21st International Symposium on
Mathematical Theory of Networks and Systems
July 7-11, 2014. Groningen, The Netherlands

Geometric Control for a Nonlinear Sampled Data System

Tanel Mullari[1] and Kurt Schlacher[2]

Abstract— **This paper is devoted to the feedforward feedback control of sampled data control system. The starting point is a nonlinear system described by a system of explicit ordinary differential equations. The continuous-time system is converted to a time discretized one by help of the implicit Euler method, which allows us to use significantly larger sampling times compared to the explicit Euler method. The input to state linearizability by static feedback is checked, the sampled data Brunovsky normal form is the starting point for both trajectory planning and its stabilization. The design of the feedforward and feedback part is performed analogous to the time continuous case. Finally, the proposed approach is applied to a nonlinear hydraulic system, its efficiency is shown by numerical simulations, where also Coulomb friction is taken into account.**

I. INTRODUCTION

This paper is motivated by following problem in the control theory – one needs to control a real physical system, which is in general a time continuous one, but the control devices are, on the other hand, the time discrete ones, and therefore the discretization of the system is necessary. The *exact* time discretization, see e.g. [1], is possible only in case of linear time invariant systems in a straightforward manner, for a nonlinear system one needs to use the approximations like the Lie-series or the numerical methods of integration (see e.g. [2]) for non trivial systems, whereby the result yielded by this method must be 1) numerically stable and 2) accurate. Moreover, the used discretization should not destroy the general structure of the system and must maintain its important properties like accessibility, observability. We will use in this paper the *explicit* Euler discretization, which transforms the discrete-time system into the sampled-data system $x_{k+1} = F(x_k, u_k, T)$. Our method for solving the above mentioned problem is applicable only to static feedback linearizable systems and is based on the Brunovsky normal form. Because the transformation of a discrete-time system into the Brunovsky form is similar to the corresponding method for a continuous-time ones, one can use the analogical procedures at each step.

Let us consider a nonlinear, accessible control system

$$\dot{x} = f(x,u), \quad x \in I\!R^n, \quad u \in I\!R^m \tag{1}$$

with the initial conditions $x_0 = x(0)$ and the terminal conditions $x_\tau = x(\tau)$. The map f is supposed to be analytic and generically submersive, i.e. the condition $\operatorname{rank}\left[\partial f(x,u)/\partial(x,u)\right] = n$ holds everywhere except on a set of measure zero. Trajectory planning of this system means the construction of a suitable trajectory $\tilde{x}(t)$ for $t \in [0,\tau]$, satisfying (1) and the initial and final conditions, see e.g. [3]. In general this problem is quite difficult since it requires methods to solve two points boundary problems, as well as the construction of a feedback in order to suppress deviations of the actual state $x(t)$ from the planned state $\tilde{x}(t)$ cause by disturbances etc.

For a certain class of systems, namely for the *feedback linearizable* ones, see e.g. [3] and all the citations therein, the suitable inputs for steering the system along the desired trajectory can be precalculated also without integrating the state equations, using the *Brunowsky form* of the system (1). In this form, the deviations of the trajectory can be easily corrected via static feedback with the help of a linear controller.

As mentioned above, the control devices are time discrete ones, therefore we have to deal with sampled data systems. A common approach is to choose a sampling time small enough such that effects caused by sampling are negligible. Although this is no problem for modern micro controller based devices, it may cause problems for the measuring equipment. Therefore we propose an approach based on the time discretized system, where the structure of the paper is as follows. First, the basic principles of our method are described. Second, the linearizability conditions and the coordinate transformation into the Brunovsky form are considered. Third, the principal construction of a control mechanism is shown. Fourth, the planning of the desired trajectories is explained. Fifth, the design of the feedback is described. Sixth, the method is applied to a nonlinear continuous-time system. Finally this paper finishes with some conclusion.

II. DISCRETIZATION AND BRUNOVSKY NORMAL FORM

The *explicit* Euler discretization is a very simple method to convert a system of differential equations into a system of difference equations given $x_{k+1} = x_k + f(x_k, u_k)T + \mathcal{O}_2$, where T is the sampling time and $\mathcal{O}_2$ stands for the second and higher order terms. Advantages are the simplicity and the fact that explicit systems are converted to explicit ones. An important disadvantage is the the small region of numerical stability. Therefore we propose to the use *implicit*, or *backward* Euler method instead, which makes the implementation more complicated, but allows us to use larger sampling times. The implicit Euler discretization of (1) is given by

$$x_{k+1} = x_k + f(x_{k+1}, u_k)T + \mathcal{O}_2 \; . \tag{2}$$

---

*This work was not supported by any organization

[1] Tanel Mullari is with Institute of Cybernetics at Tallinn University of Technology, Akadeemia tee 21, 12618 Tallinn, Estonia `tanel.mullari@ttu.ee`

[2] Kurt Schlacher ist with Institute of Automatic Control and Control Systems Technology, Johannes Kepler University, Altenberger Straße 69, 4040 Linz, Austria `kurt.schlacher@jku.at`

Now one has to solve the equations with respect to x_{k+1} to derive the explicit relations

$$x_{k+1} \quad = \quad F\left(x_k, u_k, T\right) \ , \tag{3}$$

where higher order terms are neglected. If the exact time discretization is considered then the reader my consult [4] or [1] for the connection between input to state linearization by static feedback of time continuous and sampled data systems. In general, time discretization and linearization do not commute. But these results do not apply here, since a different (approximate) way of discretization is proposed. Therefore the method developed here consists of following steps.

1) Check the input to state linearizability by static feedback of (3).
2) In case of linearizability, transform the system via a suitable (invertible) coordinate transformation into the Brunovsky form.
3) Plan the trajectories with help of the Brunovsky normal form, this is a straightforward task.
4) Stabilize the trajectories by a suitable state feedback.

Definition 1: The Brunovsky form [5] of the (static feedback linarizable) system (2) is called the system of quasi independent chains of forward shifts

$$X_{I,l,k+1} = X_{I,l+1,k} \quad \forall\, I = 1, ..., m, \ X_{I,n_I,k+1} = v_{I,k} \tag{4}$$

$\forall\, l = 1, ..., n_I$, where X are the state coordinates of the linearized system, n_I is the length of the Ith chain, and $\sum_{I=1}^{m} n_I = n$. The system is obtained from (2) via a static feedback linearization Φ:

$$X_{I,l,k} = \Phi_{I,l,k}\left(x_k\right), \quad v_{I,k} = \phi_{I,k}\left(x_k, u_k\right). \tag{5}$$

Each chain (4) can be steered independently with the choice of the corresponding Brunovsky input $v_{I,k}$ from an arbitrary initial condition to an arbitrary terminal one at least in n_I steps.

III. LINEARIZATION BY STATIC FEEDBACK AND COORDINATE TRANSFORMATIONS

One possible way to construct the chains (4) is the following step by step method, which allows us to calculate the first links $X_{l,1,k}$ of chains as the functions of $x_{,k}$, if the system is static feedback realizable. To the linearizability conditions we should come later, but now we will describe the first step.

One starts with the system (3) and defines on the state space the first coordinate transformation

$$\phi_2 := \left\{\bar{x}_{2,\alpha,k} = \phi_{2,\alpha}\left(x_k\right), \quad \bar{u}_{2,I,k} = \phi_{2,I}\left(x_k\right)\right\}, \tag{6}$$

where $\alpha = 1, ..., m_2$ and m_2 is the maximal number of independent functions $\phi_{2,\alpha}\left(x_k\right)$ with the relative degree greater or equal than 2. This means, their first order forward shift *(called also "pullback")*, the compositions $\phi_{2,\alpha}\left(F\left(x_k, u_k\right)\right)$, does not depend on u_k. These functions are used to introduce new coordinates $\bar{x}_{2,\alpha,k} = \phi_{2,\alpha}\left(x_k\right)$. In order to make the coordinate transformation (6) invertible, one defines additionally $n - m_2$ coordinates $\bar{u}_{2,I,k}$. This leads a representation of the system in the form

$$\begin{array}{rcl} \bar{x}_{2,\alpha,k+1} & = & \bar{F}_{2,\alpha}\left(\bar{x}_{2,k}, \bar{u}_{2,k}\right) := \\ & := & \phi_{2,\alpha}\left(F\left(\phi_2^{-1}\left(\bar{x}_{2,k}, \bar{u}_{2,k}\right)\right)\right), \\ \bar{u}_{2,I,k+1} & = & \bar{F}_{2,I}\left(\bar{x}_{2,k}, \bar{u}_{2,k}, u_k\right) := \\ & := & \phi_{2,I}\left(F\left(\phi_2^{-1}\left(\bar{x}_{2,k}, \bar{u}_{2,k}\right), u_k\right)\right) \ , \end{array} \tag{7}$$

where the first part does not depend on u_k any more. Now, repeats the procedure and constructs the maximal subset of functions $\phi_{3,\alpha}\left(x_k\right)$ with the relative degree 3. By repetition one gets step by step the transformations $\bar{x}_{l,k} = \phi_{l,\alpha}\left(x_k\right)$, such that the total differentals of the functions $\phi_{l,\alpha}$ span the corresponding codistributions $\mathcal{H}_l$ of the sequence $\mathcal{H}_1 \supset \mathcal{H}_2 \supset$ ..., see [6], defined as follows:

$$\begin{array}{rcl} \mathcal{H}_1 & := & \operatorname{span}\left\{\mathrm{d}x_k\right\}, \\ \mathcal{H}_{l+1} & := & \operatorname{span}\left\{\omega\left(x_k\right) \in \mathcal{H}_l \mid F^*\omega \in \mathcal{H}_l\right\}. \end{array} \tag{8}$$

According to (8), each $\mathcal{H}_{l+1}$ is spanned by all 1-forms of $\mathcal{H}_l$, whose pullbacks also belong to $\mathcal{H}_l$. Due to definition, the forward shift of an integral $\phi_{l+1}\left(x_k\right)$ of any $\mathcal{H}_{l+1}$, the composition $\phi_{l+1}\left(F\left(x_k, u_k\right)\right)$, belongs obviously to the integrals of the previous codistribution $\mathcal{H}_l$. The sequence obtained this way does not form the complete set of coordinates apart from the case that the system is static feedback linearizable. Therefore, we check the property of linearizability of the corresponding system according to

Theorem 1: The system (3) is static feedback linearizable iff

1) all $\mathcal{H}_l$ are integrable,
2) there exists $\bar{l} < n$ such that $\dim \mathcal{H}_{\bar{l}} = 0$, see [6].

If the conditions of Theorem 1 hold, one defines the coordinate transformation $X = \Phi(x)$ as follows.

Step 1. Define the first set of coordinates $X_{l,1,k}$, $l = 1, ..., \dim \mathcal{H}_{\bar{l}-1}$ as the independent integrals of $\mathcal{H}_{\bar{l}-1}$ and also the starting links of the corresponding chains (4).

Step q. Check if the integrals of $\mathcal{H}_{\bar{l}-q+1}$ and their forward shifts form the complete set of independent integrals of $\mathcal{H}_{\bar{l}-q}$. If not, define additional variables $X_{l,1,k}$ as the starting links of new chains to complete the set.

Next we explain first, how to calculate $\mathcal{H}_2$ and then following codistributions. Due to definition (8), the forward shifts of all its integrals $\phi_{2,\alpha}\left(x_k\right)$, $\alpha = 1, ..., \dim \mathcal{H}_2^*$, do not depend on u_k. Here $\mathcal{H}_2^*$ is the maximal integrable subset of $\mathcal{H}_2$. Consequently – to determine the integrals of $\mathcal{H}_2$, one should

1) find all independent functions $\bar{\phi}_2\left(x_k\right)$, whose *backward shifts* are uniquely determined in the state space, i.e. depend only on x_k,
2) calculate their backward shifts ϕ_2 such that $\phi_2\left(F\left(x_k, u_k\right)\right) = \bar{\phi}_2\left(x_k\right)$.

One can find a backward shift of a function in following way. The map F is locally a submersion $\mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ from the state-input space with coordinates $\left(x_k, u_k\right)$ to the state space with coordinates x_{k+1}. The pre-image of each point $x_{k+1} \in \mathbb{R}^n$ is locally a submanifold of $\mathbb{R}^n$ and by definition the integral surface of $\ker TF$, which stands for the kernel of the Jacobi matrix of F. With the help of additional coordinates $z_{I,k}$, $I=1,...,m$ and functions $z_{I,k}=g_{I,k}(x_k,u_k)$, one can construct a right inverese $(x_k,u_k)=F_{z_k}^{-1}(x_{k+1})$ satisfying

$$F\circ F_{z_k}^{-1}=\mathrm{Id}\,. \tag{9}$$

Theorem 2: The forward shifts $\bar{\phi}_2(x_k)$ of the integrals $\phi_2(x_k)$ are the invariants of $\ker TF$.
Proof. Differentiating of (9) with respect to z_k gives according to the chain rule

$$\begin{aligned}&\frac{\partial}{\partial z_k}\left(F\circ F_{z_k}^{-1}(x_{k+1})\right)= &(10)\\ &=\left.\frac{\partial F(x_k,u_k)}{\partial(x_k,u_k)}\right|_{(x_k,u_k)=F_{z_k}^{-1}(x_{k+1})}\frac{\partial F_{z_k}^{-1}(x_{k+1})}{\partial z_k}\equiv 0.\end{aligned}$$

Next calculate the composition of the left hand side of (10) with $F(x_k,u_k)$:

$$\left.\frac{\partial F(x_k,u_k)}{\partial(x_k,u_k)}\frac{\partial F_{z_k}^{-1}(x_{k+1})}{\partial z_k}\right|_{x_{k+1}=F(x_k,u_k)}\equiv 0, \tag{11}$$

whereby the left hand side of (11) can be interpreted as a certain vector field

$$\begin{aligned}K_I(x_k,u_k):=&\sum_{i=1}^{n}\left.\frac{\partial F_{i,z_k}^{-1}(x_{k+1})}{\partial z_{I,k}}\right|_{x_{k+1}=F(x_k,u_k)}\frac{\partial}{\partial x_{i,k}}+\\ &\sum_{J=1}^{m}\left.\frac{\partial F_{J,z_k}^{-1}(x_{k+1})}{\partial z_{I,k}}\right|_{x_{k+1}=F(x_k,u_k)}\frac{\partial}{\partial u_{J,k}},\end{aligned} \tag{12}$$

multiplied by the Jacobi matrix of $F(x_k,u_k)$. Because the product is identically zero, the vector fields K_I, $I=1,...,m$, span the kernel of TF:

$$\ker TF=\mathrm{span}\left\{K_I\right\}.$$

On the other hand, according to the definition of the right-inverse, the backward shift of an arbitrary function $\bar{\phi}(x_k)$ is the composition $\phi(x_k,z_{k-1})=\bar{\phi}\left(F_{z_{k-1}}^{-1}(x_k)\right)$. This means, due to the submersive character of F, it depends in general case also on z_{k-1}. But, as mentioned above, our task is to find the functions $\bar{\phi}_2(x_k)$ whose backward shifts ϕ_2 are determined uniquely in the state space, i.e. depend only on x_k, and then to calculate these backward shifts. Therefore, for the compositions of ϕ_2 with $F_{z_k}^{-1}$ the following conditions must hold:

$$\begin{aligned}&\frac{\partial\bar{\phi}_2\left(F_{z_k}^{-1}(x_{k+1})\right)}{\partial z_k}= &(13)\\ &=\left.\frac{\partial\bar{\phi}_2(x_k,u_k)}{\partial(x_k,u_k)}\right|_{(x_k,u_k)=F_{z_k}^{-1}(x_{k+1})}\frac{\partial F_{z_k}^{-1}(x_{k+1})}{\partial z_k}\equiv 0.\end{aligned}$$

The composition of (13) with $F(x_k,u_k)$ yields due to (12), that the functions $\bar{\phi}_2$ as the forward shifts of the integrals of $\mathcal{H}_2$ must be the invariants of $\ker TF$:

$$\left.\frac{\partial\bar{\phi}_2(x_k)}{\partial(x_k,u_k)}\frac{\partial F_{z_k}^{-1}(x_{k+1})}{\partial z_k}\right|_{x_{k+1}=F(x_k,u_k)}=\langle\mathrm{d}\phi_2,K_I\rangle\equiv 0 \tag{14}$$

for all $I=1,...,m$.

◇

Solving the system (14) yields the functions $\bar{\phi}_{2,\alpha}(x_k)$, $\alpha=1,...,\dim\mathcal{H}_2^*$, as the invariants of $\ker TF$, but to get the integrals of $\mathcal{H}_2$, one needs also calculate the backward shifts of them as follows.

1) Express $\bar{\phi}_2(x_k)$ in terms of x_{k+1} instead, using the composition of $\bar{\phi}_{2,\alpha}(x_k)$ with $F_{z_k}^{-1}$. Due to their construction described in Theorem 2, the obtained $\phi_{2,\alpha}$ depend only on x_{k+1}.
2) Replace each coordinate $x_{i,k+1}$ in the result by the corresponding $x_{i,k}$.

Because the static feedback linearizability of (3) requires the integrability of all $\mathcal{H}_l$, then also the integrability of $\mathcal{H}_2$ must be checked. One way is to compare the number of its integrals $\phi_2(x_k)$ with its dimension, which is due to definition (8) always $n-m$. On the other hand, the forward shifts $\bar{\phi}_2$ are defined as the invariants of $\ker TF$, which do not depend on u_k, i.e. as the invariants of the distribution $\ker TF+\mathrm{span}\left\{\partial/\partial u_k\right\}$. Because it is a $2m$-dimensional distribution in the $(m+n)$-dimensional space, it can have $n-m$ invariants only, iff it is involutive. Therefore from the involutivity of $\ker TF+\mathrm{span}\left\{\partial/\partial u_k\right\}$ follows the integrability of $\mathcal{H}_2$ (and vice versa).

Consequently, the procedure to determine the invariants of $\mathcal{H}_2$ consists on following steps.

1) Check the involutivity of $\ker TF+\mathrm{span}\left\{\partial/\partial u_k\right\}$. If not, then stop.
2) Determine its invariants $\bar{\phi}_2(x_k)$.
3) Compute their backward shifts $\phi_2(x_k)$ as described above.

The next procedure is to determine the codistribution $\mathcal{H}_3$. Using the obtained functions $\phi_2(x_k)$ define the coordinate transformation (6). From the involutivity of $\ker TF+\mathrm{span}\left\{\partial/\partial u_k\right\}$ follows, that $\alpha=1,...,n-m$ and $I=n-m+1,...,n$. Having now the reduced system (7), calculating the basis $\left\{\bar{K}_2\right\}$ of the kernel of the Jacobi matrix $T\bar{F}_2$. Then

1) check the involutivity of $\mathrm{span}\left\{\bar{K}_2,\partial/\partial\bar{u}_{2,k}\right\}$;
2) determine its independent invariants;
3) calculate their backward shifts as the integrals of $\mathcal{H}_3$.

Repeat the procedure until $\mathcal{H}_{\bar{l}}$ is computed. Then carry out the final coordinate transformation $X=\Phi(x)$ using the abovementioned method.

IV. PRINCIPAL DESIGN

The Figure 1 shows the principal design of a device, allowing to steer the feedback linearizable system (1) with the help of the precalculated discrete-time Brunovsky inputs $\tilde{v}_k$. The time signal t coming from the clock passes the zero order hold (block "ZOH") in order to be converted into a corresponding discrete-time signal kT, which will be lead next into the trajectory-planning block "TP", where the precalculated inputs $\tilde{v}_k$ will be computed according to the method described in the next chapter, see (16). In the following branching point the values of $\tilde{v}_k$ will be sent into the block "PB", where the precalculated values of the Brunovsky coordinates $\tilde{X}_k$ for the desired trajectory will be determined (see (16) in the next chapter). At the same

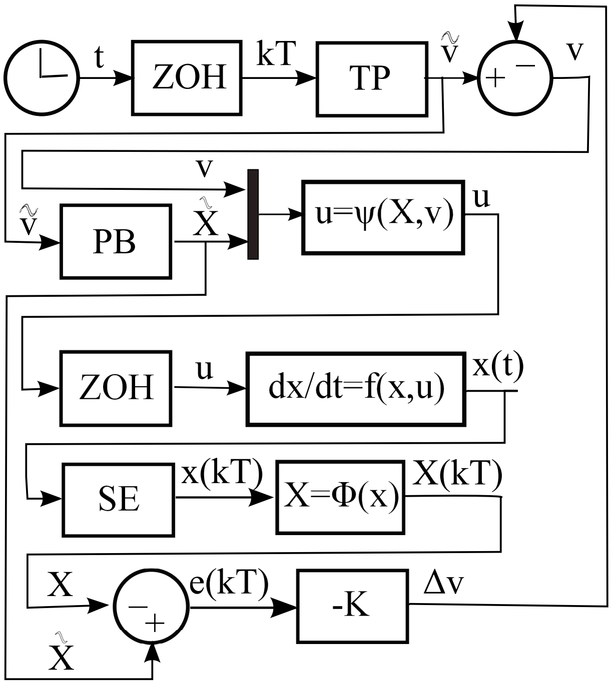

Fig. 1. Scheme of the Control Loop.

time, $\tilde{v}_k$ passes also the block denoted by the circle, where the correction Δv_k will be subtracted from its value in order to suppress the deviations of the actual (measured) trajectory from the desired one. The corrected value v_k of the Brunovsky input together with the precalculated Brunovsky coordinates $\tilde{X}_k$ will be sent via a multiplexer into the block "$u = \psi(X, v)$" (see (17) in the next chapter), where the values of the nonlinear input u_k will be calculated. After passing the next zero order hold (block "ZOH") these values will be sent after every sample time into the system described by (1) (block "$\mathrm{d}x/\mathrm{d}t = f(x,u)$"). So the input signal u_k will be kept constant during the sampling time T and the continuous system will be steered like a discrete-time one, described by its approximation (2).

The part of the device, following the block "$\mathrm{d}x/\mathrm{d}t = f(x,u)$", has the purpose to determine the corrections of $\tilde{v}_k$, necessary to keep the system tracing the desired trajectory. Its principle of working will be described in the section "Feedback". Therefore the actual (continuous-time) values $x(t)$ are measured at the output of "$\mathrm{d}x/\mathrm{d}t = f(x,u)$" and converted in the corresponding discrete-time values $x_k := x(kT)$ in the sampling element (block "SE"). In the next block "$X = \Phi(x)$" the actual values X_k of the Brunovsky coordinates are computed according to (5), describing the actual trajectory of the feedback linearized discrete-time system. Their values are in the following summation block (the second circle-shaped one) now subtracted from the precalculated ones, $\tilde{X}_k$, to obtain their deviations $e_k := e(kT)$ from the desired trajectory. Using now the values of e_k, the corrections Δv_k of the Brunovsky inputs are calculated in the block "-K" and sent into the first summation block to subtract them from the precalculated inputs.

V. PLANNING OF THE TRAJECTORIES

According to the coordinate transformation (5), defined above, the starting point of the trajectory has the coordinates $X_0 := \Phi(x_0)$ and the endpoint the coordinates $X_{\tau/T} := \Phi(x_\tau)$, where τ/T is the number of steps, necessary to reach the endpoint in time τ. Because the system moves from one equilibrium point into another, then from to the definition of the equilibrium point in the discrete-time case, $x_{k+1} = x_k$, follows, that in the Brunovsky coordinates

$$X_{I,1,k} = X_{I,1,k+1} = ... = X_{I,1,k+n_I} = \Phi_{I,1}(x_k),$$

where $I = 1, ..., m$, $k = 0$ and $k = \tau/T$. Due to definition formula (4) it means also, that

$$X_{I,1,k} = X_{I,2,k} = ... = X_{I,n_I,k} = v_{I,k} = \Phi_{I,1}(x_k) \quad (15)$$

where $I = 1, ..., m$, $k = 0$ and $k = \tau/T$. For that purpose the precalculated sequence of states and inputs in the Brunovsky form should be computed for the desired trajectory according to the formulae

$$\begin{aligned} \tilde{X}_{I,l,k} &= \sum_{q=0}^{2n_I+1} a_{I,q} \left[(k+l-1)T\right]^q, \\ \tilde{v}_{I,k} &= \sum_{q=0}^{2n_I+1} a_{I,q} \left((k+n_I)T\right)^q \end{aligned} \quad (16)$$

for all $I = 1, ..., m$, $l = 1, ..., n_I$, as the polynomials of the $(2n_I+1)$-th degree. Then the formulae (15) yield the system of equations to determine the coefficients $a_{I,q}$.
Define the coordinate and input transformation $\mathbf{\Psi}$, inverse to $\mathbf{\Phi}$ (5):

$$x_{i,k} = \Psi_i(X_{I,l,k}), \quad u_J = \psi_J(X_{I,l,k}, v_{I,k}),$$

than the precalculated nonlinear inputs $\tilde{u}_{I,k}$, necessary to move the system along the desired trajectory, should be computed as follows:

$$\tilde{u}_{J,k} = \psi_J\left(X_{I,l,k}, \sum_{q=0}^{2n_I+1} a_{I,q}\left((k+n_I)T\right)^q\right),$$

where $I, J = 1, ..., m$ und $l = 1, ..., n_I$. Here the value of each $X_{I,l,k}$ is obtained via leading the precalculated value of the corresponding $v_{I,k}$ through the chain consisting of $n_I - l + 1$ unit delay blocks. Of course other methods, based on optimization, etc., to plan the trajectories are possible.

VI. FEEDBACK

In order to suppress the deviations from the desired trajectory, one needs to correct the precalculated values of the inputs u at each step. Id est, one needs to modify the precalculated values $v_{I,k}$ according to the measured deviations. Due to the property of the Brunovsky coordinates, one can steer each chain (4) separately with the corresponding input $v_{I,k}$. For the feedback of the I-th chain, one needs to measure the actual values of the coordinates $x_{i,k}$, calculate according to (5) the actual values of the Brunovsky coordinates $X_{I,l,k}$, and then compare two systems of equations (see e.g. [9]). First, the system for $X_{I,l,k}$, and second, the system for the desired values $\tilde{X}_{I,l,k}$ in the matrix form:

$$X_{I,k+1}{=}A_I X_{I,k}{+}b_I v_{I,k},\ \tilde{X}_{I,k+1}{=}A_I \tilde{X}_{I,k}{+}b_I v_{I,k},$$

where A_I is the n_I^2-matrix and b_I the column matrix with n_I entries:

$$A = \begin{pmatrix} 0 & 1 & 0 & \cdots \\ 0 & 0 & 1 & \cdots \\ \vdots & \vdots & \vdots & \end{pmatrix}, \quad b = \begin{pmatrix} \vdots \\ 0 \\ 1 \end{pmatrix},$$

which lead to the equations for the deviations $e_{I,l,k}$ of the Brunovsky coordinates from the desired values:

$$e_{I,k+1} \;=\; A_I e_{I,k}, \quad e_{I,l,k} := \tilde{X}_{I,l,k} - X_{I,l,k}. \quad (17)$$

To keep the system moving along the desired trajectory, one needs to feedback each chain (4) with a corresponding controller, described by a row matrix k_I with n_I entries. The controlled equations of the actual trajectories will have the following form:

$$X_{I,k+1} \;=\; A_I X_{I,k} + b_I (v_I - \Delta v_I),$$

where

$$\Delta v_I := -k_I^T e_{I,k} \quad (18)$$

is the corrected input of the I-th chain. The deviations $e_{I,l,k}$ vanish, when

$$-1 \le \lambda_{I,l} \le 1 \quad \forall\, I = 1, ..., m, \quad \forall\, l = 1, ..., n_I,$$

$\lambda_{I,l}$ are the eigenvalues of the I-th controlled dynamics matrix $A_I - b_I k_I^T$. According to [9], the entries of k_I, corresponding to the desired eigenvalues, can be found with the help of the characteristic polynomial

$$p\left(\lambda_I\right) = \det\left|A_I - \lambda_I E - b_I k_I^T\right| = \sum_{i=1}^{n_I} k_{I,i} \lambda_I^{i-1}.$$

VII. EXAMPLE

As an illustration of the theory developed above let us examine a hydraulic press with the vertical cylinder, described by a system of continuous-time equations

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= \frac{1}{M}\left(S\left(p_1 - p_2\right) - Mg - \mu x_2\right), \\ \dot{p}_1 &= \frac{\beta}{l_0 + x_1}\left(-x_2 + u_1\right), \\ \dot{p}_2 &= \frac{\beta}{l - l_0 - x_1}\left(x_2 - u_2\right), \end{aligned} \quad (19)$$

see e.g. [10], where x_1 is the position of the piston, x_2 – its velocity, p_1 and p_2 are the pressures of the chambers, respectively. The system constants have the following meaning: M is the mass loaded on the piston, S – the effective piston area, μ – the damping coefficient, l_0 – the height of the chamber under the piston, l – the total length of the cylinder, and β – the isothermal bulk modulus of the oil. The inputs u_1 and u_2 are defined as follows

$$u_i = \begin{cases} \frac{K_1}{S}\sqrt{p_s - p_i}\,|U_i| & \text{if} \quad U_i \ge 0 \\ -\frac{K_2}{S}\sqrt{p_i - p_t}\,|U_i| & \text{if} \quad U_i < 0 \end{cases} \qquad i = 1, 2 \quad (20)$$

with $K_1, K_2 \in I\!R^+$, where p_s and p_t are the supply pressure and the tank pressure, respectively and U_i is the position of the valve. Please note, that these relations are invertible with respect to U_i. This input transformation is industrial standard and called servo compensation sometimes. We will make use of this compensation in the following.

We have several possibilities to steer the piston by the steering valves. We can use two valves, one for each chamber of the piston. This mode is used in high performance applications only, because it is the most expensive one. One can use one valve for both chambers, this economy mode leads to a non controllable system, but is often used in practice. Another solution is, to connect one chamber to a pressure source p_0. Here, we use the last method an introduce the new coordinates

$$x_1 = x_1,\ x_2 = x_2,\ x_3 = p_1 - p_0,\ x_4 = p_2 = p_0\ . \quad (21)$$

Therefore, we derive a simplified model, since the last state of (19) degenerates to a constant.

After the coordinate transformation (21) the system of equations, describing the remaining part of the system, reads as

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= \frac{1}{M}\left(Sx_3 - Mg - \mu x_2\right), \\ \dot{x}_3 &= \frac{\beta}{l_0 + x_1}\left(-x_2 + u_1\right). \end{aligned} \quad (22)$$

Its implicit Euler discretization (2) has the following form:

$$\begin{aligned} x_{1,k+1} &= x_{1,k} + x_{2,k+1}T, \\ x_{2,k+1} &= x_{2,k} + \frac{1}{M}\left(Sx_{3,k+1} - Mg - \mu x_{2,k+1}\right)T, \\ x_{3,k+1} &= x_{3,k} + \frac{\beta}{l_0 + x_{1,k+1}}\left(-x_{2,k+1} + u_{1,k}\right)T. \end{aligned}$$

Solving this system with respect to x_{k+1} yields the standard state space equations:

$$\begin{aligned} x_{1,k+1} &= \frac{-1}{2\left(M + \mu T\right)}\left[\left(\beta S - x_{3,k} + Mg\right)T^2 + \right. \\ &\quad \left. + \left(l_0\mu - x_{1,k}\mu - x_{2,k}M\right)T + M\left(l_0 - x_{1,k}\right) - B\right], \\ x_{2,k+1} &= \frac{-1}{2\left(M + \mu T\right)T}\left[\left(\beta S - x_{3,k} + Mg\right)T^2 + \right. \\ &\quad \left. + \left(x_{1,k}\mu - x_{2,k}M + l_0\mu\right)T + M\left(l_0 + x_{1,k}\right) - B\right], \\ x_{3,k+1} &= \frac{1}{2}\left[\frac{Mg}{S} + x_{3,k} - \beta - \right. \\ &\quad \left. - \frac{\mu\left(l_0 + x_{1,k}\right) - x_{2,k}M}{ST} + \frac{B - M\left(l_0 + x_{1,k}\right)}{ST^2}\right], \end{aligned} \quad (23)$$

where $B = \zeta\left(x_k, u_k\right)$ is an extremely complicated function, invertible with respect to u_k and obtained by solving (2) with the help of computer algebra. We will treat B as a new control variable in the process of linearization.

The next step is to define the Brunovsky coordinates X for the the accessible part according to the method described in the chapter 2. First calculate the integrals of $\mathcal{H}_2$ as follows. The Jacobi matrix of (23)

$$TF = \begin{pmatrix} \frac{1}{2} & \frac{MT}{2(M+T\mu)} & \frac{T^2}{2(M+T\mu)} & \frac{1}{2(M+T\mu)} \\ \frac{1}{2T} & \frac{M}{2(M+T\mu)} & \frac{T}{2(M+T\mu)} & \frac{1}{2T(M+T\mu)} \\ -\frac{M+T\mu}{2ST^2} & -\frac{M}{2ST} & \frac{1}{2} & \frac{1}{{}^{\prime\prime}ST^2} \end{pmatrix}$$

has an one-dimensional kernel spanned by

$$K = -\frac{1}{ST}\frac{\partial}{\partial x_{3,k}} + \frac{\partial}{\partial u_k}.$$

Then the distribution $\operatorname{span}\{K, \partial/\partial u_k\}$ is obviously involutive and has the invariants $(x_{1,k}, x_{2,k})$. Expressing them from system (23) as the functions of x_{k+1} and replacing $k+1$ by k in the result one becomes their backward shifts

$$\begin{aligned} \bar{x}_{2,1,k} &= x_{1,k} - x_{2,k}T, \\ \bar{x}_{2,2,k} &= x_{2,k} + \frac{T}{M}\left(\mu x_{2,k} - Sx_{3,k} + Mg\right) \end{aligned}$$

as the integrals of $\mathcal{H}_2$ and $\bar{u}_{2,k} = x_{3,k}$ as the new input. Calculating the forward shifts of $\bar{x}_{2,k}$ according to (23) and expressing them as functions of $\bar{x}_{2,k}$ and $\bar{u}_{2,k}$ yields the reduced system

$$\begin{aligned} \bar{x}_{2,1,k+1} &= \frac{1}{M+T\mu}\left(\bar{x}_{2,1,k}M + \right. \\ &\quad \left. + \left(\mu\bar{x}_{2,1,k} - M\bar{x}_{2,2,k}M\right)T + \left(S\bar{u}_{2,k} - Mg\right)T^2\right), \\ \bar{x}_{2,2,k+1} &= \frac{1}{M+T\mu}\left(M\bar{x}_{2,2,k} + \left(\bar{u}_{2,k}S - Mg\right)\right). \end{aligned}$$

with the Jacobi matrix

$$T\bar{F}_2 = \frac{1}{M+T\mu}\begin{pmatrix} M+\mu T & MT & ST^2 \\ 0 & M & ST \end{pmatrix},$$

whose kernel is spanned by

$$\bar{K}_2 = -\frac{ST}{M}\frac{\partial}{\partial \bar{x}_{2,2,k}} + \frac{\partial}{\partial \bar{u}_{2,k}}.$$

This vector field has the invariant

$$\bar{x}_{2,1,k} = x_{1,k} - x_{2,k}T,$$

whose backward shift is according to (23) the function

$$\bar{x}_{3,1,k} = x_{1,k} - 2x_{2,k}T + \left(\frac{1}{M}\left(Sx_{3,k} - \mu x_{2,k}\right) - g\right)T^2$$

as the only invariant of $\mathcal{H}_3$. According to the coordinate transformation method described in the third section, one sets $X_{1,k} = \bar{x}_{3,1,k}$ as the starting coordinate of the chain (4) and calculates from (23) its forward shifts up to order 2, obtaining the Brunovsky coordinates for the system (22) as follows:

$$\begin{aligned} X_{1,k} &= x_{1,k} - 2x_{2,k}T + \left(\frac{1}{M}(Sx_{3,k} - \mu x_{2,k}) - g\right)T^2, \\ X_{2,k} &= x_{1,k} - x_{2,k}T, \quad X_{3,k} = x_{1,k}, \end{aligned} \tag{24}$$

that means, using the $implicit$ Euler discretization, the position $x_{1,k}$ will be the $last$ term in the chain of forward shifts, while in case of the $explicit$ Euler discretization the chain consists of $x_{1,k}$ as the $first$ term, and its forward shifts up to order 2.

Replacing the values (23) of x_{k+1} into the last formula of (24) instead x_k allows us to determine the forward shift of $X_{3,k}$ as the Brunovsky input v_k. This yields again a complicated function of x_k, u_k and T, which contains also the abovementioned quantity B. But expressing the values of $x_{i,k}$ by $X_{i,k}$ with the help of the inverse transformation of (24), and solving the obtained equation with respect to $u_{1,k}$, we get a relatively simple formula, allowing to precalculate $u_{1,k}$ necessary to keep the system moving along the desired trajectory:

$$\begin{aligned} \tilde{u}_k = \frac{\tilde{v}_k - \tilde{X}_{3,k}}{T} + \frac{\mu(\tilde{v}_k + l_0)\left(\tilde{v}_k - 2\tilde{X}_{3,k} + \tilde{X}_{2,k}\right)}{\beta ST^2} - \\ - \frac{M\left(\tilde{v}_k + l_0\right)\left(\tilde{X}_{1,k} - 3\tilde{X}_{2,k} + 3\tilde{X}_{3,k} - \tilde{v}_k\right)}{\beta ST^2}. \end{aligned} \tag{25}$$

Here v_k and $X_{l,k}$ are calculated according to (16) in case $n_I = 3$ as the polynomials of the 7th degree, with coefficients determined from system (15), taking $x_{1,0} = l_0$, $x_{1,\tau/T} = l_\tau$, $x_{2,0} = x_{2,\tau/T} = 0$, $x_{3,0} = x_{3,\tau/T} = 0$, yielding

$$\tilde{v}_k = l_0 + \frac{(l_\tau - l_0)}{\tau^7}\left(35\tau^3 - 84t\tau^2 + 70t^2\tau - 20t^3\right)_{t=(k+3)T}. \tag{26}$$

By computing $\tilde{X}_{l,k}$ one should take $t = (k+l-1)T$ instead. Is worth of mentioning, that the precalculated input, obtained with the help of the $explicit$ Euler discretization, consists only of the first term on the right hand side of (25) and is therefore much easier to calculate, but, on other hand, it leads to significantly greater deviations from the desired tajectory, as one can see on the figure 2. There the middle curve (red) is obtained by steering the system with the precalculated input (25), the upper curve (green) corresponds to its $explicit$ counterpart. The lower curve (blue) is the desired trajectory, calculated from the right hand side of (26) taking $t = kT$. One can easily see, that the deviation from the desired trajectory is significantly greater by the precalculated input, corresponding to the explicit discretization. The trajectories have the sampling time $T = 0.02$, which is especially chosen so large in order to make the deviations comparable. This example demonstrates, that the $implicit$ Euler discretization allows the using of greater sampling times.

Besides the deviations, caused by the inaccuracies by the calculating of the approximated trajectory and which can be reduced choosing the smaller sampling times, exists an another class of deviations, caused by incorrect modeling, or also by the incorrect starting conditions. For example, the state equations (22) do not contain the Coulomb friction of the piston. If the Coulomb friction is also taken into account, the system will be significantly more complicated due to the second equation, which can have two possible forms. First – is the velocity of the piston zero and the force, caused by the oil pressure, does not exceed the sum of the weight of the

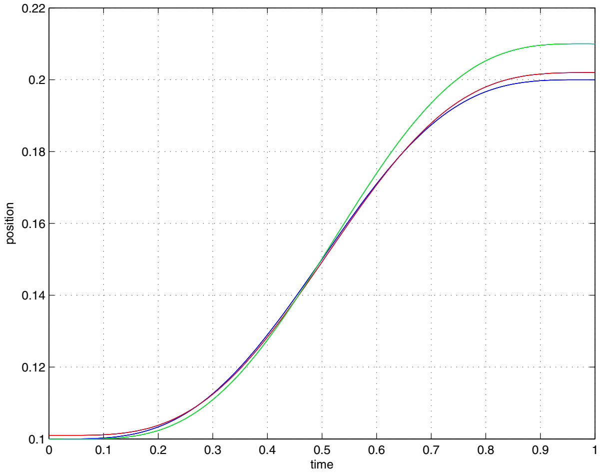

Fig. 2. The desired trajectory (blue), the trajectory corresponding to the implicit Euler discretization (red) and the trajectory corresponding to the explicit Euler discretization (green) without feedback. The position of the piston is measured in meters, the time in seconds.

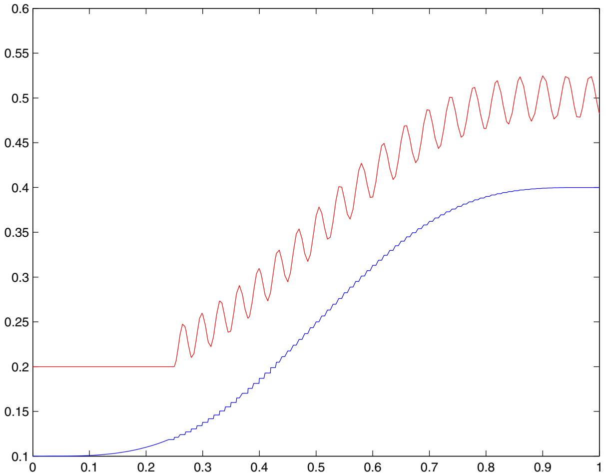

Fig. 3. Motion of the piston with the Coulomb friction. The trajectory without feedback (red) compared with the desired one (blue). The horizontal axis shows the time measured in seconds, the vertical one the position of the piston measured in meters.

load, Mg, and the Coulomb friction, F_s, then the piston will not accelerated. Second – in the case of the already moving piston the additional quantity $-\text{sgn}(x_2)F_R/M$ will be added to the right hand side of the second equation. This means

$$F_s \geq |Sx_3 - Mg| \;\&\; x_2 = 0 \quad \Rightarrow \quad \dot{x}_2 = 0, \tag{27}$$

but for $x_2 \neq 0$ the equation will read

$$\dot{x}_2 = \frac{1}{M}\left(Sx_3 - Mg - \mu x_2 - \text{sgn}(x_2)F_R\right). \tag{28}$$

The red curve on figure 3 shows the motion of the piston under the Coulomb friction, modeled with the help of MatlabR2007b$^{\text{TM}}$ [1] and based on system (22), where the second equation was replaced by (27) or (28) (the upper curve). By modeling the following values of the system constants have been used: $M = 800kg$, $l = 0.7m$, $\beta = 2.8 \cdot 10^{10}Pa$, $S = 0.01m^2$, $\mu = 300N \cdot s/m$. The measured starting position of the piston is $l_0 = 0.2m$, the desired endposition $l_\tau = 0.5m$ and the desired time of motion $\tau = 1s$. The Coulomb friction is supposed to be constant and set $F_s = 350N$. The system is steered again as in the previous case with the precalculated Brunovsky input $\tilde{v}_k$ (26), but the corresponding nonlinear input $\tilde{u}_{i,k}$ is computed with the help of (25), i.e. not taking into account the Coulomb friction. Therefore the shape of the obtained trajectory significantly differs from the desired one (the blue curve), which is shifted for the better comparison and starts at $l_0 = 0.1m$.

The motion of the piston starts, as one can see, when the force of the oil pressure exceeds the sum $Mg + F_s$ and contains also the oscillations (the red curve). The modeling with the different system constants shows, that the amplitude of the oscillation is the greater and the frequency the smaller, the greater are the mass of the load and the Coulomb friction, and the smaller is the bulk modulus of the oil. In this example $T = 0.002s$ is especially chosen ten times smaller as in the previous example, in order to diminish the inaccuracies caused by the imprecise approximation.

One possible way to get rid of the stick slip effects at the start and the oscillations would be the modifying of (25), so that also the Coulomb friction is taken into account, but this can make the formulae extremely complicated and therefore the modifying is not purposeful. The simpler way is to eliminate the deviations of this class using the feedback, described in the previous chapter. By each step the correction Δv_k (18) is added to the precalculated input (26), while the *actual* values of $X_{i,k}$, necessary to compute $e_{i,k}$ (17), are calculated with the help of (24) using the *measured* values $x_{i,k}$ of the nonlinear coordinates.

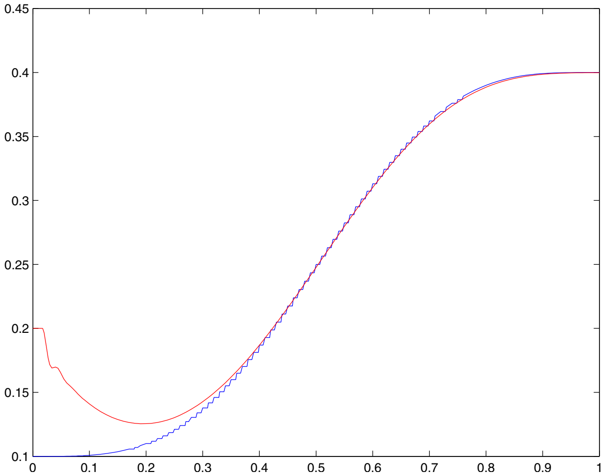

Fig. 4. Motion of the piston with the Coulomb friction. The trajectory with the feedback (red) compared with the desired one (blue). The horizontal axis shows the time measured in seconds, the vertical one the position of the piston measured in meters.

Steering the system with the input $\tilde{u}_k$ (25), where

---

[1] Matlab2007b is a product of firma Mathworks.

$v_k + \Delta v_k$ is used instead $\tilde{v}_k$ (26), the measured trajectory (red) is forced to approach the desired one (blue) already with the weak feedback. The Figure 4 illustrates the case, when the eigenvalues of the controlled dynamics matrix $\lambda_i = 0.95$ for all i, and $T = 0.002$ seconds. The motion of the piston starts with a significantly smaller stick slip effects and the oscillations are suppressed after some sampling times are elapsed.

VIII. CONCLUSIONS

In this work the method of trajectory planning of a feedback linearizable continuous-time control system is introduced, which uses its *implicit* Euler discretization. Despite the more complicated calculations, the great advantage of the implicit discretization compared with the explicit one is, that the implicit method allows to use siginificantly larger sampling times. The method is based on the Brunovsky form of the discretized equations and consists on the checking of the linearizability, the coordinate transformation into the Brunowsky form, the precalculating inputs in order to move the system between two equilibrium points and the feedback to correct the deviations from the desired trajectory. As an example, the steering of a simple hydraulic press is considered and simulated. With the help of feedback the non-predictable trajectory deviations, for example the ones caused by the Coulomb friction, are succesfully suppressed.

REFERENCES

[1] Arapostathis, A., Jakubczyk, B., Lee, H.G., Marcus, S.I. and Sontag, E.D, *The effect of sampling on linear equivalence and feedback linearization, System and Contol Letters, 13, 1989, 373-381.*

[2] Hildebrand, F.B., *Introduction to Numerical Analysis*, Dover Publications, New York, 1974.

[3] Lévine, J., *Analysis and Control of Nonlinear Systems*, Mathematical Engineering, Springer Verlag, Berlin-Heidelberg, 2009.

[4] Grizzle, J.W. and Kokotovic, P.,*Feedback linearization of sampled-data systems*, IEEE Transactions on Automatic Control, 33(9), 1988, 857-859.

[5] Brunovsky, P.A., *A classification of linear controllable systems*, Kybernetika, 3, 1970, 173-187.

[6] Aranda-Bricaire,E., Kotta,Ü., Moog,C.H. *Linearization of discrete-time systems.* SIAM J. Contr. and Optim., 34, 1996, 1999-2023.

[7] Nijmeijer, H., van der Schaft, A.J., *Nonlinear Dynamical Control Systems*, Springer-Verlag, New York 1990.

[8] Lunze, J., *Regelungstechnik 1*. Springer-Vieweg, Heidelberg-Dordrecht-London-New York, 2013.

[9] Tretelman, H.L., Stoorvogel, A.A., Hautus, M. *Control Theory for Linear Systems,* Spriner-Verlag, London-Berlin-Heidelberg, 2001.

[10] Kugi, A., *Non-linear Control Based on Physical Models*, Springer-Verlag, London-Berlin-Heidelberg, 2001.