Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Identification and Control of an Injection Moulding Machine

Sven-Olaf Lindert * **Gerald Reindl** ** **Kurt Schlacher** *

* *Institute of Automatic Control and Control Systems Technology, Johannes Kepler University Linz, Altenberger Strasse 69, 4040 Linz, Austria, (e-mail: Sven-Olaf.Lindert@jku.at)*
** *KEBA AG, Gewerbepark Urfahr, 4041 Linz, Austria*

Abstract: This paper deals with modelling and control of an injection moulding process. During the packing phase the pressure in the mould must follow certain trajectories. First a mathematical model for the packing phase is presented. Therein a characteristic function is needed which depends mainly on the form of the mould. Since the characteristic function is hardly determinable it has to be estimated automatically. This is done with cubic basis splines. The mathematical model turns out to be input to state linearisable resp. differentially flat. So trajectory planning is easily feasible and a flattnes based controller is presented. Since not every state variable can be measured an observer is added.

Keywords: Injection moulding; Identification algorithms; Control nonlinearities; Control technology; Trajectory planning; Feedback linearization; State observers.

1. INTRODUCTION

Injection moulding is the standard method to produce plastic parts. In the following the process shall be briefly described (see Rosato et al. [2000]). A turning screw transports melt into the antechamber in front of the nozzle. When there is enough melt accumulated it is injected into the mould by a sudden forward motion of the screw. The non-return valve closes to prevent the melt from flowing back. This leaves a residual melt cushion. In this article the axial movement of the screw is hydraulically performed. After the injection the melt solidifies under pressure, which is generated by the axial force onto the screw. When the melt is fully solidified the so called *packing phase* follows. During that the pressure is kept up according to certain trajectories to produce a good surface finish. The packing phase is vital for the good optical appearance of the moulded part (e. g. a faceplate for a cellular phone). The modelling and control of the packing phase is the main subject of this article.

In injection moulding one machine has to produce many different parts and the parameters of the process depend on the actual shape of the moulding part. It is neither reasonable nor practically feasible to develop a theoretical 3D model of the behaviour of each individual part. So an automatic identification is preferable. Prior to this a mathematical model is introduced. The model with the identified characteristic function is input to state linearisable and flat. Therefore flattnes based trajectory planning and tracking control is employed. Finally an observer is presented which has linear error dynamics.

There are numerous control strategies available, see e. g. Havlicsek and Alleyne [1999] and Reischl et al. [2012]. A survey can be found in Chen and Turng [2005] and references therein. In Lindert et al. [2013] a non-linear model and a flatness based control strategy are already presented. This article continues the work. Namely the identification and the observer are refined.

2. MATHEMATICAL MODEL

The mathematical model consists of the interconnected models for the mechanical and the hydraulic part. So the hydraulic part can be replaced by an electromechanical part if the screw is driven by an electrical drive.

2.1 Mechanical model

The mechanical model (see also Fig. 1) is the simple

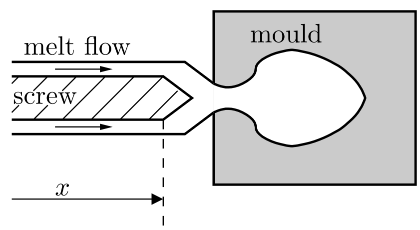

Fig. 1. Sketch of the injection moulding process

equation of linear motion with the axial coordinate x and the speed v.

$$\dot{x} = v \tag{1a}$$

$$m\dot{v} = -f_c(x, \operatorname{sgn} v) + AP \tag{1b}$$

One part of the force is applied by the hydraulic pressure P acting on the effective cross section A of the piston, which moves the screw. The mass of the screw and all attached parts is denoted with m. Since we consider short packing phases shrinkages can be neglected. During the packing phase the moulding part is almost at rest. Therefore friction is of little importance and modelled as Coulomb friction. The main part of the counterforce is modelled as the elastic force of a non-linear spring. Summing up we model the counterforce with the characteristic function $f_c(x, \operatorname{sgn} v)$ which depends on x and the sign of the speed v. Experiments indicate that this is an adequate assumption.

2.2 Hydraulic model

Due to the high pressure the oil can not be considered to be incompressible. The derivation of the hydraulic model is based on the definition of the bulk modulus E (see e.g. Kugi [2001]). An isotropic process is assumed with a constant entropy and a constant temperature, thus

$$E = \frac{\Delta P}{\frac{\Delta \rho}{\rho}}$$

with the density $\rho = \frac{m_o}{V}$ of the oil, the mass m_o of the oil, and the volume V of the piston chamber. From there one derives

$$\dot{P} = E\left(\frac{\dot{m}_o}{m_o} - \frac{\dot{V}}{V}\right) .$$

The volume of the piston chamber is $V = Ax + V_0$, with the remaining volume V_0 at $x = 0$. One continues

$$\dot{P} = E\left(\frac{Q_c \rho}{V \rho} - \frac{Av}{V}\right)$$

with the volumetric flow Q_c flowing into the piston chamber and finally

$$\dot{P} = E\frac{-Av + Q_c}{Ax + V_0} \tag{2}$$

The volumetric flow Q_c depends on the characteristic curve $g(x_v)$ of the valve and the incoming and the outgoing pressures. For instance a critically centred three land four way spool valve has

$$Q_c = \begin{cases} g(x_v)\sqrt{P_S - P}\,, & x_v \geq 0 \\ -g(x_v)\sqrt{P - P_T}\,, & x_v \leq 0 \end{cases}$$

with the servo controlled input x_v of the valve, the supply pressure P_S and the pressure P_T in the tank. The flow Q_C may be considered as an input because the pressures P, P_T and P_S are measured resp. known and the known characteristic curve $g(x_v)$ may be inverted. Thereby the static input non-linearity of the servo valve is compensated.

2.3 Identification of the characteristic function f_c with least square splines

In the mathematical model the constants are sufficiently well known. But the characteristic function f_c of the moulding part is unknown and analytically complicated. Add to this that the characteristic function changes with every moulding part. Thus it is preferable to identify it automatically. In injection moulding the first couple of parts of one series are always waste due to the putting into operation. During the putting into operation the whole process is brought into a stationary state and it is adjusted to the particular moulded part. So the first parts are waste and melted down again. So one can run certain test trajectories with a preliminary controller.

- The trajectories should extend outside the operating range which is needed for the moulding part.
- There should be periods in which the sign of the velocity v does not change.

With the latter one can divide (1b) into two cases

$$\begin{aligned} f_c^+(x) &= AP - m\dot{v} \\ f_c^-(x) &= AP - m\dot{v} \end{aligned} \quad \text{for} \quad \begin{aligned} v &> 0 \\ v &< 0\,. \end{aligned}$$

The two cases shall be separately identified. Since the identification is done off-line the measurements can be filtered with a non-causal low-pass filter to suppress measurement noise. Then it is possible to numerically differentiate x and obtain $\dot{v} = \ddot{x}$. Moreover the product $m\dot{v}$ only plays a minor role. The characteristic function is assumed to be smooth, but nothing more. So least square splines are employed. First K grid points are chosen. Then a basis is formed of cubic interpolation spline polynomials $p(x)$ of third order,

$$p_k(x_i) = \begin{cases} 1 & \text{for} \quad k = i \\ 0 & \text{otherwise} \end{cases} \qquad k, i = 1 \ldots K\,.$$

Then an overdetermined system of equations is formed

$$A\boldsymbol{P} - m\dot{\boldsymbol{v}} = \begin{pmatrix} p_1(\boldsymbol{x}) & p_2(\boldsymbol{x}) & \cdots & p_K(\boldsymbol{x}) \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_K \end{pmatrix} \tag{3}$$

with $\boldsymbol{P}$, $\boldsymbol{x}$ and $\dot{\boldsymbol{v}}$ being the vectors of the corresponding measured values. This equation is doubled for $v > 0$ and $v < 0$. They are solved with the least square method. There must be enough measurements to get an overdetermined system. Since the sampling frequency is high, this comes naturally. The identified characteristic function consist then of two spline polynomials

$$f_c^{\pm}(x) = \sum_{k=1}^{K} c_k\, p_k(x)\,. \tag{4}$$

Figure 2 shows an example of such a spline interpolation. The circles denote the measured values and the continuous lines are the spline polynomials. If one compares the

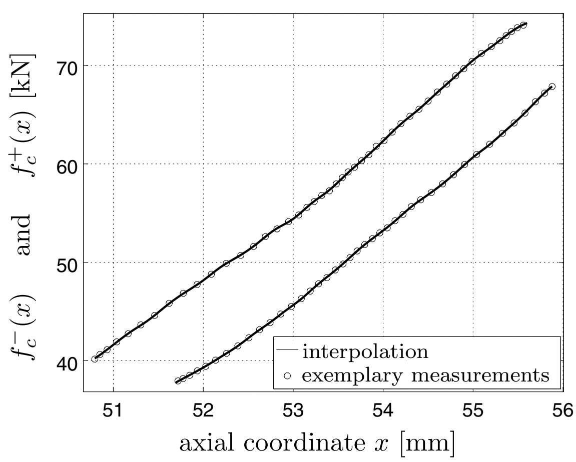

Fig. 2. characteristic function $f_c(x)$
$f_c^+(x)$ upper curve and $f_c^-(x)$ lower curve

splines $f_c^{\pm}(x)$ of different injections (Fig 3) one finds that they the corresponding splines lie approximately parallel

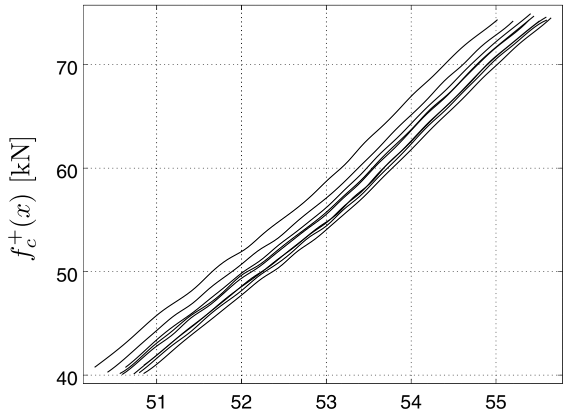

Fig. 3. interpolated characteristic functions $f_c^+(x)$ $(v > 0)$ of eight different injections

to each other with an offset. This is mainly because of the non-return valve. In every injection it closes at slightly different times leaving different residual melt cushions. So an extra constant c_{k_i} is introduced into (3). If K_i injections are measured they yield

$$A\boldsymbol{P}_{k_i} - m\dot{\boldsymbol{v}}_{k_i} = \left(p_1(\boldsymbol{x}_{k_i})\ p_2(\boldsymbol{x}_{k_i})\ \cdots\ p_K(\boldsymbol{x}_{k_i})\right)\begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_K \end{pmatrix} + c_{k_i}$$

with $k_i = 1 \ldots K_i$. The offset must be the same for $v > 0$ and $v < 0$. So the coefficients c_{k_i} are not doubled. These equations are simultaneously solved with the least square method. The last coefficient is set $k_{K_i} = -\sum_{\kappa=1}^{K_i-1} c_\kappa$. So the average value of the coefficients is zero. The varying offset is estimated later in section 5.

3. EXACT LINEARISATION VIA FEEDBACK

The following considerations are based on Isidori [1989], initially introduced by Jakubczyk and Respondek [1980]. In there affine-input systems (AI-Systems)

$$\dot{\boldsymbol{z}} = \boldsymbol{f}(\boldsymbol{z}) + \sum_{i=0}^{m} \boldsymbol{g}_i(\boldsymbol{z})\, u_i$$

are defined and the following methods are discussed with their differential geometric background. Putting eq (1) and (2) together one derives the AI-System:

$$\begin{pmatrix} \dot{x} \\ \dot{v} \\ \dot{P} \end{pmatrix} = \underbrace{\begin{pmatrix} v \\ \frac{-f_c(x,\mathrm{sgn}\, v) + AP}{m} \\ \frac{-EAv}{Ax+V_0} \end{pmatrix}}_{\boldsymbol{f}(\boldsymbol{z})} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ \frac{E}{Ax+V_0} \end{pmatrix}}_{\boldsymbol{g}(\boldsymbol{z})} Q_c$$

with the state $\boldsymbol{z}$ defined as $\boldsymbol{z} = (x\ v\ P)^T$ and the mappings $\boldsymbol{f}$ and $\boldsymbol{g}$. The flow Q_C may be considered as an input (see section 2.2).

The system has a comparatively simple structure. It is a mechanical system with one degree of freedom, which is always input to state linearisable. A hydraulic system drives the mechanical system. So the composite system is certainly linearisable. To make sure that there exits an output of relative degree 3, one calculates the distribution

$$D = \Big(\boldsymbol{g}\,,\ [\boldsymbol{f}, \boldsymbol{g}]\Big) = \begin{pmatrix} 0 & 0 \\ 0 & -\frac{AE}{m(Ax+V_0)} \\ \frac{E}{Ax+V_0} & -\frac{vAE}{(Ax+V_0)^2} \end{pmatrix}\,, \qquad (5)$$

which is indeed involutive (since the first row is zero). The squared brackets [,] denote the Lie-bracket. Next one checks, whether the following distribution has full rank:

$$\mathrm{rank}\Big(\boldsymbol{g}\,,\ [\boldsymbol{f}, \boldsymbol{g}]\,,\ [\boldsymbol{f}, [\boldsymbol{f}, \boldsymbol{g}]]\Big) = \mathrm{rank}\begin{pmatrix} 0 & 0 & \frac{E}{m(Ax+V_0)} \\ 0 & -\frac{AE}{m(Ax+V_0)} & \frac{2vA^2E}{m(Ax+V_0)^2} \\ \frac{E}{Ax+V_0} & -\frac{vAE}{(Ax+V_0)^2} & D_{33}(\boldsymbol{z}) \end{pmatrix} = 3\,.$$

This is obviously true for $Ax + V_0 \neq 0$, which is technically no problem. The working stroke of the piston limits $x \geq 0$ and $V_0 > 0$ anyway. From eq.(5) it is clear, that any function of x is a possible output of relative degree 3. So the output is chosen as $z_{k1} = x$, yielding the change of coordinates $\Phi(\boldsymbol{z})$ and its inverse $\Phi^{-1}(\boldsymbol{z}_k)$:

$$\begin{aligned} z_{k1} &= x & x &= z_{k1} \\ z_{k2} &= v & v &= z_{k2} \\ z_{k3} &= \tfrac{A\,P}{m} - \tfrac{f_c(x,\mathrm{sgn}\,v)}{m} & P &= \tfrac{m\,z_{k3} + f_c(z_{k1},\mathrm{sgn}\,z_{k2})}{A} \end{aligned} \qquad (6)$$

In the new coordinates $\boldsymbol{z}_k$ the system appears as

$$\begin{pmatrix} \dot{z}_{k1} \\ \dot{z}_{k2} \\ \dot{z}_{k3} \end{pmatrix} = \begin{pmatrix} z_{k2} \\ z_{k3} \\ \alpha(\boldsymbol{z}_k) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \beta(\boldsymbol{z}_k) \end{pmatrix} Q_c \qquad (7)$$

$$\alpha(\boldsymbol{z}_k) = -\frac{z_{k2}}{m}\left(\frac{A^2E}{Az_{k1}+V_0} + \frac{\partial f_c(z_{k1},\mathrm{sgn}\,z_{k2})}{\partial z_{k1}}\right)$$

$$\beta(\boldsymbol{z}_k) = \frac{AE}{m(Az_{k1}+V_0)}\,.$$

Since we are dealing with a single input system, $z_{k1} = x$ may serve as a flat output (Fliess et al. [1992]), which has a handy physical interpretation. With $a(\boldsymbol{z}) = \alpha\big(\Phi^{-1}(\boldsymbol{z}_k)\big)$ and $b(\boldsymbol{z}) = \beta\big(\Phi^{-1}(\boldsymbol{z}_k)\big)$ we define a new input u

$$u = a(\boldsymbol{z}) + b(\boldsymbol{z})\,Q_c \qquad \text{and} \qquad Q_c = \frac{-a(\boldsymbol{z}) + u}{b(\boldsymbol{z})}\,. \qquad (8)$$

When the entire state $\boldsymbol{z}$ is measured, the inputs u and Q_c can easily be converted into each other. With the new input u the system (7) becomes a linear system

$$\dot{z}_{k1} = z_{k2} \qquad (9a)$$
$$\dot{z}_{k2} = z_{k3} \qquad (9b)$$
$$\dot{z}_{k3} = u\,, \qquad (9c)$$

which can be treated with all known linear methods.

4. TRAJECTORY PLANNING AND TRACKING CONTROL

The planning and the tracking control is based on the differential flatness of the system. See Delaleau and Rudolph [1998] and chapter one of Rudolph [2003] for an introduction to this concept.

4.1 Trajectory planning

The flat output x may be chosen freely and arbitrarily. If x is determined and sufficiently often differentiable all other system variables are also determined and they can be calculated without solving a differential equation. One uses (9) to calculate $\boldsymbol{z}_k$, then (6) for $\boldsymbol{z}$ and eq. (8) to finally get the input Q_c.

During the packing phase it is important to hold high pressure levels for certain times and to change them as fast as possible. At the same time the maximum speed, acceleration and jerk are limited. Additionally the sign of the speed v must not change. Especially the latter together with a too high jerk r causes imperfect appearances. An increase of pressure has the limits

$$0 < v < v_{\max}\,, \qquad |\dot{v}| < a_{\max} \qquad \text{and} \qquad |\ddot{v}| < r_{\max}\,.$$

Let T denote the planning time. One starts with given pressure levels $P(0) = \overline{P}^b$ and $P(T) = \overline{P}^e$. Without loss of generality we assume that $\overline{P}^b < \overline{P}^e$. Next one calculates the associated positions $\overline{x}^b$ and $\overline{x}^e$ from (1b) as

$$A\overline{P}^b = f_c(\overline{x}^b, \operatorname{sgn} v) \quad \text{and} \quad A\overline{P}^e = f_c(\overline{x}^e, \operatorname{sgn} v)\,.$$

Between those rest positions the trajectory for $z_{k1} = x$ is planned with piecewise polynomials which are continuous and twice differentiable. The latter ensures that the functions of time $\boldsymbol{z}(t) = (x(t)\ v(t)\ P(t))^T$ of the state are continuous. The input Q_c is allowed to be discontinuous. If this were not the case the trajectories would have to be thrice differentiable. We are looking for a planning procedure which is quick to evaluate and straightforward to implement on a control unit. So a simple consideration leads to the strategy to achieve the shortest possible planning time T.

To reach $\overline{x}^e$ as fast as possible one would probably use $\dot{x} = \text{const} = v_{\max}$. But one can not jump to $\dot{x} = v_{\max}$ instantaneously. So one has to use $\ddot{x} = \text{const} = a_{\max}$ until $\dot{x} = v_{\max}$ is reached. But $\ddot{x}$ may not jump either. So the previous steps repeat until a derivative is reached which is allowed to jump. In this case this is $\dddot{x} = r$.

This leads to seven polynomials $p_\kappa(t)$ which are defined on an interval $t_{\kappa-1} \le t < t_\kappa$ with $t_0 = 0$ and $t_7 = T$:

$$\begin{aligned} p_1(t) &= \tfrac{r_{\max}}{6}(t-t_0)^3 + \overline{x}^b \\ p_2(t) &= \tfrac{a_{\max}}{2}(t-t_1)^2 + \sum_{\nu=0}^{1} g_{2\nu}\,(t-t_1)^\nu \\ \vdots\ &= \qquad \vdots \\ p_7(t) &= \tfrac{r_{\max}}{6}(t-t_6)^3 + \sum_{\nu=0}^{2} g_{7\nu}\,(t-t_6)^\nu\,. \end{aligned} \tag{10}$$

The leading coefficients are $\frac{r_{\max}}{6}$, $\frac{a_{\max}}{2}$, $-\frac{r_{\max}}{6}$, $v_{\max}$, $-\frac{r_{\max}}{6}$, $-\frac{a_{\max}}{2}$, and finally $\frac{r_{\max}}{6}$. The demand for smoothness yields $3 \cdot 8 = 24$ equations at the eight limits $t_0 \dots t_7$. At the same time we have 17 coefficients $g_{\kappa\nu}$ and seven ranges $t_\kappa - t_{\kappa-1}$. The number of equations matches the number of unknowns. Because the equations consist of $p_\kappa(t)$ and its derivatives the coefficients $g_{\kappa\nu}$ appear linearly only. So first one eliminates them to calculate the ranges

$$\begin{aligned} t_1 - t_0 &= t_7 - t_6 = \tfrac{a_{\max}}{r_{\max}} \\ t_2 - t_1 &= t_6 - t_5 = \tfrac{v_{\max}}{a_{\max}} - \tfrac{a_{\max}}{r_{\max}} \\ t_3 - t_2 &= t_5 - t_4 = \tfrac{a_{\max}}{r_{\max}} \\ t_4 - t_3 &= \tfrac{\overline{x}^e - \overline{x}^b}{v_{\max}} - \tfrac{v_{\max}}{a_{\max}} - \tfrac{a_{\max}}{r_{\max}}\,. \end{aligned}$$

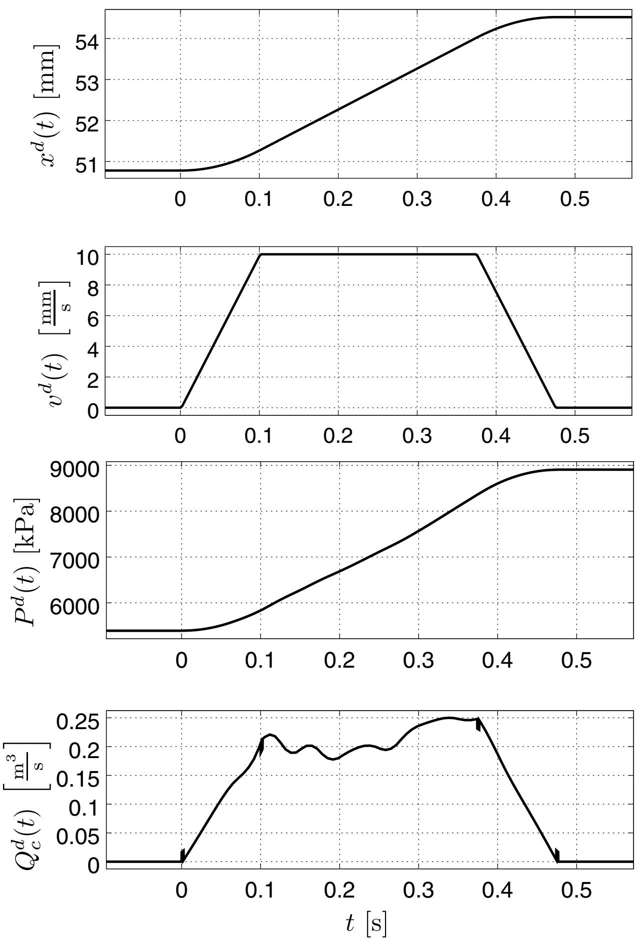

Fig. 4. result of the trajectory planning

Since the trajectory starts and ends in a rest position the terms are rather simple. If the trajectory starts and ends in arbitrary states $\boldsymbol{z}_k^b$ and $\boldsymbol{z}_k^e$ their elements would appear in the ranges. The planning time $T = t_7$ comes off the procedure along the way. One has to check whether all ranges are positive. Otherwise the corresponding polynomial is not needed because the limit is not reached. So on leaves it out and starts at eqs. (10) again. With the ranges the coefficients $g_{\kappa\nu}$ are calculated. The result of the described procedure for $v_{\max} = 0.01$, $a_{\max} = 0.1$ and $r_{\max} = 40$ can be seen in figure 4.

The above procedure also works when the trajectories must be thrice differentiable or even n-times. Then one would have $2^{n+1} \cdot (n+1)$ equations, $(2^{n+1}-1)$ ranges and $(2^{n+1} \cdot n + 1)$ coefficients $g_{\kappa\nu}$. So a symbolic calculation software becomes inevitable.

The above considerations are made for an increase of pressure ($\overline{P}^b < \overline{P}^e$). For a decrease of pressure one simply reflects the trajectory at the center axis $t = \frac{T}{2}$. If the trajectory does not start and end in a rest position all initial and terminal values have to be interchanged.

4.2 Tracking controller

The planning does not consider any disturbances or uncertainties. So a tracking controller is needed to keep the system close to the planned trajectories. By use of the differential flattnes a state-feedback controller is designed with linear error dynamics. In equation (9) the linearised system is derived. If the error of the linear system converges to zero, so does the error of the non-linear system. One defines a tracking error $z_e = z_{1k}^d - z_{1k} = x^d - x$ and postulates

$$0 = \dddot{z}_e + \alpha_2 \ddot{z}_e + \alpha_1 \dot{z}_e + \alpha_0 z_e$$

$$0 = \dddot{x}^d - \dot{z}_{3k} + \alpha_2 \left(\ddot{x}^d - z_{3k}\right) + \alpha_1 \left(\dot{x}^d - z_{2k}\right) + \alpha_0 \left(x^d - z_{1k}\right) .$$

From eq. (7) and (8) one gets $\dot{z}_{3k} = a(\boldsymbol{z}) + b(\boldsymbol{z})Q_c$ which yields the controller equation

$$Q_c = \frac{1}{b(\boldsymbol{z})}\Big[\dddot{x}^d - a(\boldsymbol{z}) + \alpha_2\big(\ddot{x}^d - z_{3k}\big) + \alpha_1\big(\dot{x}^d - z_{2k}\big) + \alpha_0\big(x^d - z_{1k}\big)\Big] . \quad (11)$$

The plan for x^d is set and $\boldsymbol{z}$ is assumed to be measured. Therefore $\boldsymbol{z}_k$ can be calculated with the transformation $\Phi(\boldsymbol{z})$, eq. (6).

4.3 Simulation results

The uncertain offset is considered to be the most disturbing factor. For the simulation we take two different characteristic functions. The mean value characteristic function is used as the "true" function whereas the planning and controlling is done with the function of one single injection. Figure 5 shows the difference. One can

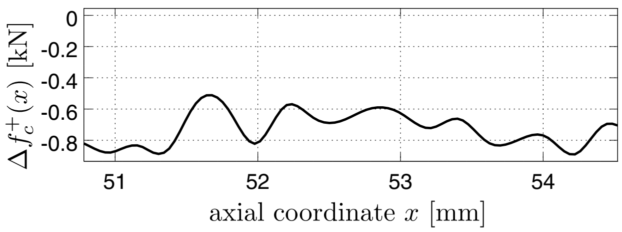

Fig. 5. difference between the used characteristic functions

see that the difference is not constant. So there are other disturbances active as well. The planned trajectories of figure 4 are taken. It is advisable to let the system start with a trajectory which is slightly above the initial value x to avoid negative velocities. So the initial state error is helpful. The simulation result is shown in figure 6. One can see, that the controller is stable but the performance is certainly not satisfactory. The sign of the speed changes and the desired final values are not met. The controller has severe problems with these constant disturbances.

5. REDUCED OBSERVER WITH LINEAR ERROR DYNAMICS

This motivates the development of an observer to estimate the offset. Add to this that the measurement of speed is questionable. So a non-linear reduced observer is designed to estimate the speed v and the offset f_{off} in (1b)

$$m\dot{v} = -f_c(x, \operatorname{sgn} v) - f_{\text{off}} + AP . \quad (12)$$

In addition this adds an integrating part to the controller. Designing an observer for a non-linear system in general is still an open task. But this system has a special property

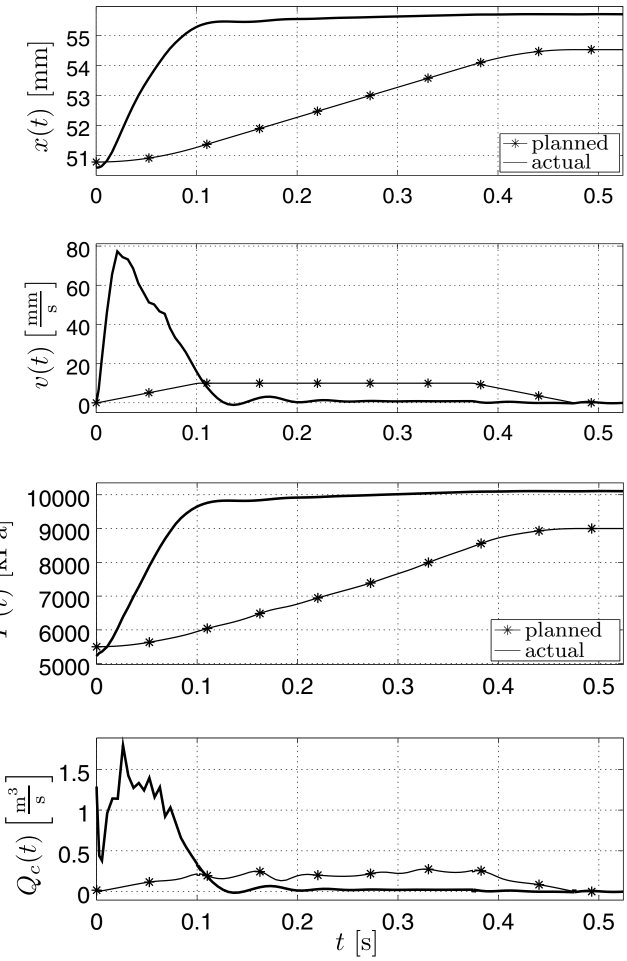

Fig. 6. Simulation of the system with tracking controller

which simplifies the task a lot. When we consider the sign of the speed known, the non-linearities appear in measured variables alone. By designing a reduced observer we can keep them out of the error dynamics. The observer state $\hat{\boldsymbol{z}}_o$ is defined alike the linear reduced observer (The hat ˆ denotes an estimated state.)

$$\boldsymbol{z}_o = \begin{pmatrix} v \\ f_{\text{off}} \end{pmatrix} + K \begin{pmatrix} x \\ P \end{pmatrix} \quad \text{and} \quad \hat{\boldsymbol{z}}_o = \begin{pmatrix} \hat{v} \\ \hat{f}_{\text{off}} \end{pmatrix} + K \begin{pmatrix} x \\ P \end{pmatrix} .$$

Then the error is

$$\boldsymbol{z}_e = \boldsymbol{z}_o - \hat{\boldsymbol{z}}_o = \begin{pmatrix} v - \hat{v} \\ f_{\text{off}} - \hat{f}_{\text{off}} \end{pmatrix}$$

and by differentiating one gets its dynamics by use of (12) and (2)

$$\dot{z}_e = \begin{pmatrix} \frac{AP - f_c(x, \operatorname{sgn} v) - f_{\text{off}}}{m} \\ 0 \end{pmatrix} + \begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix} \begin{pmatrix} v \\ E\frac{-Av + Q_c}{Ax + V_0} \end{pmatrix} - \begin{pmatrix} \frac{AP - f_c(x, \operatorname{sgn} v) - \hat{f}_{\text{off}}}{m} \\ 0 \end{pmatrix} - \begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix} \begin{pmatrix} \hat{v} \\ E\frac{-A\hat{v} + Q_c}{Ax + V_0} \end{pmatrix} .$$

So one chooses $k_{12} = 0$ and $k_{22} = 0$ and gets the linear error dynamics

$$\dot{\boldsymbol{z}}_e = \begin{pmatrix} k_{11} & -\frac{1}{m} \\ k_{21} & 0 \end{pmatrix} \boldsymbol{z}_e .$$

The characteristic polynomial $p_c(\lambda)$ of the matrix is

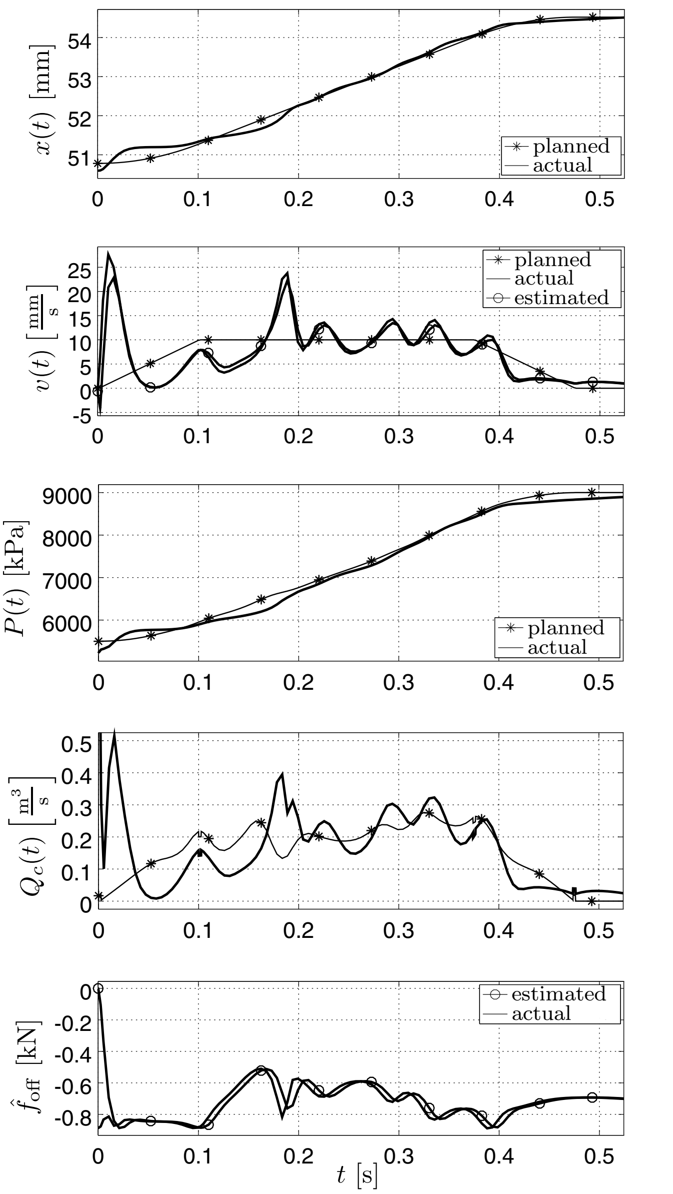

Fig. 7. Convergence of the controlled system with observer

$$p_c(\lambda) = \lambda^2 - k_{11}\lambda + \frac{k_{21}}{m}$$

which is stable for $k_{11} < 0$ and $k_{21} > 0$. The observer is added to the closed loop. In the feedback law (11) v is replaced with $\hat{v}$ and $f_c(x, \operatorname{sgn} v)$ with $f_c(x, \operatorname{sgn} v) + \hat{f}_{\text{off}}$. All other settings of the simulation 6 remain the same. The result can be seen in figure 7. This result is satisfactory. The key variable P is followed closely. The sign of the speed v does not change. The comparison with figure 5 shows that the observer estimates f_{off} quite well. The simulation indicates that the overall system with controller and observer is stable, although the stability is not rigorously proven.

6. CONCLUDING REMARK

The authors would like to mention that for the calculations in this article there was only open source software used which is released under GNU General Public License.

The authors would like to thank Jinwei Zhou for preliminary work.

Currently KEBA AG is working at the implementation of those methods on an industrial control unit and first experiments are promising.

ACKNOWLEDGEMENTS

The authors would like to thank KEBA AG for the fruitful and continuing cooperation.

Support of the present work from the Comet K2 Austrian Center of Competence in Mechatronics (ACCM) is gratefully acknowledged.

REFERENCES

Zhongbao Chen and Lih-Sheng Turng. A review of current developments in process and quality control for injection molding. *Adv. Polym. Technol.*, 24(3):165–182, 2005. doi: 10.1002/adv.20046.

E. Delaleau and J. Rudolph. Control of flat systems by quasi-static feedback of generalized states. *Int. J. Control*, 71(5):745–765, 1998. doi: 10.1080/002071798221551.

M. Fliess, J. Lévine, P. Martin, and P. Rouchon. On differentially flat nonlinear systems. In M. Fliess, editor, *Nonlinear Control Systems Design*, pages 408–412. Pergamon Press, 1992.

H. Havlicsek and A. Alleyne. Nonlinear control of an electrohydraulic injection molding machine via iterative adaptive learning. *Mechatronics, IEEE/ASME Transactions on*, 4(3):312–323, 1999. ISSN 1083-4435. doi: 10.1109/3516.789689.

Alberto Isidori. *Nonlinear Control Systems*. Springer-Verlag, Berlin, Heidelberg, 2nd. edition, 1989.

B. Jakubczyk and W. Respondek. On linearization of control systems. *Bull. Acad. Polonaise Sci. Ser. Sci. Math.*, 28:517–522, 1980.

Andreas Kugi. *Non-Linear Control Based on Physical Models: Electrical, Hydraulic, and Mechanical Systems*. Springer-Verlag, London Berlin Heidelberg, 2001. ISBN 1-85233-329-4.

S.-O. Lindert, K. Schlacher, and G. Reindl. Flatness based control of an injection moulding machine. In *Computer Aided Systems Theory - EUROCAST 2013*, number Part II, pages 279–288. Springer, 2013. ISBN 978-3-642-53861-2.

D. Reischl, R. Ludwig, and J. Gerstmayr. Automated identification of a nonlinear viscoelastic model for injection molding machines. In R. Scheidl and B. Jakoby, editors, *Proceedings of The 13th Mechatronics Forum International Conference*, pages 171–177, Linz, Austria, 2012. Trauner Verlag.

Dominick V. Rosato, Donald V. Rosato, and Marlene G. Rosato. *Injection Molding Handbook*. Kluwer Academic Publishers, Norwell, Massachusetts, 3rd. edition, 2000. ISBN 0-7923-8619-1.

Joachim Rudolph. *Flatness Based Control of Distributed Parameter Systems*. Shaker Verlag, Aachen, 2003.