input-output equation

$$\begin{aligned}
&\dot{y}_1^3(t-1)u_1\ddot{y}_1(t-2) - \ddot{y}_1(t-1)\dot{y}_1^2(t-2)\dot{y}_1\\
&\quad -\dot{y}_1^2(t-1)u_1\ddot{y}_1(t-1)\dot{y}_1(t-2) + \dot{y}_1(t-1)\ddot{y}_1\dot{y}_1^2(t-2) = 0. \qquad (20)
\end{aligned}$$

For y_2 we can constitute the set of equations according to (14) and (15) as

$$\begin{aligned} x_2(t-1)\dot{y}_2(t-1) - x_2(t-1)u_2(t-1) - \dot{y}_2 + u_2 &= 0 \\ x_2(t-1)\dot{y}_1(t-2) - \dot{y}_1(t-1) &= 0. \end{aligned}$$

After choosing the lexicographical ordering $x_2(t-1) > \dot{y}_2(t-1)$ one can compute the input-output equation

$$-\dot{y}_1(t-1)u_2(t-1) + \dot{y}_2(t-1)\dot{y}_1(t-1) - \dot{y}_1(t-2)\dot{y}_2 + \dot{y}_1(t-2)u_2 = 0. \quad (21)$$

Thus, the input-output representation for the system is given by (20) and (21).

4.1. Computational complexity

Gröbner bases are implemented in many modern computer algebra systems, as for instance Maple or Mathematica, which enable to solve effectively reasonable sized systems of polynomial equations. In our case, the complexity of the computations is affected by the number of the equations (12) and (15), respectively. Note that the number of these equations is determined by the polynomials in (11) and (14), respectively. To constitute the set (12) one needs to consider a corresponding equation for every (nonzero) summand in every (nonzero) polynomial $p(\delta]$ and $p_j(\delta]$ in (11). In general, the presence of the respective polynomial corresponds to the presence of the respective row in (12), and the presence of the respective summands in that polynomial corresponds to the time-delays present int that row. Similarly, for the number of equations in (15) the polynomials $b_i(\delta]$ and $b_{r,j}(\delta]$ in (14) play the crucial role.

Besides the number of equations, the computational complexity is affected also by the choice of the monomial ordering (18) and (19). Note that the lexicographical ordering we employed is only one choice among the possible orderings one can consider here.

5. CONCLUSIONS

In this paper the state elimination procedure has been extended further to the class of nonlinear neutral state space systems. It was shown that for such a system there always exists, at least locally, a set of input-output differential-delay equations which we can think of as an external, or input-output, representation of the system. In general, such an input-output representation is neutral again. In addition, it might possess more delays than the original state-space system (see e. g. Example 3.1). The analysis and synthesis by using such a representation can then be more complicated.

In case the systems under consideration are polynomial or rational, one can easily employ the Gröbner bases technique to eliminate the necessary states. Note that the Gröbner bases package is implemented practically in every computer algebra system. In this paper, the computations in the examples were carried out by the computer algebra system Maple, for besides the Gröbner bases package it also has Ore tools package available, useful for handling non-commutative polynomials. Naturally, the Gröbner bases technique of the state elimination, as presented in this paper, is applicable also to nonlinear retarded systems and nonlinear systems without delays, as these form subclasses of neutral systems.

ACKNOWLEDGEMENT

This work was supported by the Slovak Grant Agency grant no. VEGA 1/0276/14, Application of Algebraic Methods to Nonlinear Control Systems.

(Received May 3, 2013)

REFERENCES

[1] M. Anguelova and B. Wennberg: State elimination and identifiability of the delay parameter for nonlinear time-delay systems. Automatica *44* (2008), 1373–1378.

[2] T. Becker and V. Weispfenning: Gröbner Bases. Springer-Verlag, New York 1993.

[3] B. Buchberger and F. Winkler: Gröbner Bases and Applications. Cambridge University Press, Cambridge 1998.

[4] P. M. Cohn: Free Rings and Their Relations. Academic Press, London 1985.

[5] G. Conte, C. H. Moog, and A. M. Perdon: Algebraic Methods for Nonlinear Control Systems. Theory and Applications. Second edition. Communications and Control Engineering. Springer-Verlag, London 2007.

[6] D. Cox, J. Little, and D. O'Shea: Ideals, Varieties, and Algorithms. Springer-Verlag, New York 2007.

[7] S. Diop: Elimination in control theory. Math. Contr. Signals Syst. *4* (1991), 72–86.

[8] S. T. Glad: Nonlinear regulators and Ritt's remainder algorithm. In: Analysis of Controlled Dynamical Systems (B. Bournard, B. Bride, J. P. Gauthier, and I. Kupka, eds.), Progress in systems and control theory *8*, Birkhäuser, Boston 1991, pp. 224–232

[9] A. Glumineau, C. H. Moog, and F. Plestan: New algebro-geometric conditions for the linearization by input-output injection. IEEE Trans. Automat. Control *41* (1996), 598–603.

[10] M. Halás: An algebraic framework generalizing the concept of transfer functions to nonlinear systems. Automatica *44* (2008), 1181–1190.

[11] M. Halás: Nonlinear time-delay systems: a polynomial approach using Ore algebras. In: Topics in Time-Delay Systems: Analysis, Algorithms and Control (J. J. Loiseau, W. Michiels, S. Niculescu, and R. Sipahi, eds.), Lecture Notes in Control and Information Sciences, Springer, 2009.

[12] M. Halás: Computing an input-output representation of a neutral state-space system. In: IFAC Workshop on Time Delay Systems, Grenoble 2013.

[13] M. Halás and M. Anguelova: When retarded nonlinear time-delay systems admit an input-output representation of neutral type. Automatica *49* (2013) 561–567.

[14] M. Halás and Ü. Kotta: A transfer function approach to the realisation problem of nonlinear systems. Internat. J. Control *85* (2012), 320–331.

[15] M. Halás, Ü. Kotta, and C. H. Moog: Transfer function approach to the model matching problem of nonlinear systems. In: 17th IFAC World Congress, Seoul 2008.

[16] M. Halás and C. H. Moog: A polynomial solution to the model matching problem of nonlinear time-delay systems. In: European Control Conference, Budapest 2009.

[17] M. Huba: Comparing 2DOF PI and predictive disturbance observer based filtered PI control. J. Process Control *23* (2013), 1379–1400.

[18] Ü. Kotta, Z. Bartosiewicz, E. Pawluszewicz, and M. Wyrwas: Irreducibility, reduction and transfer equivalence of nonlinear input-output equations on homogeneous time scales. Syst. Control Lett. *58* (2009), 646–651.

[19] Ü. Kotta, P. Kotta, and M. Halás: Reduction and transfer equivalence of nonlinear control systems: unification and extension via pseudo-linear algebra. Kybernetika *46* (2010), 831–849.

[20] L. A. Márquez-Martínez, C. H. Moog, and M. Velasco-Villa: The structure of nonlinear time-delay systems. Kybernetika *36* (2000), 53–62.

[21] L. A. Márquez-Martínez, C. H. Moog, and M. Velasco-Villa: Observability and observers for nonlinear systems with time delays. Kybernetika *38* (2002), 445–456.

[22] T. Ohtsuka: Model structure simplification of nonlinear systems via immersion. IEEE Trans. Automat. Control *50* (2005), 607–618.

[23] O. Ore: Linear equations in non-commutative fields. Ann. Math. *32* (1931), 463–477.

[24] O. Ore: Theory of non-commutative polynomials. Ann. Math. *34*(1933), 480–508.

[25] P. Picard, J. F. Lafay, and V. Kučera: Model matching for linear systems with delays and 2D systems. Automatica *34* (1998), 183–191.

[26] J. Rudolph: Viewing input-output system equivalence from differential algebra. J. Math. Systems Estim. Control *4* (1994), 353–383.

[27] U. Walther, T. T. Georgiou, and A. Tannenbaum: On the computation of switching surfaces in optimal control: a Gröbner basis approach. IEEE Trans. Automat. Control *46* (2001), 534–540.

[28] X. Xia, L. A. Márquez-Martínez, P. Zagalak, and C. H. Moog: Analysis of nonlinear time-delay systems using modules over non-commutative rings. Automatica *38* (2002), 1549–1555.

[29] J. Zhang, X. Xia, and C. H. Moog: Parameter identifiability of nonlinear systems with time-delay. IEEE Trans. Automat. Control *51* (2006), 371–375.

[30] Y. Zheng, J. Willems, and C. Zhang: A polynomial approach to nonlinear system controllability. IEEE Trans. Automat. Control *46* (2001), 1782–1788.

Miroslav Halás, Institute of Robotics and Cybernetics, Fac. of Electrical Engineering and IT, Slovak University of Technology, Ilkovičova 3, 812 19 Bratislava. Slovak Republic.

e-mail: miroslav.halas@stuba.sk

Pavol Bisták, Institute of Automotive Mechatronics, Fac. of Electrical Engineering and IT, Slovak University of Technology, Ilkovičova 3, 812 19 Bratislava. Slovak Republic.

e-mail: pavol.bistak@stuba.sk