

# Practical polynomial formulas in MIMO nonlinear realization problem

J. Belikov, P. Kotta, Ü. Kotta and M. Tönso

**Abstract**—The explicit and simple polynomial formulas are given for computation of the (differentials) of state coordinates from the set of nonlinear input-output (i/o) equations under assumption that the set of i/o equations is realizable in the state-space form. The paper addresses a very wide class of nonlinear control systems, that accommodates continuous- as well as two discrete-time cases, based either on the shift or difference operator. Two types of polynomial formulas are suggested, based either on polynomial left division or on the application of the concept of adjoint polynomials and implemented in Mathematica-based software.

**Index Terms**—nonlinear control system, input-output model, polynomial method, state-space realization, pseudo-linear algebra

## I. INTRODUCTION

This paper has to be understood as continuation of research in [12] in two aspects. First, the algebraic tools we employ are the same as those in [12]. This unifying formalism, called pseudo-linear algebra [5], allows to describe and study continuous- and discrete-time nonlinear control systems as a single object using the so-called pseudo-linear operator instead of conventional time derivative, difference or forward shift operators. The main advantage of the pseudo-linear approach is that it allows to optimize the code in implementation of the algorithms in symbolic software. The widespread use of computers nowadays promotes the idea of ‘computability’ to be an important criterion for evaluation of theoretical tools since the ultimate goal is, in general, a computer program. The advantage of software tools is clear: they make the sophisticated mathematical tools available to practitioners.

Second, the former paper [12] unifies the results on system reduction whereas this paper focuses on the closely related problem of state-space realization. Namely, the realization procedure ends up with the accessible (controllable) realization iff the set of input-output (i/o) equations is reduced to its simplest form, being transfer equivalent to the original set of equations. We will give, for realizable i/o equations, the *explicit polynomial formulas* that allow to compute the differentials of the state coordinates directly from the ‘tangent linearized system equations’ expressed in terms of two polynomial matrices. Moreover, the polynomial formulas also simplify computation of the related 1-forms that play a key role in the realizability conditions. The realization

problem for nonlinear MIMO systems has been addressed in many papers, but the polynomial method has several advantages if compared to the earlier algorithm-related methods, based either on the sequences of the subspaces of differential 1-forms [8], on the sequence of distributions of vector fields as in [16], or on the iterative Lie brackets of the vector fields as in [9]. The most powerful arguments are explicit formulas and computation speed, but also the program code is shorter and more compact.

Note that the results of this paper may be also understood as the generalization of the results in [13] and [4] where the results for shift-operator-based discrete-time and continuous-time cases are presented. However, the results for difference-operator-based case are new. The alternative formalism for unification is time scale calculus [6], but the latter does not accommodate the case of shift operator.

The paper is organized as follows. Section II recalls the definitions / formulas of pseudo-linear algebra to be applied later, and provides the unified system description in terms of pseudo-linear operator, construction of the sigma-differential field, associated with the control system and the related vector space of differential 1-forms that corresponds to the ‘tangent linearized system description’. In Section III the sequence of the vector spaces is defined in terms of which the realizability conditions are formulated. In Section IV the polynomial formulas for computation of the state coordinate differentials are presented. Section V gives few examples and discussion about Mathematica implementation. Last Section concludes the paper.

## II. PSEUDO-LINEAR ALGEBRA AND CONTROL SYSTEMS

**Definition 1:** Let  $\mathcal{K}$  be a field and  $\sigma : \mathcal{K} \rightarrow \mathcal{K}$  an automorphism of  $\mathcal{K}$ . A map  $\delta : \mathcal{K} \rightarrow \mathcal{K}$  which satisfies

$$\begin{aligned}\delta(a+b) &= \delta(a) + \delta(b) \\ \delta(ab) &= \sigma(a)\delta(b) + \delta(a)b\end{aligned}\tag{1}$$

is called a pseudo- (with respect to  $\sigma$ ) or  $\sigma$ -derivation.

Note that  $\sigma(a/b) = \sigma(a)/\sigma(b)$  and  $\delta(a/b) = (b\delta(a) - a\delta(b))/(\sigma(b)b)$  for  $a, b \in \mathcal{K}$  with  $b \neq 0$ . Obviously the derivative is  $\sigma$ -derivation, with  $\sigma(a) = a$ . Also a difference operator,  $\Delta = \sigma - 1_{\mathcal{K}}$ , is a pseudo-derivation. Note that (1) is again satisfied, since  $\Delta(ab) = \sigma(a)\Delta(b) + \Delta(a)b = \sigma(a)(\sigma(b) - b) + (\sigma(a) - a)b = \sigma(ab) - ab$ .

**Definition 2:** A  $\sigma$ -differential field is a triple  $(\mathcal{K}, \sigma, \delta)$  where  $\mathcal{K}$  is a field,  $\sigma$  is an automorphism of  $\mathcal{K}$  and  $\delta$  is a  $\sigma$ -derivation.

**Definition 3:** Let  $\mathcal{E}$  be a vector space over  $\mathcal{K}$ . A map  $\theta : \mathcal{E} \rightarrow \mathcal{E}$  is called  $\mathcal{K}$ -pseudo-linear (with respect to  $\sigma$  and  $\delta$ )

The work of was supported by the European Union through the European Regional Development Fund, the target funding project SF0140018s08 of Estonian Ministry of Education and Research, and by the ESF grants N8787, 8365.

J. Belikov, P. Kotta, Ü. Kotta and M. Tönso are with Institute of Cybernetics at TUT, Akadeemia tee 21, 12618 Tallinn, Estonia

if

$$\begin{aligned}\theta(u+v) &= \theta(u) + \theta(v) \\ \theta(au) &= \sigma(a)\theta(u) + \delta(a)u\end{aligned}\quad (2)$$

for any  $a \in \mathcal{K}$ ,  $u, v \in \mathcal{E}$ .

Note that any field  $\mathcal{K}$  is a vector space itself. Hence, (2) holds for an  $a, u, v \in \mathcal{K}$  and  $\delta : \mathcal{K} \rightarrow \mathcal{K}$  is a pseudo-linear map, simply by letting  $\theta = \delta$ . For  $\theta = \sigma$  and  $\delta = 0$ , (2) is clearly satisfied. Thus, pseudo-linear maps address differential, shift and difference operators from a unified viewpoint.

This paper addresses a wide class of nonlinear control systems. For the state  $x(t)$ , the control  $u(t)$ , and the output  $y(t)$  we write just  $x$ ,  $u$  and  $y$ , respectively. In what follows, the symbol  $x^{(1)}$  stands for a pseudo-linear operator, applied to  $x$ :  $x^{(1)} = \theta(x)$ . It can be a derivation,  $x^{(1)} = \dot{x}$ , that corresponds to the continuous-time case, a forward shift  $x^{(1)} = \sigma(x)$ , that sends  $x(t)$  to  $x(t+1)$ , or a difference,  $x^{(1)} = \alpha(\sigma(x) - x)$  with  $\alpha \in \mathbb{R}$ , that correspond to two alternative discrete-time cases. Moreover,  $x^{(k)} := \theta^k(x)$ , the  $k$ -fold application of operator  $\theta$  to  $x$ . In a similar manner  $\sigma^k(x)$  denotes  $k$  times application of  $\sigma$  to  $x$ . The nonlinear control systems considered in this paper are defined either by the state equations

$$x^{(1)} = f(x, u), \quad y = h(x) \quad (3)$$

or by the set of i/o equations

$$\begin{aligned}y_i^{(n_i)} &= \phi_i(y_j, \dots, y_j^{(n_{ij}-1)}, u_k, \dots, u_k^{(s_{ik})}), \\ j &= 1, \dots, p, \quad k = 1, \dots, m, \quad i = 1, \dots, p,\end{aligned}\quad (4)$$

where  $s_{ik} < n_i$  for  $i = 1, \dots, p$ ,  $k = 1, \dots, m$  and  $n_{ij} < \min(n_i, n_j)$ . The latter condition means that system is in a canonical form<sup>1</sup>. Define  $s := \max_{i,k} s_{ik}$  and  $n_1 + \dots + n_p =: n$ . In (3) and (4),  $x \in \mathbb{R}^n$ ,  $u = (u_1, \dots, u_m)^T \in \mathbb{R}^m$ ,  $y = (y_1, \dots, y_p)^T \in \mathbb{R}^p$ , and the entries of the  $f$ ,  $h$  and  $\phi_i$  are meromorphic functions. Moreover, we assume that system (4) is generically *submersive*, i.e.

$$\text{rank} \frac{\partial [\sigma^{n_1}(y_1), \dots, \sigma^{n_p}(y_p)]^T}{\partial (y, u)} = p \quad (5)$$

holds everywhere possibly except on a set of measure zero. Assumption (5) reduces to the well-known condition in the case of discrete-time nonlinear systems [13] and is trivially satisfied in the case of continuous-time nonlinear systems when  $\sigma(y) = y$ .

Let  $\text{dg}^{(k)}$  denote  $(\text{dg}, \dots, \text{dg}^{(k)})$ . It is convenient to associate with the set of i/o equations (4) its extended state-space system  $z^{(1)} = f_e(z, v)$  with the input  $v = u^{(s+1)}$ , the state  $z = (y_j^{(n_j-1)}, j = 1, \dots, p, u_k^{(s)}, k = 1, \dots, m)^T$ , and

<sup>1</sup>The form (4) is an extension of the Guidorzi canonical form, introduced in [3] for linear systems and typically applied in system identification. The other forms, like Hermite or Popov form, may be also used. Assumption  $n_{ij} < \min(n_i, n_j)$  implies that whenever (4) admits a state-space realization, the indices  $n_i$  associated to each output  $y_i$ ,  $i = 1, \dots, p$ , are the observability indices of any observable state-space realization of order  $n$ .

a state transition map  $f_e(z, v)$  defined by

$$\begin{aligned}z_1^{(1)} &= z_2 \\ &\vdots \\ z_{n_1}^{(1)} &= \phi_1(z) \\ &\vdots \\ z_{n_1+\dots+n_{p-1}+1}^{(1)} &= z_{n_1+\dots+n_{p-1}+2} \\ &\vdots \\ z_{n_1+\dots+n_p}^{(1)} &= \phi_p(z) \\ z_{n+(j-1)(s+1)+k}^{(1)} &= z_{n+(j-1)(s+1)+k+1} \\ z_{n+j(s+1)}^{(1)} &= v_j\end{aligned}\quad (6)$$

for  $j = 1, \dots, m$ ,  $k = 1, \dots, s$ . Note that (6) is not claimed to be a realization of (4); however, the sets of solutions  $\{u(t), y(t)\}$  of (4) and (6) are equal. System (6) allows to construct the inversive  $\sigma$ -differential field  $\mathcal{K}^*$  for equations (4) in a similar manner as for state equations (3), following the lines in [8], [11] for the continuous- and discrete-time counterparts, respectively. Associate with system (4) the field  $\mathcal{K}$  of meromorphic functions of the independent system variables  $\{y_i^{(n_i-1)}, 1 \leq i \leq p, u_r^{(k)}, 1 \leq r \leq m, k \geq 0\}$  and let  $\delta$  be a  $\sigma$ -derivation, defined on  $\mathcal{K}$ . We define a pseudo-linear operator  $\theta$ , acting on  $\mathcal{K}$ , determined by system equations (6) (or (4), separately for derivation, shift and difference operators.

First, if  $\sigma = \text{id}_{\mathcal{K}}$  and  $\delta = d/dt$ , a pseudo-linear operator  $\theta = \delta$  and

$$\delta \varphi(y_j^{(r)}, u_k^{(\nu)}) = \frac{\partial \varphi}{\partial y_j^{(r)}} \delta y_j^{(r)} + \frac{\partial \varphi}{\partial u_k^{(\nu)}} \delta u_k^{(\nu)}, \quad (7)$$

where  $\delta y_j^{(r)} = y_j^{(r+1)}$ ,  $r = 0, \dots, n_i - 2$ ,  $\delta y_i^{(n_i-1)} = \phi_i(\cdot)$  and  $\delta u_k^{(\nu)} = u_k^{(\nu+1)}$ . Second, if  $\delta = 0$ , a pseudo-linear operator  $\theta = \sigma$  and

$$\sigma \varphi(y_i^{(r)}, u_k^{(\nu)}) = \varphi(\sigma y_i^{(r)}, \sigma u_k^{(\nu)}), \quad (8)$$

where  $\sigma y_i^{(r)} = y_i^{(r+1)}$ ,  $r = 0, \dots, n_i - 2$ ,  $\sigma y_i^{(n_i-1)} = \phi_i(\cdot)$  and  $\sigma u_k^{(\nu)} = u_k^{(\nu+1)}$ . Finally, if  $\delta = \alpha(\sigma - \text{id}_{\mathcal{K}}) := \Delta$  with  $\alpha \in \mathbb{R}$ , then a pseudo-linear operator  $\theta = \Delta$  and

$$\Delta \varphi(y_i^{(r)}, u_k^{(\nu)}) = \alpha \left[ \varphi(\sigma y_i^{(r)}, \sigma u_k^{(\nu)}) - \varphi(y_i^{(r)}, u_k^{(\nu)}) \right], \quad (9)$$

where  $\sigma = \Delta/\alpha + \text{id}_{\mathcal{K}}$ ,  $\Delta y_i^{(r)} = \Delta y_i^{(r+1)}$ ,  $r = 0, \dots, n_i - 2$ ,  $\Delta y_i^{(n_i-1)} = \phi_i(\cdot)$  and  $\Delta u_k^{(\nu)} = u_k^{(\nu+1)}$ .

Under submersivity assumption,  $\sigma$  is an automorphism of  $\mathcal{K}$  and there exists, up to an isomorphism, a unique field extension  $\mathcal{K}^*$  called the inversive closure of  $\mathcal{K}$  [7]. Here we assume  $\mathcal{K}^*$  to be given and by slight abuse of notation use the same symbol  $\mathcal{K}$  for both. For explicit construction of  $\mathcal{K}^*$ , see [11] and [2] for the cases of shift and difference operators, respectively. Note that in the continuous-time case when  $\sigma = \text{id}_{\mathcal{K}}$ ,  $\mathcal{K}^* = \mathcal{K}$ .

Next, define the vector space of 1-forms  $\mathcal{E} = \text{span}_{\mathcal{K}}\{d\xi; \xi \in \mathcal{K}\}$ . Any element  $v \in \mathcal{E}$  is a vector of the form  $v = \sum_{i=1}^s c_i d\xi_i$  where all  $c_i \in \mathcal{K}$ . We say

that  $v \in \mathcal{E}$  is exact if  $v = d\varphi$  for some  $\varphi \in \mathcal{K}$ . The vector space  $\mathcal{E}$  can also be endowed with the  $\sigma$ -differential structure. Each  $\theta : \mathcal{K} \rightarrow \mathcal{K}$  induces  $\theta : \mathcal{E} \rightarrow \mathcal{E}$  as follows  $\theta(v) = v^{(1)} = \sum_i [\sigma(c_i)d(\theta(\xi_i)) + \delta(c_i)d\xi_i]$ . The operator  $\theta$  commutes with operator  $d$ ,  $\theta(d\varphi) = d(\theta(\varphi))$ , and reduces to the well-known rules  $\delta v = \sum_i [c_i d(\delta\xi_i) + \delta(c_i)d\xi_i]$  and  $\sigma v = \sum_i \sigma(c_i)d(\sigma\xi_i)$ , for the special cases of continuous-time systems ( $\sigma = \text{id}_{\mathcal{K}}$ ,  $\theta = \delta = d/dt$ ) and discrete-time systems ( $\delta = 0$ ,  $\theta = \sigma$ ), respectively.

### III. REALIZATION

The *relative degree*  $r$  of a 1-form  $\omega \in \mathcal{E}$  with respect to the control variable  $v$  of the extended system (6) is defined to be the least integer such that  $\theta^r(\omega) \notin \text{span}_{\mathcal{K}}\{dz\}$ . If such an integer does not exist,  $r := \infty$ . The relative degree of a meromorphic function  $\varphi(z, v)$  is defined as the relative degree of  $d\varphi(z, v)$ . The non-increasing sequence of subspaces  $\{\mathcal{H}_k\}$  of  $\mathcal{E}$  defined by

$$\begin{aligned} \mathcal{H}_0 &= \text{span}_{\mathcal{K}}\{dz, dv\} \\ &= \text{span}_{\mathcal{K}}\left\{d\mathbf{y}_j^{(n_j-1)}, j=1, \dots, p, \right. \\ &\quad \left. d\mathbf{u}_v^{(s+1)}, v=1, \dots, m\right\} \\ \mathcal{H}_k &= \{\omega \in \mathcal{H}_{k-1} \mid \theta(\omega) \in \mathcal{H}_{k-1}\}, \quad k \geq 1, \end{aligned} \quad (10)$$

plays the key role in the study of realizability problem. There exists an integer  $k^* \leq n + m(s+1) = \dim_{\mathcal{K}} \mathcal{H}_1$  such that  $\mathcal{H}_0 \supset \dots \supset \mathcal{H}_{k^*} \supset \mathcal{H}_{k^*+1} = \mathcal{H}_{k^*+2} = \dots =: \mathcal{H}_{\infty}$ . Note that  $\mathcal{H}_k$  contains the 1-forms with relative degree greater than or equal to  $k$ , and  $\mathcal{H}_{\infty}$  is the largest subspace of  $\mathcal{H}_0$ , invariant under application of operator  $\theta$ . Moreover, since the relative degree is invariant under the (extended) state diffeomorphism, so are the subspaces  $\mathcal{H}_k$ . One says that  $\mathcal{V} \subset \mathcal{E}$  is completely integrable, if  $\mathcal{V}$  can be, at least locally, generated by exact 1-forms,  $\mathcal{V} = \text{span}_{\mathcal{K}}\{d\psi_1, \dots, d\psi_r\}$ .

**Definition 4:** System (3) is generically single-experiment observable if

$$\text{rank}_{\mathcal{K}} \frac{\partial [h^T(x), \dots, (h^{(n-1)})^T(x, \mathbf{u}^{(n-2)})]^T}{\partial x} = n.$$

If the input  $u$  applied to (3) (for any initial state  $\mathbf{x}_0$ ) generates an output  $y$  such that  $u$  and  $y$  satisfy equation (4), the set of i/o equations (4) is called *realizable* and (3) is called a *realization* of (4).

**Theorem 1:** The set of i/o equations (4) has an observable state-space realization iff the subspace  $\mathcal{H}_{s+2}$ , defined by (10), is completely integrable. Moreover, the state coordinates can be found by integrating the basis vectors of  $\mathcal{H}_{s+2}$ .

**PROOF.** (Sufficiency). Assume that  $\mathcal{H}_{s+2}$  is completely integrable, with basis  $\{d\xi_1, \dots, d\xi_n\}$ . Note that  $d\xi_1, \dots, d\xi_n$  are 1-forms with relative degree equal to or greater than  $s+2$ . From the structure of (6), the relative degree of  $du_k$ , for all  $k = 1, \dots, m$ , is equal to  $s+1$ ; therefore  $du_k \in \mathcal{H}_{s+1}$ . However, since the control variables are independent system variables,  $du_k \in \mathcal{H}_{s+2}$  would yield a contradiction,  $du_k = \sum_{i=1}^n \alpha_{ik} d\xi_i$ . Hence  $\mathcal{H}_{s+1} = \mathcal{H}_{s+2} \oplus \text{span}_{\mathcal{K}}\{du\}$ . In a similar manner it can be proven that  $\mathcal{H}_i = \mathcal{H}_{s+2} \oplus \text{span}_{\mathcal{K}}\{d\mathbf{u}^{(s-i+1)}\}$ , for  $i = 1, \dots, s$ . Introduce a coordinate

transformation in the extended state space, defined by

$$x_i = \xi_i(z), \quad x_{n+(j-1)(s+1)+k} = z_{n+(j-1)(s+1)+k} = u_k^{(j-1)}, \quad (11)$$

for  $i = 1, \dots, n$  and  $j = 1, \dots, s+1$ ,  $k = 1, \dots, m$ . Since  $\mathcal{H}_k$ , for  $k \geq 0$ , are invariant under the (extended) state diffeomorphism, from the definition of  $\mathcal{H}_k$ ,  $\theta(dx_i) = \theta(d\xi_i(z)) \in \mathcal{H}_{s+2} \oplus \text{span}_{\mathcal{K}}\{du\}$  and hence  $\theta(dx_i) = \sum_{j=1}^n \beta_{ij} dx_j + \sum_{k=1}^m \gamma_{ik} du_k$ , for  $i = 1, \dots, n$ . Thus the extended system, in the new coordinates, has the form

$$\begin{aligned} x_i^{(1)} &= f_i(x, u), \quad i = 1, \dots, n, \\ x_{n+(j-1)(s+1)+k}^{(1)} &= x_{n+(j-1)(s+1)+k+1}, \\ &\quad j = 1, \dots, s, \quad k = 1, \dots, m \\ x_{n+j(s+1)}^{(1)} &= v_j, \quad j = 1, \dots, s \\ y_i &= x_{n_1+\dots+n_{i-1}+1}, \quad i = 1, \dots, p \end{aligned} \quad (12)$$

for some functions  $f_i$ . The last  $(s+1)m$  equations can be omitted, since, due to the second set of equalities in (11), they are identities. The i/o map of the extended system in the new coordinates remains the same, therefore the sets of solutions of (12) and (6) (or alternatively, (4)) are the same. The proof can be concluded by observing that by construction,  $x_i = \xi_i(\mathbf{y}^{(n-1)}, \mathbf{u}^{(s)})$ , for  $i = 1, \dots, n$ , the state-space representation is observable.

(Necessity). Suppose, now, that the equation (4) has an observable state-space realization (3). By Definition 4, the set of equations  $y = h(x)$ ,  $y^{(i)} = h_i(x, u, \dots, u^{(i-1)})$  for  $i = 1, \dots, n-1$  can be solved (generically) for  $x$ , yielding  $x = \xi(\mathbf{y}^{(n-1)}, \mathbf{u}^{(n-2)})$ . Substituting this quantity into  $y^{(n)} = h_n(x, \mathbf{u}^{(n-1)})$ , we get the i/o equation  $y^{(n)} = h_n[\xi(\mathbf{y}^{(n-1)}, \mathbf{u}^{(n-2)}), \mathbf{u}^{(n-1)}]$  with realization (3). The subspace  $\mathcal{H}_{n+1} = \text{span}_{\mathcal{K}}\{d\xi_1, \dots, d\xi_n\}$  (for  $s = n-1$ ) is completely integrable. ■

Given the subspace of 1-forms, its integrability can be checked by the Frobenius theorem, where by  $d\omega_k$  is denoted the exterior derivative of the 1-form and by  $\wedge$  the wedge product of the differential forms.

**Theorem 2:** (Frobenius) [8]. The subspace  $\mathcal{V} = \text{span}_{\mathcal{K}}\{\omega_1, \dots, \omega_r\} \subset \mathcal{E}$  is locally exact if and only if  $d\omega_k \wedge \omega_1 \wedge \dots \wedge \omega_r = 0$ , for all  $k = 1, \dots, r$ .

The sufficiency part of the proof of Theorem 1 provides the realization algorithm: find the exact basis vectors for  $\mathcal{H}_{s+2}$  (which is possible since  $\mathcal{H}_{s+2}$  is completely integrable) and integrate them to get the state coordinates.

### IV. POLYNOMIAL FORMULAS

#### A. Polynomial system description

Polynomial framework is built upon the framework of 1-forms. A *left skew polynomial* is an expression in the form  $a = \sum_{i=0}^n a_i Z^{n-i}$ ,  $a_i \in \mathcal{K}$  where  $Z$  is a polynomial indeterminate. Note that  $a \neq 0$  iff at least one  $a_i$ ,  $i = 0, \dots, n$  is nonzero. If  $a_0 \neq 0$ , then the positive integer  $n$  is called the *degree* of  $a(Z)$  and denoted by  $d^0(a)$ . In addition, we set  $d^0(0) = -\infty$ . Any automorphism  $\sigma$  and  $\sigma$ -derivation  $\delta$  induce a left skew polynomial ring  $\mathcal{K}[Z; \sigma, \delta]$

over  $\mathcal{K}$ , endowed with the non-commutative multiplication rule [1]

$$Z \cdot a = \sigma(a) \cdot Z + \delta(a), \quad a \in \mathcal{K}. \quad (13)$$

The ring  $\mathcal{K}[Z; \sigma, \delta]$  is an integral domain, i.e. it does not contain any zero divisors [5].

Define

$$Z^\nu dy_j := dy_j^{(\nu)}, \quad Z^r du_k := du_k^{(r)}. \quad (14)$$

Any pseudo-linear map  $\theta : \mathcal{E} \rightarrow \mathcal{E}$  induces a polynomial operator  $r(Z) = \sum_{i=0}^n a_i Z^i$ , acting on  $\mathcal{E}$  for any  $\alpha d\zeta \in \mathcal{E}$ , satisfying the property  $\sum_{i=0}^n a_i Z^i (\alpha d\zeta) = \sum_{i=0}^n a_i (Z^i \cdot \alpha) d\zeta$  with  $a_i, \alpha \in \mathcal{K}$  and  $d\zeta \in \mathcal{E}$ . Multiplication in  $\mathcal{K}[Z; \sigma, \delta]$  corresponds to the composition of operators and  $(r(Z)s(Z))d\zeta = r(Z)(s(Z)d\zeta)$  for any  $r(Z), s(Z) \in \mathcal{K}[Z; \sigma, \delta]$  and  $d\zeta \in \mathcal{E}$ . So the elements of  $\mathcal{K}[Z; \sigma, \delta]$  can be viewed as operators acting from  $\mathcal{E}$  to  $\mathcal{E}$ . It is easy to see that  $Z(\omega) = \theta(\omega)$  for  $\omega \in \mathcal{E}$ . For any differential field  $\mathcal{K}$  with a derivation  $\delta$ ,  $\mathcal{K}[Z; \text{id}_{\mathcal{K}}, \delta]$  is the ring of linear ordinary differential operators. If  $\sigma$  is the automorphism over  $\mathcal{K}$  which takes  $t$  to  $t+1$ , then  $\mathcal{K}[Z; \sigma, 0]$  is the ring of linear shift operators, while  $\mathcal{K}[Z; \sigma, \Delta]$ , where  $\Delta = \alpha(\sigma - 1_{\mathcal{K}})$ ,  $\alpha \in \mathbb{R}$  is the ring of linear difference operators. Moreover, if  $\sigma$  is the automorphism over  $\mathcal{K}$  which takes  $t$  to  $qt$ , then  $\mathcal{K}[Z; \sigma, \Delta]$  with  $\Delta = \sigma - 1_{\mathcal{K}}$  is the ring of linear  $q$ -difference operators, see [1], [5].

The nonlinear system (4) can be represented in terms of two polynomial matrices. For that we apply the differentiation operation to (4) and use (14) to obtain

$$P(Z)dy + Q(Z)du = 0. \quad (15)$$

Here  $P(Z)$  and  $Q(Z)$  are  $p \times p$  and  $p \times m$ -dimensional matrices respectively, whose elements  $p_{ij}, q_{ik} \in \mathcal{K}[Z; \sigma, \delta]$ :

$$p_{ij}(Z) = \delta_{ij} Z^{n_i} - \sum_{\nu=0}^{n_{ij}} \frac{\partial \phi_i}{\partial y_j^{(\nu)}} Z^\nu, \quad q_{ik}(Z) = - \sum_{r=0}^{s_{ik}} \frac{\partial \phi_i}{\partial u_k^{(r)}} Z^r$$

with  $\delta_{ij}$  being Kronecker delta and  $dy = [dy_1, \dots, dy_p]^T$ ,  $du = [du_1, \dots, du_m]^T$ . Further, the notations  $p_i(Z) := [p_{i1}(Z), \dots, p_{ip}(Z)]$  and  $q_i(Z) := [q_{i1}(Z), \dots, q_{im}(Z)]$  are used for row vectors of  $P(Z)$  and  $Q(Z)$ , respectively.

*Example 1:* For the system

$$y_1^{(2)} = y_2 u_1^{(1)} - u_2^{(1)}, \quad y_2^{(2)} = y_1 u_2^{(1)} \quad (16)$$

the matrices  $P(Z)$  and  $Q(Z)$  are

$$P(Z) = \begin{bmatrix} Z^2 & -u_1^{(1)} \\ -u_2^{(1)} & Z^2 \end{bmatrix}, \quad Q(Z) = \begin{bmatrix} -y_2 Z & Z \\ 0 & -y_1 Z \end{bmatrix}. \quad (17)$$

### B. Basis vectors of $\mathcal{H}_{s+2}$

The formulas presented underneath for computing the basis of  $\mathcal{H}_{s+2}$  rely on the *left division* operation. The left division can be performed in  $\mathcal{K}[Z; \sigma, \delta]$  since  $\sigma$  is an automorphism on  $\mathcal{K}$ . If  $p_1$  and  $p_2$  are the polynomials from  $\mathcal{K}[Z; \sigma, \delta]$  with  $d^0(p_1) > d^0(p_2)$ , then there exist a unique *left quotient* polynomial  $\gamma$  and a unique *left remainder* polynomial  $\rho$  such that  $p_1 = p_2 \gamma + \rho$  and  $d^0(\rho) < d^0(p_2)$ .

The left quotient and the left remainder polynomials can be computed by *left Euclidean division algorithm* [5]. We introduce the 1-forms in terms of which the result will be formulated. Let

$$\omega_{i,l} = [p_{i,l}(Z) \quad q_{i,l}(Z)] \begin{bmatrix} dy \\ du \end{bmatrix} \quad (18)$$

for  $i = 1, \dots, p$ ,  $l = 1, \dots, n_i$ , where  $p_{i,l}(Z)$  and  $q_{i,l}(Z)$  are skew polynomials, recursively calculated from the equalities

$$\begin{aligned} p_{i,l-1}(Z) &= Z \cdot p_{i,l}(Z) + \xi_{i,l}, \quad d^0 \xi_{i,l} = 0, \\ q_{i,l-1}(Z) &= Z \cdot q_{i,l}(Z) + \gamma_{i,l}, \quad d^0 \gamma_{i,l} = 0, \end{aligned} \quad (19)$$

with initializations  $p_{i,0}(Z) := p_i(Z)$ ,  $q_{i,0}(Z) := q_i(Z)$ .

*Example 2:* (Continuation of Example 1) Let us find the one-forms  $\omega_{i,l}$  for  $P(Z)$  and  $Q(Z)$  given by (17). For that it is necessary to fix the operators  $\delta$  and  $\sigma$ , for instance, let  $\delta = d/dt$  and  $\sigma = \text{id}_{\mathcal{K}}$ . Then, according to (19),

$$\begin{aligned} [p_{1,0}, q_{1,0}] &= [Z^2, -u_1^{(1)}, -y_2 Z, Z], \\ [p_{1,1}, q_{1,1}] &= [Z, 0, -y_2, 1], \quad [p_{1,2}, q_{1,2}] = [1, 0, 0, 0], \\ [p_{2,0}, q_{2,0}] &= [-u_2^{(1)}, Z^2, 0, -y_1 Z], \\ [p_{2,1}, q_{2,1}] &= [0, Z, 0, -y_1], \quad [p_{2,2}, q_{2,2}] = [0, 1, 0, 0], \end{aligned}$$

and by (18) the one-forms  $\omega_{1,1} = dy_1^{(1)} - y_2 du_1 + du_2$ ,  $\omega_{1,2} = dy_1$ ,  $\omega_{2,1} = dy_2^{(1)} - y_1 du_2$ ,  $\omega_{2,2} = dy_2$ .

*Theorem 3:* For the i/o model (4), and  $k = 1, \dots, s+2$ ,

$$\mathcal{H}_k = \text{span}_{\mathcal{K}} \{ \omega_{i,l}, \quad i = 1, \dots, p, \quad l = 1, \dots, n_i, \quad d\mathbf{u}_v^{(s-k+1)}, \quad v = 1, \dots, m \}. \quad (20)$$

Due to the space limitations we omit the proof, which follows the ideas of the proof in [13] for the shift-operator case.

*Example 3:* (Continuation of Examples 1 and 2). Now we find the realization of (16). For the system (16),  $m = 2$ ,  $s = 1$  and  $n_1 = n_2 = 2$ . According to Theorem 1, the differentials of the state coordinates can be found from subspace  $\mathcal{H}_{s+2} = \mathcal{H}_3$ . By Theorem 3,  $\mathcal{H}_3 = \text{span}_{\mathcal{K}} \{ \omega_{1,1}, \omega_{1,2}, \omega_{2,1}, \omega_{2,2} \}$ . Since  $\mathcal{H}_3$  is exact by Theorem 2, there exists such basis for  $\mathcal{H}_3$  whose elements are total differentials. We may choose the differentials of the state coordinates as follows  $dx_1 = \omega_{1,2} = dy_1$ ,  $dx_2 = \omega_{2,2} = dy_2$ ,  $dx_3 = \omega_{2,1} - u_2 \omega_{1,2} = d(y_2^{(1)} - y_1 u_2)$ ,  $dx_4 = \omega_{1,1} - u_1 \omega_{2,2} = d(y_1^{(1)} + u_2 - u_1 y_2)$ . In these coordinates system (16) takes the form  $x_1^{(1)} = u_1 x_2 + x_4 - u_2$ ,  $x_2^{(1)} = u_2 x_1 + x_3$ ,  $x_3^{(1)} = u_2(u_2 - u_1 x_2 - x_4)$ ,  $x_4^{(1)} = -u_1(u_2 x_1 + x_3)$ , where  $x_1 = y_1$ ,  $x_2 = y_2$ ,  $x_3 = y_2^{(1)} - y_1 u_2$  and  $x_4 = y_1^{(1)} + u_2 - u_1 y_2$ .

An alternative method for computing the 1-forms (18) in the continuous-time case was given in [10], based on the concept of adjoint polynomials. In the present paper the result of [10] is generalised in two directions. First, the formulas, obtained in [10] for MISO and SIMO systems, are extended to the MIMO case; and second, while [10] deals with the continuous-time systems, here the systems are described in terms of pseudo-linear map. Moreover, our approach is slightly different from that of [10], which used the transfer function as a starting point of the realization

while our's is the polynomial representation (15) of the i/o equations. Note that the idea of application the adjoint polynomials does not necessarily require to rely on the transfer function. On the contrary, the polynomial matrices  $P(Z)$  and  $Q(Z)$  provide a better starting point for realization, since in the MIMO case it is not possible to determine the required polynomials from the transfer matrix by direct inspection, like in the MISO case [10].

*Definition 5:* [1] The adjoint of the skew polynomial ring  $\mathcal{K}[Z; \sigma, \delta]$  is defined as the skew polynomial ring  $\mathcal{K}[Z; \sigma^*, \delta^*]$ , where  $\sigma^* = \sigma^{-1}$  and  $\delta^* = -\delta\sigma^{-1}$ .

From Definition 5, multiplication in the adjoint ring is defined by the commutation rule  $Z \cdot a = \sigma^{-1}(a) \cdot Z - \delta(a)$  for  $a \in \mathcal{K}$ . If  $p = p_n Z^n + \dots + p_1 Z + p_0$  is a polynomial in  $\mathcal{K}[Z; \sigma, \delta]$ , then the adjoint polynomial  $p^*$  is defined by the formula  $p^* = Z^n p_n + \dots + Z p_1 + p_0 \in \mathcal{K}[Z; \sigma^*, \delta^*]$ , where the products  $Z^i p_i$  must be computed in  $\mathcal{K}[Z; \sigma^*, \delta^*]$ , to yield  $p^* = p_n^* Z^n + \dots + p_1^* Z + p_0^*$ .

As the first step of the realization procedure, the adjoint operator is applied to the elements of  $P(Z)$ ,  $Q(Z)$  and the resulting matrices are denoted respectively as  $P^*(Z)$ ,  $Q^*(Z)$ , their elements being in the form

$$p_{ij}^*(Z) = \sum_{l=0}^{n_i} p_{ij,l}^* Z^l, \quad q_{ik}^*(Z) = \sum_{l=0}^{n_i} q_{ik,l}^* Z^l. \quad (21)$$

The adjoint polynomials (21) are, by definition, of degree  $n_i$ , though some of their coefficients may be equal to zero:  $n_{ij} < n_i$  yields  $p_{ij,l}^* = 0$ , for  $l > n_{ij}$ . In the similar manner, the condition  $s_{ik} < n_i$  yields  $q_{ik,l}^* = 0$  for  $l > s_{ik}$ . Next, introduce the set of 1-forms<sup>2</sup>

$$\tilde{\omega}_{i,l} := \sum_{j=1}^p p_{ij,l}^* dy_j + \sum_{k=1}^m q_{ik,l}^* du_k, \quad (22)$$

where  $i = 1, \dots, p$  and  $l = 1, \dots, n_i$ .

*Theorem 4:* For the realizable i/o model (4) the differentials of the state coordinates can be calculated as the linear combinations of the 1-forms  $\tilde{\omega}_{i,l} := \sum_{\kappa=l}^{n_i} \tilde{\omega}_{i,\kappa}^{(\kappa-l)}$ , where  $i = 1, \dots, p$  and  $l = 1, \dots, n_i$ .

The proof of the theorem is straightforward. After substituting  $p_{ji}^*(Z)$  and  $q_{jk}^*(Z)$  from (21) into (22), and  $\tilde{\omega}_{i,l}$  from (22) in turn into expression of  $\tilde{\omega}_{i,l}$ , it is easy to see that coefficients of  $\tilde{\omega}_{i,l}$  are equal to polynomial quotients in (19).

*Remark 1:* The 1-forms in Theorem 4 may be alternatively computed recursively by formula

$$\tilde{\omega}_{i,n_i} := \tilde{\omega}_{i,n_i}, \quad \tilde{\omega}_{i,l} := \tilde{\omega}_{i,l+1}^{(1)} + \tilde{\omega}_{i,l}, \quad (23)$$

where  $i = 1, \dots, p$  and  $l = 1, \dots, n_i$ .

*Example 4:* (Continuation of Examples 1 – 3). The entries of the adjoint polynomial matrices  $P^*(Z)$  and  $Q^*(Z)$  of (17) are as follows:  $p_{11}^*(Z) = Z^2$ ,  $p_{12}^*(Z) = -u_1^{(1)}$ ,  $p_{21}^*(Z) = -y_2 Z + y_2^{(1)}$ ,  $p_{22}^*(Z) = Z$ ;  $q_{11}^*(Z) = -u_2^{(1)}$ ,  $q_{12}^*(Z) = Z^2$ ,  $q_{21}^*(Z) = 0$ , and  $q_{22}^*(Z) = -y_1 Z + y_1^{(1)}$ . By (22),  $\tilde{\omega}_{11} = -y_2 du_1 + du_2$ ,  $\tilde{\omega}_{12} = dy_1$ ,  $\tilde{\omega}_{21} = -y_1 du_2$ ,  $\tilde{\omega}_{22} = dy_2$  and

<sup>2</sup>Our notation is different form that used in [10]; the 1-form  $\tilde{\omega}_{i,l}$  here is equal to  $\omega_{i+1}$  in [10].

by (23),  $\tilde{\omega}_{12} = \tilde{\omega}_{12} = dy_1$ ,  $\tilde{\omega}_{11} = \tilde{\omega}_{12}^{(1)} + \tilde{\omega}_{11} = dy_1^{(1)} - y_2 du_1 + du_2$ ,  $\tilde{\omega}_{22} = \tilde{\omega}_{22} = dy_2$ ,  $\tilde{\omega}_{21} = \tilde{\omega}_{22}^{(1)} + \tilde{\omega}_{21} = dy_2^{(1)} - y_1 du_2$ . As expected, the latter one-forms coincide with those found in Example 2.

## V. DISCUSSION AND EXAMPLES

Though the realizability conditions and the polynomial formulas show remarkable similarity for systems described in terms of shift, difference or derivation operator, it has to be pointed out that in computations one has to specify the operators  $\sigma$  and  $\delta$ . We have implemented both polynomial formulas in symbolic software package NLControl, within *Mathematica* environment. The developed programs are partly accessible via website [19]. To perform the realization task, one has first to choose from the left-hand menu whether to work with continuous- or with discrete-time systems, and then select the menu item *Realization*.

The comparison of realization programs, based on formulas (18) and (23) indicates that their efficiency depends on specification of the operators  $\sigma$  and  $\delta$ . If  $\sigma = \text{id}_{\mathcal{K}}$  and  $\delta = d/dt$ , which corresponds to the continuous-time case, the method of adjoint polynomials is slightly faster. This is due to the fact that method of adjoints can be easily carried out by applying (repeatedly) the pseudo-linear map to the skew polynomial. This is a low-level polynomial function and is performed faster than polynomial division, necessary for the method of quotients. If  $\delta = \Delta$ , which corresponds to the difference operator based discrete-time case, the situation is similar. However, if  $\delta = 0$  (the shift operator based discrete-time case), computation of the polynomial quotient can be replaced by application of the cut-and-shift operator<sup>3</sup>, defined as  $\sigma_c^{-1}(p(Z)) = \sum_{i=1}^n \sigma^{-1}(p_i) Z^{i-1}$ , which enables to express the 1-forms (18) as

$$\omega_{i,l} = \sigma_c^{-l}[p_i \cdot (Z) \quad q_i \cdot (Z)], \quad (24)$$

where  $l = 1, \dots, n_i$ ,  $i = 1, \dots, p$ , see [13] for details. This allows to compute the quotient just by applying the operator  $\sigma^{-1}$  to polynomial coefficients as the elements of  $\mathcal{K}$  and therefore, it can produce the results faster than the program involving the adjoint polynomials. The realization function in the NLControl package involves possibility to choose whether to compute the state differentials using adjoints or left quotients. However, the NLControl website based on webMathematica tools does not involve this option, the most efficient method is chosen by default – that is, in case of the continuous-time systems adjoint polynomials are used, and in case of the discrete-time systems the cut-and-shift operator as given by (24).

*Example 5:* Consider the model<sup>4</sup> describing the dynamics of the rigid flying bar, used to simulate the attitude dynamics of a helicopter [17]:

$$\begin{aligned} \text{Amg} \cos \phi - \frac{1}{2} \Lambda \psi^{(1)2} + I_x \phi^{(2)} &= T_p \\ \Lambda \phi^{(1)} \psi^{(1)} + \Xi \psi^{(2)} &= T_y, \end{aligned} \quad (25)$$

<sup>3</sup>We borrowed the definition of ‘cut-and-shift operator’ from [15], though suitably adapted for nonlinear case. Note that the extension [15] to linear parameter-varying case was suggested in [18].

<sup>4</sup>The system equations are rewritten in terms of pseudo-linear operator.

where  $\Lambda = (I_y - I_z) \sin(2\phi)$ ,  $\Xi = I_z \cos^2 \phi + I_y \sin^2 \phi$ ,  $\phi$  is the pitch angle,  $\psi$  is the yaw angle,  $I_x, I_y, I_z$  are inertia constants about the point of rotation,  $m$  is the total mass of the system, and  $T_p$  and  $T_y$  are the pitch and yaw control torques, respectively. In (25),  $p = 2$  and  $n_1 = n_2 = 2$ . Note that the realization of the system (25) can be found easily since the pseudo-derivatives of input variables  $T_y$  and  $T_p$  do not appear in (25), that is  $s = 0$ . However, we follow the formula (18) to demonstrate the applicability of the polynomial method. Note that in this example  $\sigma = \text{id}$ , and  $\delta = d/dt$ . The elements of the matrix  $P(Z)$  are  $p_{11}(Z) = Z^2 - [Amg \sin \phi + (I_y - I_z) \cos(2\phi) \psi^{(1)}]^2 / I_x$ ,  $p_{12}(Z) = -(\Lambda \phi^{(1)} / I_x) Z$ ,  $p_{21}(Z) = (\Lambda \psi^{(1)} / \Xi) Z + [\Lambda T_y + 2(I_y - I_z)(\Xi - 2I_y \sin^2 \phi)] / \Xi^2$ ,  $p_{22}(Z) = Z^2 + (\Lambda \phi^{(1)} / \Xi) Z$ , and the elements of  $Q(Z)$  as  $q_{11}(Z) = -1/I_x$ ,  $q_{12}(Z) = q_{21}(Z) = 0$ , and  $q_{22}(Z) = -1/\Xi$ .

The 1-forms  $\omega_{i,l}$ , ( $i = 1, 2, l = i = 1, 2$ ) are computed by (18):  $\omega_{1,1} = d\phi$ ,  $\omega_{1,2} = d\phi^{(1)} - (\Lambda \psi^{(1)} / I_x) d\psi$ ,  $\omega_{2,1} = d\psi$ ,  $\omega_{2,2} = (\Lambda \psi^{(1)} / \Xi) d\phi + (\Lambda \phi^{(1)} / \Xi) d\psi + d\psi^{(1)}$ . The differentials of the state coordinates can be found as integrable linear combinations of the latter 1-forms:  $dx_1 = \omega_{1,1} = d\phi$ ,  $dx_2 = \omega_{1,2} + \omega_{1,2} + (\Lambda \psi^{(1)} / I_x) \omega_{2,1} = d\phi^{(1)}$ ,  $dx_3 = \omega_{2,1} = d\psi$ ,  $dx_4 = \omega_{2,2} - (\Lambda \psi^{(1)} / \Xi) \omega_{1,1} - (\Lambda \phi^{(1)} / \Xi) \omega_{2,1} = d\psi^{(1)}$ . The state equations are as follows:  $x_1^{(1)} = x_2$ ,  $x_2^{(1)} = [2T_p - 2Amg \cos x_1 + (I_y - I_z) \sin(2x_1) x_4^2] / 2I_x$ ,  $x_3^{(1)} = x_4$ ,  $x_4^{(1)} = [T_y - (I_y - I_z) \sin(2x_1) x_2 x_4] / (I_z \cos^2 x_1 + I_y \sin^2 x_1)$ , where  $x_1 = \phi$ ,  $x_2 = \phi^{(1)}$ ,  $x_3 = \psi$  and  $x_4 = \psi^{(1)}$ .

**Example 6:** Consider the model describing the dynamics of the solar heated house [14]:

$$y^{(2)} = \vartheta_1 y^{(1)} + \vartheta_2 \frac{y^{(1)} u_1^{(1)}}{u_1} + \vartheta_3 \frac{y u_1^{(1)}}{u_1} + \vartheta_4 u_1^{(1)} u_2 - \vartheta_5 u_1^{(1)} y^{(1)} + \vartheta_6 u_1^{(1)} y, \quad (26)$$

where  $y$  is the storage temperature,  $u_1$  is pump velocity,  $u_2$  is solar intensity and  $\vartheta_1, \dots, \vartheta_6$  are constants. The system is rewritten in terms of pseudo-linear operator; however, we have to keep in mind that for this system  $\delta = 0$  and  $\theta = \sigma$ . Here  $p = 1$ ,  $m = 2$  and  $n = 2$ . The polynomial matrices, describing the system, are  $P(Z) = [p_{11}(Z)]$  and  $Q(Z) = [q_{11}(Z) \ q_{12}(Z)]$ , where  $p_{11}(Z) = Z^2 - [\vartheta_1 + (\vartheta_2 - \vartheta_5 u_1) u_1^{(1)} / u_1] Z - (\vartheta_3 + \vartheta_6 u_1) u_1^{(1)} / u_1$ ,  $q_{11}(Z) = [\vartheta_5 y^{(1)} - \vartheta_6 y - \vartheta_3 y / u_1 - \vartheta_4 u_2 - \vartheta_2 y^{(1)} / u_1] Z + (\vartheta_3 y + \vartheta_2 y^{(1)}) u_1^{(1)} / u_1^2$ ,  $q_{12}(Z) = -\vartheta_4 u_1^{(1)}$ .

Regarding that  $\delta = 0$ , the 1-forms (18) are as follows  $\omega_{11} = dy$ ,  $\omega_{12} = A dy + dy^{(1)} + (\vartheta_1 y - y^{(1)}) / u_1 du_1$ , where  $A = (-\vartheta_1 \vartheta_3 \sigma^{-1}(y) + \vartheta_3 \vartheta_5) \sigma^{-1}(y) u_1 + \vartheta_2 (-y^{(1)} + u_1 (\vartheta_6 \sigma^{-1}(y) + \vartheta_4 \sigma^{-1}(u_2))) / (\vartheta_3 \sigma^{-1}(y) + \vartheta_2 y)$ . The differentials of the state coordinates may be expressed as linear combination of  $\omega_{11}$  and  $\omega_{12}$ :  $dx_1 = \omega_{11} = dy$ ,  $dx_2 = -[(\vartheta_1 + A) / u_1] \omega_{11} + (1 / u_1) \omega_{12} = d[(y^{(1)} - \vartheta_1 y) / u_1]$ , yielding  $x_1 = y$  and  $x_2 = (y^{(1)} - \vartheta_1 y) / u_1$ . In these coordinates the state equations have the form  $x_1^{(1)} = \vartheta_1 x_1 + u_1 x_2$ ,  $x_2^{(1)} = [u_1 (\vartheta_4 u_2 + (\vartheta_6 - \vartheta_1 \vartheta_5) x_1 + (\vartheta_2 - \vartheta_5 u_1) x_2) + (\vartheta_1 \vartheta_2 + \vartheta_3) x_1] / u_1$ .

## VI. CONCLUSIONS

Simple and explicit polynomial formulas, based either on the left division or using the dual representation via adjoint polynomials have been suggested to compute the differentials of the state coordinates from the i/o representation. The set of i/o equations considered is very general, accommodating both continuous- and discrete-time systems. These two types of formulas have been implemented in Mathematica-based symbolic software and compared regarding their speed. The comparison demonstrates that the efficiency depends on system description. In case of continuous-time models and difference operator based discrete-time models the method based on adjoint polynomials is slightly faster. However, if the discrete-time system is described in terms of shift operator, the other method produces the results faster.

## REFERENCES

- [1] S.A. Abramov, H.Q. Le and Z. Li, Univariate Ore polynomial rings in computer algebra, *Journal of Mathematical Sciences*, vol. 131, no 5, 2005, pp 5885–5903.
- [2] Z. Bartosiewicz, Ü. Kotta, E. Pawluszewicz and M. Wyrwas, Algebraic formalism of differential one-forms for nonlinear control systems on time scales, *Proceedings of the Estonian Academy of Sciences. Physics. Mathematics*, vol. 56, no 3, 2007, pp 264–282.
- [3] S. Beghelli and R. Guidorzi, A new input-output canonical form for multivariable systems, *IEEE Transactions on Automatic Control*, vol. 21, no 5, 1976, pp 692–696.
- [4] J. Belikov, Ü. Kotta, and M. Tönso, “Minimal Realization of Nonlinear MIMO Equations in State-Space Form: Polynomial Approach”, *Proc. of the 50th IEEE Conference on Decision and Control and European Control Conference*, Orlando, FL, USA, 2011, pp 678–683.
- [5] M. Bronstein, M. Petkovšek, An introduction to pseudo-linear algebra, *Theoretical Computer Science*, vol. 157, 1996, pp 3–33.
- [6] D. Casagrande, Ü. Kotta, M. Tönso and M. Wyrwas, Transfer equivalence and realization of nonlinear input-output delta-differential equations on homogeneous time scales, *IEEE Transactions on Automatic Control*, vol. 55, no 11, 2010, pp 2601–2606.
- [7] R.M. Cohn, *Difference algebra*, Wiley-Interscience, New York, 1965.
- [8] G. Conte, C.H. Moog and A.M. Perdon, *Algebraic Methods for Nonlinear Control Systems*, 2nd ed., Springer-Verlag, London, 2007.
- [9] E. Delaleau and W. Respondek, Lowering the orders of derivatives of controls in generalized state space systems, *J Math Systems, Estim and Control*, vol. 5, 1995, pp 127.
- [10] M. Halás, Ü. Kotta, A transfer function approach to the realisation problem of nonlinear control systems, *Int J Contr*, 2012, accepted.
- [11] M. Halás, Ü. Kotta, Z. Li, H. Wang, C. Yuan, Submersive rational difference systems and their accessibility. In: *ISSAC 2009: Proc. of the 2009 Int. Symp. on Symbolic and Algebraic Computation*, Seoul, Korea: (eds.) May, John P. . New York, N.Y.: ACM, 2009, pp 175–182.
- [12] Ü. Kotta, P. Kotta and M. Halás, “Irreducibility and reduction of MIMO nonlinear control systems: unification and extension via pseudo-linear algebra”, *Proc. of the 2009 IEEE Int. Conf. on Control and Automation*, Christchurch, New Zealand, 2009, pp 763–768.
- [13] Ü. Kotta and M. Tönso, Realization of discrete-time nonlinear input-output equations: polynomial approach, *Automatica*, vol. 48, 2012, no. 2, pp 255–262.
- [14] L. Ljung, *System identification. Theory for the user*. 2nd edition, Prentice Hall, 1999.
- [15] P. Rapisarda, and J.C. Willems, State maps for linear systems. *SIAM J. on Contr. and Optim.*, vol. 35, 1997, no. 3, pp 1053–1091.
- [16] A. van der Schaft, Transformations of nonlinear systems under external equivalence, *Lecture Notes in Contr. and Inf. Sci.*, vol. 122, 1989.
- [17] S. Zenieh and A.L. Elshafei, Robust control of uncertain nonlinear mechanical systems using a high gain observer. *Proceedings of the ACC*, Chicago, 2000, pp 3620–3625.
- [18] R. Toth, J.C. Willems, P.S.C. Heuberger and P.M.J. van der Hof, A behavioral approach to lpv systems. *Proceedings of the European Control Conference*, Budapest, Hungary, 2009, pp. 2015–2020.
- [19] <http://www.nlcontrol.ioc.ee>