Symplectic Discretization Methods for Parameter Estimation of a Nonlinear Mechanical System using an Extended Kalman Filter

Daniel Beckmann, Matthias Dagen and Tobias Ortmaier
Institute of Mechatronic Systems, Leibniz Universität Hannover, Appelstraße 11a, Hanover, Germany

Keywords: Online Estimation, Kalman Filter, Discretization Methods, Mechanical System.

Abstract: This paper presents two symplectic discretization methods in the context of online parameter estimation for a nonlinear mechanical system. These symplectic approaches are compared to established discretization methods (e.g. Euler Forward and Runge Kutta) regarding accuracy and computational effort. In addition, the influence of the discretization method on the performance of an augmented Extended Kalman Filter (EKF) for parameter estimation is analyzed. The methods are compared with a nonlinear mechanical simulation model, based on a belt-drive system. The simulation shows improved accuracy using simplectic integrators in comparison to the conventional methods, with almost the same or lower computational cost. Parameter estimation based on the EKF in combination with the simplectic integration scheme leads to more accurate values.

1 INTRODUCTION

Online state and parameter estimation is important in various fields, e.g. adaptive control, predictive maintenance and other engineering areas. Especially for stochastic state space systems the state-augmented Extended Kalman Filter (EKF) is a widely used observer to estimate states and parameters in real time (Grewal and Andrews, 2008). In particular it is used in several industrial applications for control, diagnostics, sensor data fusion and signal processing (Auger et al., 2013; Szabat and Orlowska-Kowalska, 2012). The EKF uses linearized system and measurement models at the current estimate using partial derivations (Welch and Bishop, 2006). This can lead to poor performance when models are highly nonlinear, because the covariance is propagated through the linearization. Furthermore, using the EKF as parameter estimator for linear systems can lead to biased parameter estimates, if covariance values are incorrect (Ljung, 1979).

Moreover, accurate state and parameter estimation by the observer also requires precise system modeling and proper initial values. Traditionally, mechanical systems are modeled using continuous-time methods (e.g. Lagrange's equations) to obtain the equations of motion. The resulting equations (typically second order ordinary differential equations) are transformed into continuous-time state space form. A continuous-discrete Extended Kalman Filter can be used to solve the continuous-time state space system and its covariance propagation numerically (Schütte et al., 1997; Bohn, 2000).

Another approach uses explicit discretization methods to find an analytical expression of a discrete-time model (Riva et al., 2015). The most common practice to find the analytical expression is to use the Euler Forward method with the discretization order one. The Jacobian of the discrete-time model, which can be calculated analytically and evaluated at the current estimate, is used to propagate the error covariance matrix.

However, using explicit discretization methods can lead to unstable discrete-time models, if the sampling time is too large compared to the time constant of the continuous model. In many applications, the sampling time is fixed by the used hardware (e.g. for industrial application usually 1 ms). Therefore, the way to obtain a stable discrete-time model is to raise the order of the discretization method. Increasing the order results in higher accuracy of the discrete-time solution, but the computational effort will be increased, too. In general, a trade-off between calculation time and accuracy needs to be found. In fact, there are many other implicit methods to calculate a discrete-time solution (like collocation or implicit Runge-Kutta methods), but usually all of these methods need an iterative gradient-based optimization algorithm (e.g. Newton-Raphson method) at each time step. Due to this fact, this paper focuses explicit (and

Beckmann, D., Dagen, M. and Ortmaier, T.
Symplectic Discretization Methods for Parameter Estimation of a Nonlinear Mechanical System using an Extended Kalman Filter.
DOI: 10.5220/0005973503270334
In *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2016) - Volume 1*, pages 327-334
ISBN: 978-989-758-198-4

semi implicit) methods.

As already mentioned, the most common discretization method is the Euler Forward approach. Higher order methods are Heun method (order two), Simpson rule (order three) and classical Runge Kutta method (order four). In the context of online parameter estimation for a mechanical system, this paper presents the first application and results using semi implicit Euler and leapfrog discretization method. Both approaches belong to the scheme of symplectic integrators (Hairer et al., 2006). This paper compares these methods to the conventional methods mentioned above, in context of accuracy and computing time of the discrete-time model (and its Jacobian). In addition, the influence of the discretization method to the accuracy of estimated parameter using an EKF is analyzed. The model, which is used for the investigation, is based on a nonlinear belt-drive system for positioning tasks. The model parameters are identified using Grey-box identification methods.

The paper is organized as follows. Section 2.1 shortly presents the algorithm of EKF and state-augmented EKF. The discretization methods are introduced in section 2.2. In section 3 the testbed and its modeling are given. The simulation results are shown in section 4. The paper closes with a conclusion in section 5.

2 METHODS

2.1 Extended Kalman Filter

Consider the following nonlinear discrete-time state space model with additive noise:

$$x_k = f(x_{k-1}, \theta_{k-1}, u_k) + w_{k-1}, \tag{1}$$
$$y_k = h(x_k, \theta_{k-1}) + v_k, \tag{2}$$

with the state vector $x_k \in \mathbb{R}^{n_x}$, the unknown parameter vector $\theta_k \in \mathbb{R}^{n_\theta}$, the input $u_k \in \mathbb{R}^{n_u}$ and the output $y_k \in \mathbb{R}^{n_y}$. The (non)linear system and measurement functions are described by f and h. Process and measurement noises are represented by $w_{k-1} \sim \mathcal{N}(0, Q_x)$ and $v_k \sim \mathcal{N}(0, R_k)$, with Q_x and R_k the process and measurement noise covariance matrices, respectively.

The initialization of EKF is defined by setting the start values for the state vector x_0 and its error covariance matrix P_0:

$$\begin{aligned} \hat{x}_0 &= x_{\text{init}} \in \mathbb{R}^{n_x} \quad \text{and} \\ P_0 &= P_{\text{init}} \in \mathbb{R}^{n_x \times n_x}. \end{aligned} \tag{3}$$

In addition $Q_x \in \mathbb{R}^{n_x \times n_x}$ and $R_k \in \mathbb{R}^{n_y \times n_y}$ need to be set. These matrices are assumed to be diagonal. The recursion of the EKF starts with the time update step as follows:

$$\hat{x}_k^- = f(\hat{x}_{k-1}, \theta_{k-1}, u_k), \tag{4}$$
$$P_k^- = A_k P_{k-1} A_k^{\mathrm{T}} + Q_x, \tag{5}$$

with

$$A_k = \left.\frac{\partial f}{\partial x_k}\right|_{x_k=\hat{x}_k^-, \theta_{k-1}, u_k} \in \mathbb{R}^{n_x \times n_x}. \tag{6}$$

After this prediction step, the correction step can be calculated as:

$$K_k = P_k^- H_k^{\mathrm{T}} \left(H_k P_k^- H_k^{\mathrm{T}} + R_k\right), \tag{7}$$
$$\hat{x}_k = \hat{x}_k^- + K_k \left(y_k - h(\hat{x}_k^-, \theta_{k-1})\right), \tag{8}$$
$$P_k = (I - K_k H_k) P_k^-, \tag{9}$$

with

$$H_k = \left.\frac{\partial h}{\partial x_k}\right|_{x_k=\hat{x}_k^-, \theta_{k-1}} \in \mathbb{R}^{n_y \times n_x}. \tag{10}$$

In order to estimate unknown parameters, the state vector can be augmented to ${}_{(a)}x_k = [x_k^{\mathrm{T}}, \theta_k^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{n_a = n_x + n_\theta}$ which leads to a new nonlinear discrete-time state space model as:

$$\underbrace{\begin{bmatrix} x_k \\ \theta_k \end{bmatrix}}_{{}_{(a)}x_k} = \underbrace{\begin{bmatrix} f(x_{k-1}, \theta_{k-1}, u_k) \\ \theta_{k-1} \end{bmatrix}}_{\tilde{f}({}_{(a)}x_{k-1}, u_k)} + \begin{bmatrix} w_{k-1} \\ \omega_{k-1} \end{bmatrix}, \tag{11}$$

$$y_k = \tilde{h}\left({}_{(a)}x_k\right) + v_k, \tag{12}$$

with the modified system and measurement functions $\tilde{f}$ and $\tilde{h}$. The dimension is increased by the number of unknown parameters. Then the initialization of the EKF is now defined as:

$$\begin{aligned} {}_{(a)}\hat{x}_0 &= \left[x_{\text{init}}^{\mathrm{T}}, \theta_{\text{init}}^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{n_a} \quad \text{and} \\ {}_{(a)}P_0 &= \begin{bmatrix} P_{\text{init}} & 0 \\ 0 & \Theta_{\text{init}} \end{bmatrix} \in \mathbb{R}^{n_a \times n_a}. \end{aligned} \tag{13}$$

Furthermore, the dimension of process noise covariance matrix is also increased to:

$${}_{(a)}Q = \begin{bmatrix} Q_x & 0 \\ 0 & Q_\theta \end{bmatrix} \in \mathbb{R}^{n_a \times n_a}, \tag{14}$$

where $Q_\theta \in \mathbb{R}^{n_\theta \times n_\theta}$ describes the covariance matrix of pseudo noise ω_{k-1} for the parameter estimation and is also assumed to be diagonal.

2.2 Discretization Methods

Consider the following nonlinear continuous-time state equation

$$\dot{x} = f(x, u), \tag{15}$$

with the state vector x and the control vector u. Using the Euler Forward method is the simplest way to obtain a discrete-time state equation which is given by

$$x_k = x_{k-1} + T_S f\left(x_{k-1}, u_{k-1}\right), \tag{16}$$

where T_S is defined by the sampling time. The order of the Euler Forward method is one. However, due to the order one, the method is often not accurate enough to reproduce the true dynamics of the continuous-time model. In addition, the numerical solution can be unstable, especially for stiff equations. To obtain a stable numerical solution, either the sampling time can be reduced or the order of the discretization method has to be raised. In many industrial applications the sampling time is limited by the hardware. Raising the order leads to Runge Kutta methods. For example, the Heun method has the order two and is given by:

$$\begin{aligned} \tilde{x}_k &= x_{k-1} + T_s f\left(x_{k-1}, u_{k-1}\right), \\ x_k &= x_{k-1} + \frac{T_s}{2}\big(f\left(x_{k-1}, u_{k-1}\right) \\ &\quad + f\left(\tilde{x}_k, u_k\right)\big). \end{aligned} \tag{17}$$

It is based on the prediction of the state vector $\tilde{x}_k$ in the first step and a correction in the second step. A method of order three, is given based on the Simpson rule (or Runge Kutta three). The calculation is defined by:

$$\begin{aligned} a_1 &= f(x_{k-1}, u_{k-1}), \\ a_2 &= f\left(x_{k-1} + \frac{T_S}{2} a_1, u_{k-1/2}\right), \\ a_3 &= f\left(x_{k-1} - T_S a_1 + 2T_S a_2, u_k\right), \\ x_k &= x_{k-1} + \frac{T_S}{6}\left(a_1 + 4a_2 + a_3\right), \end{aligned} \tag{18}$$

where $u_{k-1/2}$ is given by the mean value between u_{k-1} and u_k. The classical Runge Kutta method with its order four is given by:

$$\begin{aligned} a_1 &= f(x_{k-1}, u_{k-1}), \\ a_2 &= f\left(x_{k-1} + \frac{T_S}{2} a_1, u_{k-1/2}\right), \\ a_3 &= f\left(x_{k-1} + \frac{T_S}{2} a_2, u_{k-1/2}\right), \\ a_4 &= f\left(x_{k-1} + T_S a_3, u_k\right), \\ x_k &= x_{k-1} + \frac{T_S}{6}\left(a_1 + 2a_2 + 2a_3 + a_4\right). \end{aligned} \tag{19}$$

All these methods are explicit, i.e. only previous values of the state vector x are used to calculate the actual value. The input variable u is used at time step k and $k-1$, which is usually available in many applications.

However, each of the described methods can result in an unstable solution. Raising the order is an option to obtain a stable equation, but it also increases the computational time. Hence, a trade-off between accuracy and computing time has to be found. In fact, there are many other methods for discretization. Especially for mechanical systems, symplectic integration methods are notable and are discussed in the following.

The semi implicit Euler method (also called symplectic Euler) is the simplest way to obtain a symplectic integration (Niiranen, 1999). It is a modification of the Euler Forward method to solve Hamilton's equation and the results are more accurate than classical approaches. However, this method requires a pair of differential equations of the form

$$\begin{aligned} \frac{\mathrm{d}s}{\mathrm{d}t} &= \dot{s} = \Phi\left(v\right), \\ \frac{\mathrm{d}v}{\mathrm{d}t} &= \dot{v} = \Psi\left(s, v\right), \end{aligned} \tag{20}$$

where s represents a position vector and v the corresponding velocity vector. These equations are typical for classical mechanics. Using the symplectic Euler method leads to the discrete-time equations

$$\begin{aligned} v_k &= v_{k-1} + T_s \Psi\left(s_{k-1}, v_{k-1}\right), \\ s_k &= s_{k-1} + T_s \Phi\left(v_k\right). \end{aligned} \tag{21}$$

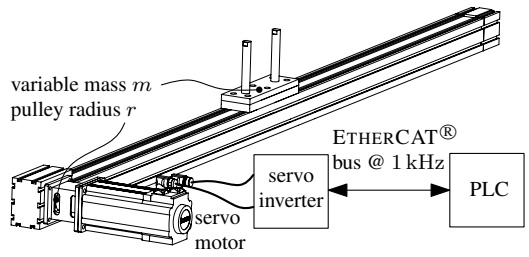
It is important to note, that the velocity vector is calculated first, using only old values of velocity and position. In the second step, the position is calculated using the new value of velocity and the old position value. The simplectic Euler method has order one. A second order method, called leapfrog integration, is given by two steps of symplectic Euler:

$$\begin{aligned} v_{k-1/2} &= v_{k-1} + \frac{T_s}{2}\Psi\left(s_{k-1}, v_{k-1}\right), \\ s_{k-1/2} &= s_{k-1} + \frac{T_s}{2}\Phi\left(v_{k-1/2}\right), \\ s_k &= s_{k-1/2} + \frac{T_s}{2}\Phi\left(v_{k-1/2}\right), \\ v_k &= v_{k-1/2} + \frac{T_s}{2}\Psi\left(s_k, v_{k-1/2}\right). \end{aligned} \tag{22}$$

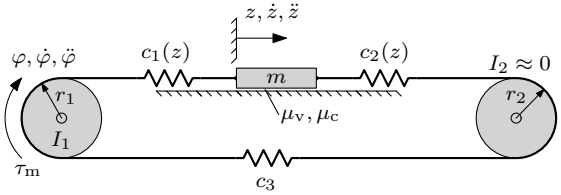
The mentioned discretization methods in combination with the parameter estimation are evaluated based on a mechanical testbed model, which is described in the following section.

3 TESTBED AND MODELING

The structure of the testbed is shown in Figure 1(a) and consists of a belt-drive for positioning tasks with



(a) Structure of the testbed with commercial hardware.



(b) Schematic of mechanical model.

Figure 1: Structure and mechanical model of the testbed.

variable transporting masses. It is directly driven by a permanent magnet synchronous motor and a servo inverter. The latter is connected via a real-time ETHERCAT® bus to a programmable logic controller (PLC). The motor is controlled by a conventional cascade structure with P- and PI-feedback controller for motor position and speed (filtered position derivative).

The main focus in industrial motion design is time optimality, so acceleration trapezoid profiles (ATP) are used to calculate the setpoint values for the motion control. Due to the exploitation of at least one limitation of velocity, acceleration and jerk ($v_{\max}$, $a_{\max}$ and $j_{\max}$) at each instant of a movement, these profiles guarantee the shortest possible traveling time (Biagiotti and Melchiorri, 2008). In this context, the jerk time is defined as $t_j = a_{\max} j_{\max}^{-1}$.

A mechanical model is given by a flexible body system with two degrees of freedom and the following equations of motion (see Figure 1(b))

$$\mathcal{F} = \mathcal{M}\ddot{q} + \mathcal{D}\dot{q} + \mathcal{C}(q)q + \mathcal{H}(\dot{q}), \tag{23}$$

with the generalized coordinate vector $q = [\varphi, z]^T$ and

$$\mathcal{M} = \begin{bmatrix} I_m + I_1 & 0 \\ 0 & m \end{bmatrix}, \mathcal{D} = \begin{bmatrix} dr_1^2 & -dr_1 \\ -dr_1 & d + \mu_v \end{bmatrix}, \tag{24}$$

$$\mathcal{C}(q) = \begin{bmatrix} c_{\text{eff}}(z) r_1^2 & -c_{\text{eff}}(z) r_1 \\ -c_{\text{eff}}(z) r_1 & c_{\text{eff}}(z) \end{bmatrix}, \tag{25}$$

$$\mathcal{H}(\dot{q}) = \begin{bmatrix} 0 \\ \mu_c \tanh(k\dot{z}) \end{bmatrix}, \mathcal{F} = \begin{bmatrix} \tau_m \\ 0 \end{bmatrix}. \tag{26}$$

Here, τ_m is the actual motor torque, r_1 is the pulley radius, φ and z the motor angle and load position, respectively. The parameters in the mass matrix $\mathcal{M}$ are the motor inertia I_m, the inertia of the connection between motor and belt drive I_1 and the load mass m. The elements of the damping matrix $\mathcal{D}$ are the belt damping d (not shown in the figure) and a viscous friction coefficient on the sliding mass μ_v. The stiffness matrix $\mathcal{C}$ contains the position depending belt stiffness $c_{\text{eff}}(z)$.

The belt stiffness is calculated by the series and parallel connection of the spring constants of each belt segment (see Figure 1(b)):

$$c(z) = c_1(z) + \frac{c_2(z) c_3}{c_2(z) + c_3}, \tag{27}$$

where $c_1(z)$ and $c_2(z)$ are functions of the load position (Nevaranta et al., 2015). An approximation of (27) is given by:

$$c_{\text{eff}}(z) = c_{\text{spec}} \left(1 + \frac{1}{z_0 + z}\right) \approx c(z), \tag{28}$$

where c_{spec} is a belt specific spring constant and z_0 a position constant depending on the initial value of load position z. In addition to the viscous friction at the sliding mass, an approximation of a static Coulomb friction model (parameter μ_c and k) is assumed in the nonlinear term $\mathcal{H}(\dot{q})$.

Using the state space vector $x = [q^T, \dot{q}^T]^T = [\varphi, z, \dot{\varphi}, \dot{z}]^T$, the system can be transformed into the continuous-time state space equation

$$\dot{x} = \begin{bmatrix} x_3 \\ x_4 \\ (I_m + I_1)^{-1} (\tau_m - r_1 F_d - r_1 F_c) \\ m^{-1} (F_d + F_c - F_f) \end{bmatrix}, \tag{29}$$

with

$$\begin{aligned} F_d &= d\,(x_3 r_1 - x_4), \\ F_c &= c_{\text{eff}}(x_2)(x_1 r_1 - x_2), \\ F_f &= \mu_v x_4 + \mu_c \tanh(k x_4). \end{aligned} \tag{30}$$

The motor angle and its velocity can be measured, which leads to the following measurement function:

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} x. \tag{31}$$

Due to the transformation to state space, the pair of equations needed for the symplectic integration methods is already given.

The parameter values of the continuous-time state space model are identified using Grey-box identification method (Bohlin, 2006) with a dynamic ATP (jerk time < 10 ms) to excite all system parameters. The result of the identification is shown in Figure 2, where the measured signals and the model are compared.

The model fits the measured position with 99.88 % and the measured velocity with 98.32 %. The identified parameter values are summarized in Table 1.

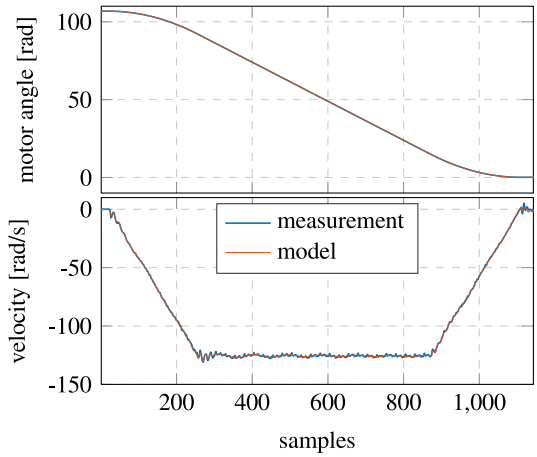


Figure 2: Comparison of measured motor angle and velocity against the identified model output.

Table 1: Identified parameter values.

Parameter	Value	Unit
$I_m + I_1$	13.00	kgcm^2
m	12.50	kg
c_{spec}	8.94	kN
d	97.66	Ns/m
μ_v	13.67	Ns/m
μ_c	28.13	N
k	50.00	s/m
z_0	0.13	m
r_1	15.90	mm

4 RESULTS

The simulation results are divided in two parts. The first part focuses the comparison of the continuous-time model and the described discretization methods, which is pointed out in section 4.1. In the second part, the influence of discretization method to parameter estimation based on an augmented EKF is shown (section 4.2).

4.1 Comparison of Continuous and Discrete Model

First, the simulation results using the mentioned discretization methods (see section 2.2) are compared to the continuous-time model. The system structure and the data flow are shown in Figure 3. The time continuous solution is calculated using Dormand-Prince method with order eight (`ode8`-solver in SIMULINK®) with a sampling frequency of 16 kHz. The discrete-time model is calculated with a fixed sampling time of 1 ms. The exact analytical ex-

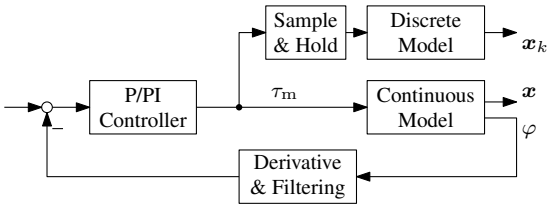


Figure 3: Control structure.

pressions for the discrete-time models are generated and optimized with the symbolic calculation software MAPLE®.

The comparison of the discrete-time models is shown in Figure 4. The first plot illustrates the continuous motor velocity for a given trajectory. To compare the discretization methods, the system dynamics need to be excited by the trajectory. The system vibrations occur in areas where the acceleration changes with small jerk time, exemplary shown in the detail plots 1 and 2. The first detail plot (middle) shows the behavior of the models near maximum velocity and the second detail plot (bottom) at minimum velocity.

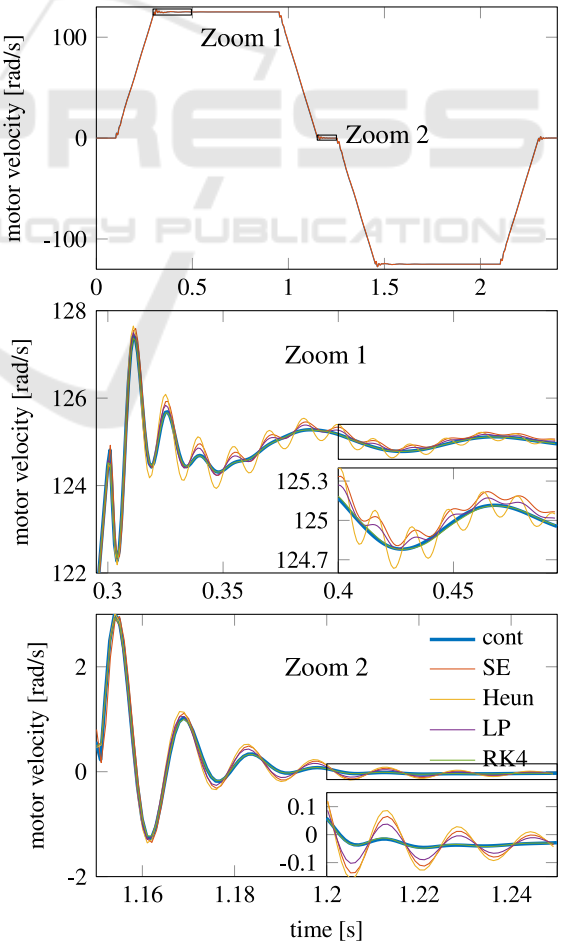


Figure 4: Comparison of discrete-time models.

Table 2: Comparison of computing cost and normalized root mean square errors of discretization methods.

	Mult	Add	Div	NRMSE1	NRMSE2
EF	11	13	1	-	-
SE	11	13	1	0.18	0.16
Heun	28	33	2	0.29	0.19
LP	22	31	2	0.13	0.14
RK3	53	58	3	0.07	0.08
RK4	64	72	4	0.06	0.07

The classical Runge Kutta method (RK4, green line) fits the continuous signal (thick blue line) nearly perfect. In comparison to the Simpson rule, nearly no difference is visible, so this method is not shown in the figure. Also not shown is the Euler Forward method (EF), which is immediately unstable in this case. The Heun method (yellow line) is able to capture the first oscillations, but can't reproduce the correct damping.

The symplectic integration methods are more accurate than the Heun method reproducing the vibrations. In fact, the RK4 method is still better than the semi implicit Euler (SE, orange) and the leapfrog (LP, purple), but the oscillations settle earlier in comparison to the Heun method. It should be noted, that the SE method has the order one and the classical Heun method has the order two. The normalized root mean square errors (NRMSE) of the detail plots 1 and 2 are printed in Table 2.

Furthermore, the number of operations for the numerical evaluation of the discretization methods is analyzed using the `cost`-function of MAPLE. The results are summarized in Table 2 and the abbreviations are defined as: additions (Add), multiplications (Mult), and divisions (Div). As expected, the methods with order one have the same smallest number of all numerical evaluations. Methods with second order (Heun and LP) need one more division plus extra multiplications and additions compared to order one approaches. To reach Runge Kutta three method, the effort is nearly doubled from methods with order two.

It is important to note, that the symplectic integrators have another characteristic in this scenario. As seen in Figure 4 (middle plot), the error between the time continuous and the time discrete signal at constant velocity is slightly higher compared to the established methods. For example, the relative error of symplectic Euler method at 0.8 s is about 0.05 %, while the error of RK4 is lower than 0.002 %. However, due to the lower implementation effort, this error is acceptable. In context of parameter estimation this error has nearly no negative effect on the estimated values, which is shown in the next section.

4.2 Parameter Estimation

Next, the discrete-time models are compared in order to estimate parameters with an augmented EKF. The parameters to estimate are chosen to be the specific spring constant c_{spec} and the belt damping d, because these parameters mainly determine the oscillation behavior. For example, the estimated parameters can be used for feedforward control methods to reduce or eliminate these vibrations (e.g. flatness-based control (Beckmann et al., 2015) or input-shaping technique (Öltjen et al., 2015)).

The state space vector is augmented to ${}_{(a)}x_k = [\varphi, z, \dot{\varphi}, \dot{z}, c_{\text{spec}}, d]^{\mathrm{T}}$. The discrete-time model from the control structure (see Figure 3) is changed to an augmented EKF. The continuous-time model is simulated without process noise, only the measurement noise is added on the position and velocity. The EKF parameters are set to:

$$\begin{aligned} Q_x &= P_{\text{init}} = I\,10^{-10}, \\ Q_\theta &= \text{diag}(8\times10^{-12}, 8\times10^{-12}), \\ \Theta_{\text{init}} &= \text{diag}(1\times10^{-1}, 1\times10^{-6}), \\ R &= \text{diag}(1\times10^{-9}, 1\times10^{-2}), \\ x_{\text{init}} &= [0,0,0,0]^{\mathrm{T}}, \\ \theta_{\text{init}} &= \left[0.2c_{\text{spec,ident}}, 2d_{\text{ident}}\right]^{\mathrm{T}} \\ &= \left[1.79\times10^{3}, 195.32\right]^{\mathrm{T}}. \end{aligned} \tag{32}$$

The resulting parameter estimation with the augmented EKF is shown in Figure 5. The first plot illustrates the behavior of the estimated specific spring constant and the second plot shows the belt damping. The real simulated value is represented by the dashed black line. With the Euler Forward method, the estimates of both parameter are biased. However, due to the feedback, the Kalman Filter can stabilize the unstable discrete-time solution. But, especially the belt damping parameter is highly biased for this scenario (about 8 times higher).

The best result for estimating the spring constant is achieved by the RK4 method (RK3 method is similar, therefore not shown in the picture). The relative error of the mean values of the last two seconds is about 0.58 % and 0.67 % for the RK4 and RK3 method. As seen from the results of the symplectic integrators, it is obvious that the estimates are more accurate than the Heun method. Even the SE method (which has order one) reaches a slightly better estimate in comparison to the Heun method (relative error SE: 3.26 %, Heun: 3.61 %). The relative error of the leapfrog method is about 2.48 %. All relative errors of the specific spring constant are printed in Table 3 marked as E1 (fifth column).

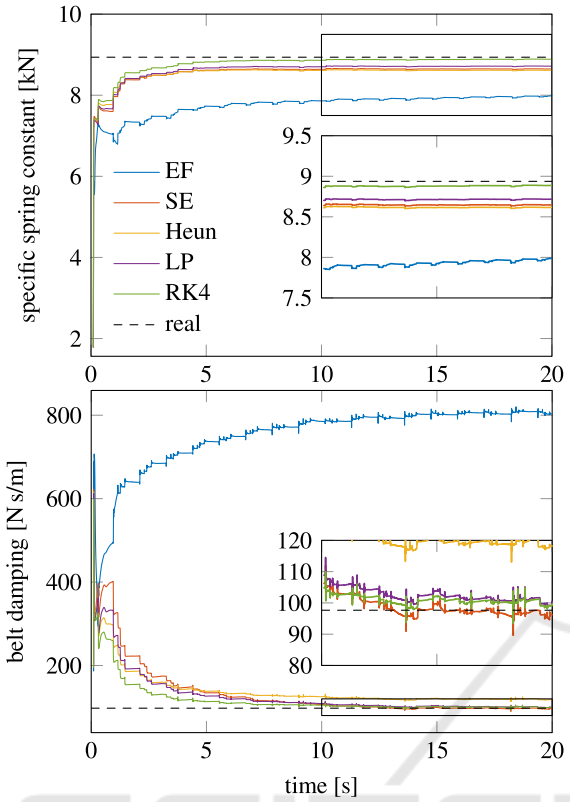


Figure 5: Parameter estimation of augmented EKF.

The belt damping constant is estimated well with the SE, RK4, RK3 and LP method (relative error lower than 5 %, see second picture and sixth column of Table 3). Only the estimated value resulting from the Heun method is higher than 20 % compared to the real value. It should be noted, that is EKF is sub-optimal estimating parameters in the augmented form, so the remaining parameter errors still depend on the choice of covariance matrices. However, this paper only compares the influence of the discretization method to the parameter estimation.

To sum up, estimating the spring constant is possible with all tested discretization methods and the augmented EKF. Even the unstable discrete-time model using EF approach is able to estimate the spring constant with a relative error of about 10 %. Estimating the damping constant is more difficult and the parameter bias of the Euler Forward method is about 8 times higher than the real value. Increasing the discretization order by one (Heun method) leads to better results, but the estimated value is still about 20 % higher than the real value. Only the RK4 (RK3) and the symplectic methods are able to estimate the real damping constant with a relative error lower than 5 %.

However, comparing SE, LP, RK3 and RK4 regarding the computational costs for the augmented

Table 3: Comparison of computing cost and relative parameter estimation error of discretization method with augmented state vector and calculating the Jacobian.

	Mult	Add	Div	E1	E2
EF	29	25	2	10.84	725.32
SE	44	29	2	3.26	1.18
Heun	147	110	4	3.61	21.90
LP	172	122	4	2.48	3.08
RK3	327	251	6	0.67	4.56
RK4	434	329	8	0.58	2.51

EKF (including the dicretization method and calculating its Jacobian), it is obvious, that the evaluation counts, especially for RK3 and RK4, are highly increased (see Table 3). The first reason is the increase of the state vector and the calculation of the Jacobian. The second reason is the more complex calculation of the Jacobian for the symplectic integrators, that's why the computational cost of first order methods differs (compare EF and SE). But, the parameter estimation with the SE approach is much more accurate compared to the EF method, especially for the belt damping. Furthermore, the SE approach reaches even better estimates than the Heun method, which has second discretization order and needs more than 300 % of multiplications. The multiplications needed for the RK3 method are about 740 % higher than the SE method, but both methods estimate the parameter with a relative error lower than 5 %, therefore in the authors opinion the semi implicit Euler approach is the best trade-off between accuracy and computational cost.

5 CONCLUSION

In this paper two symplectic discretization methods are compared to established schemes in the context of online parameter estimation. For the considered mechanical system these symplectic integrators are able to solve the time continuous equations very accurately and with the same (semi implicit Euler) or lower computing cost (leapfrog method). In addition, the classical and commonly used Euler Forward method leads to an unstable discrete-time system, while the SE method with the same numerical evaluations results in a stable discrete-time solution. However, if the highest accuracy is needed, Runge Kutta methods still provide the best results with the drawback of the highest computational cost. Nevertheless, if the mechanical system is poorly damped, the sampling time is fixed and computing cost should be minimal (e.g. for real time implementation), the simplectic integrators are

a possibility to provide adequate discrete-time models. In addition, the simplectic integration methods are scalable, so mechanical systems with more than two degrees of freedom can be calculated.

In the context of online parameter estimation using an augmented EKF, the advantage of symplectic integration methods increases. The SE method outperformed the Euler Forward method, where the estimated parameters are totally biased, while the results of the SE approach are more accurate (relative error lower than 5 %). Furthermore, the estimation results performed by the SE approach reach similar (and higher) accuracy compared to the conventional Runge Kutta methods needing a fraction of computational effort.

REFERENCES

Auger, F., Hilairet, M., Guerrero, J. M., Monmasson, E., and Orlowska-Kowalska, T. (2013). Industrial applications of the Kalman filter: A review. *IEEE Transactions on Industrial Electronics*, 60:5458 – 5471.

Beckmann, D., Schappler, M., Dagen, M., and Ortmaier, T. (2015). New approach using flatness-based control in high speed positioning: Experimental results. In *IEEE International Conference on Industrial Technology (ICIT)*, Sevilla.

Biagiotti, L. and Melchiorri, C. (2008). *Trajectory Planning for Automatic Machines and Robotics*. Springer.

Bohlin, T. (2006). *Practical Grey-box Process Identification*. Springer.

Bohn, C. (2000). *Recursive Parameter Estimation for Nonlinear Continuous-Time Systems through Sensitivity-Model-Based Adaptive Filters*. PhD thesis, Department of Electrical Engineering and Information Sciences, Ruhr-Universität Bochum.

Grewal, M. and Andrews, A. (2008). *Kalman Filtering: Theory and Practice Using MATLAB*. Wiley.

Hairer, E., Lubich, C., and Wanner, G. (2006). *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer.

Ljung, L. (1979). Asymptotic behavior of the extended Kalman filter as a parameter estimator for linear systems. *IEEE Transactions on Automatic Control*, 24(1):36–50.

Nevaranta, N., Parkkinen, J., Lindh, T., Niemela, M., Pyrhonen, O., and Pyrhonen, J. (2015). Online estimation of linear tooth belt drive system parameters. *IEEE Transactions on Industrial Electronics*, 62(11):7214–7223.

Niiranen, J. (1999). Fast and accurate symmetric Euler algorithm for electromechanical simulations. In *Proceedings of 6th. Int. Conf. Electrimacs*, volume 1, pages 71 – 78, Lisboa, Portugal.

Öltjen, J., Kotlarski, J., and Ortmaier, T. (2015). Reduction of end effector oscillations of a parallel mechanism with modified motion profiles. In *Proceedings of IEEE 10th Conference on Industrial Electronics and Applications (ICIEA)*, pages 823–829.

Riva, M. H., Beckmann, D., Dagen, M., and Ortmaier, T. (2015). Online parameter and process covariance estimation using adaptive EKF and SRCuKF approaches. In *IEEE Multi-Conference on Systems and Control (MSC 2015)*, Sydney, Australia.

Schütte, F., Beineke, S., Rolfsmeier, A., and Grotstollen, H. (1997). Online identification of mechanical parameters using extended Kalman filter. In *Proceedings of IEEE Annual Meeting of Industry Applications Society*.

Szabat, K. and Orlowska-Kowalska, T. (2012). Application of the Kalman Filters to the High-Performance Drive System With Elastic Coupling. *IEEE Transactions on Industrial Electronics*, 59(11):4226 – 4235.

Welch, G. and Bishop, G. (2006). An introduction to the Kalman filter. Technical report, Chapel Hill, NC, USA.