

Rauch-Tung-Striebel Smoother for Position Estimation of Short-Stroke Reluctance Actuators

Eduardo Moya-Lasheras, Jan M. Schellekens, and Carlos Sagües

Abstract—This paper presents a novel state estimator for short-stroke reluctance actuators, intended for soft-landing control applications in which the position cannot be measured in real time. One of the most important contributions regards the system modeling for the estimator. The discrete state of the hybrid system is treated as an input. Moreover, the model is simplified to facilitate the identification of parameters and the implementation of the estimator. Thus, auxiliary variables are added to the state vector in order to indirectly account for modeling errors. Another important contribution is the state estimation approach. It is based on the Rauch-Tung-Striebel fixed-interval smoother, which allows refining past data from later observations. Numerous simulations are performed to analyze and compare the proposal and several alternatives. In addition, experimental testing is presented to evaluate and validate the estimator. As the simulated and experimental analyses demonstrate, the combined effect of the novel additions results in significantly smaller estimation errors of position and velocity.

I. INTRODUCTION

Reluctance actuators are a type of electromagnetic devices in which the positioning is mainly governed by magnetic reluctance forces. They are used in a wide range of applications due to their high force densities and low heat dissipation. For switch-type electromagnetic devices, small short-stroke single-coil reluctance actuators are preferable over other solutions because of their relatively low cost. However, in many cases, they are discarded because of the strong impacts during commutations. In consequence, there is a great interest in achieving soft landing when switching in order to reduce bouncing and clicking noise, lengthen their service life, and make them suitable for a wider range of applications.

In numerous publications, soft-landing controllers have been presented for reluctance actuators. Several authors propose tracking the position with nonlinear feedback controllers based on feedback linearization [1], [2], [3], or sliding-mode techniques [4], [5], among other alternatives. To circumvent the need of measuring or estimating the position in real time, some works focus on designing open-loop control methods based on optimal control [6], [7], or flatness-based feedforward controllers [8]. In order to improve the robustness of the

This work was supported in part by the Ministerio de Economía y Competitividad, Gobierno de España - European Union under project RTC-2017-5965-6 of subprogram Retos-Colaboración, in part by Aragón ERDF 2014-20 Operational Programme, in part by project DGA-T45_17R/FSE, and in part by project PGC2018-098719-B-I00 (MCIU/AEI/FEDER, UE). (Corresponding author: Eduardo Moya-Lasheras.)

E. Moya-Lasheras and C. Sagües are with the Departamento de Informática e Ingeniería de Sistemas (DIIS) and the Instituto de Investigación en Ingeniería de Aragón (I3A), Universidad de Zaragoza, Zaragoza 50018, Spain (email: emoya@unizar.es; csagues@unizar.es).

J. M. Schellekens is with the Department of Electrical Engineering, Eindhoven University of Technology, 5612 AZ Eindhoven, The Netherlands (e-mail: jschellekens@tue.nl).

controllers against estimation, measurement, modeling or other errors, some proposals exploit the repetitive operations with cycle-to-cycle adaptations of feedback [9], open-loop [10] or feedforward [11], [12], [13] controllers.

Most of the proposed soft-landing solutions use some kind of position feedback loop. However, measuring the position during motion is impractical in many scenarios, especially when using low-cost reluctance actuators. In those cases, position sensors tend to be too expensive in relation to the devices themselves. Alternatively, the position may be estimated from other measurements. Different estimation techniques have been designed for reluctance actuators. One approach is estimating the inductance of the coil from the electrical measurements, and then relating it to a certain position using an inverse model [14] or a look-up table [15]. A similar idea is calculating the position from an estimated magnetic flux and a measured current [16]. These techniques are open loop, and rely solely on the characterization of the inductance—or magnetic flux—for every position and current. Therefore, they are very sensitive to any errors in that mapping. Furthermore, generating bijective maps between position and inductance—or flux—is only possible when neglecting certain phenomena, e.g. eddy currents and magnetic hysteresis. Then, some proposals directly account for the motion dynamics by including a model of the mechanical subsystem. These are based on traditional state observers, such as sliding-mode observers [17], or Kalman filter extensions for nonlinear systems [18], [19]. Nonetheless, these observers still neglect certain electromagnetic phenomena, most notably the magnetic hysteresis. Still, there are examples of position estimators that take into account this complex phenomenon using heuristic models. For example, [20] proposes an open-loop estimator based on a Preisach model.

Nevertheless, designing position estimators for low-cost reluctance actuators is still a very challenging problem. Ideally, they should be based on a complex and accurately identified model. However, as their dynamics are very fast (commutations typically last few milliseconds), the implementation cost of these complex estimators with a high enough sampling rate is usually prohibitive. Additionally, simplified models are much easier to identify. This is especially relevant for cost-effective manufacturing processes in which each ensemble of actuators has a substantial unit-to-unit variability, which, in turn, implies that model parameters must be estimated or adapted for every single device.

Then, this paper proposes a novel position estimator for reluctance actuators, introducing three ideas that have not been previously explored for these devices. First, the state estimation is approached as a smoothing problem of a stochastic process, in which the state at a given time is refined by

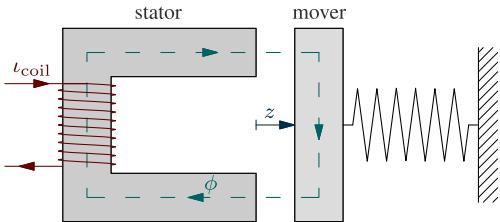


Fig. 1. Schematic representation of a single-coil reluctance actuator.

using future observation samples. Although smoothers are not causal, they are more precise than their filter counterparts. Besides, they can be formulated and used in repetitive operations, as in the cycle-to-cycle learning strategies commonly proposed for many real devices, such as the reluctance actuators under study. The second main contribution is the purposely simplified estimator model. On the one hand, it neglects important electromagnetic phenomena—especially magnetic hysteresis—to make it simpler and more efficient. On the other hand, it incorporates auxiliary variables in order to account for modeling errors during the estimation process. Regarding the third main contribution of this paper, the estimator model is designed assuming a novel set of observable variables. In addition to the electrical signals, commonly considered as the input and output of the system, the proposed estimator uses discrete information related to its state, in particular whether the armature is resting at one of the contacts or moving.

The paper is structured as follows. Section II describes the actuators through a generalized dynamical model and presents its adaptation for state estimation. Then, III describes the estimation algorithm. In Section IV, simulated and experimental results are presented to analyze the proposal and compare it with other alternatives. Finally, the conclusions are included in Section V.

II. ACTUATOR DESCRIPTION

The class of actuators under study is schematically represented in Fig. 1. It is essentially a magnetic core divided into a mover and a stator. The current through the coil generates a magnetic flux through the core parts and the air gaps between them, which results in a magnetic force that tends to attract the mover toward the stator. Single-coil actuators cannot generate magnetic forces in the opposite direction, so they require passive forces to separate the mover from the stator (e.g. elastic forces).

A. Generalized dynamics

The dynamical system is divided into two coupled parts: the electromagnetic and the mechanical subsystems. Regarding the former, the dynamics of the magnetic flux ϕ is given by the electrical circuit equation,

$$v_{\text{coil}} = R \iota_{\text{coil}} + N \dot{\phi}, \quad (1)$$

where v_{coil} is the voltage between the coil terminals, ι_{coil} is the current through the coil, R is the internal resistance of the coil, and N is the number of coil turns. Additionally, the magnetic flux is related to the current given Ampere's circuital law. It

states that the integrated magnetic field strength H through a path $\delta\Sigma$ is equal to the current passing through its enclosed surface Σ . Choosing $\delta\Sigma$ as the path of the magnetic flux, as presented in Fig. 1, the derived relation is

$$\int_{\delta\Sigma_{\text{gap}}} H \, dl + \int_{\delta\Sigma_{\text{core}}} H \, dl = N \iota_{\text{coil}} + \iota_{\text{eddy}}, \quad (2)$$

where the core $\delta\Sigma_{\text{core}}$ and gap $\delta\Sigma_{\text{gap}}$ paths are integrated separately. Moreover, note that the total current includes ι_{eddy} , which represents the net eddy current generated in the core. To model the eddy currents, it is assumed that the magnetic flux is uniform within the cross section of the core. This first-order approximation results in a current proportional to the time derivative of the magnetic flux,

$$\iota_{\text{eddy}} = -k_{\text{eddy}} \dot{\phi}, \quad (3)$$

in which k_{eddy} is a constant that depends on the geometry and conductivity of the core [21].

With respect to the magnetic field strength H , it can be directly related to the magnetic flux through Hopkinson's law,

$$\mathcal{R} = \frac{\int H \, dl}{\phi} = \int \frac{dl}{\mu A}, \quad (4)$$

where \mathcal{R} is the magnetic reluctance, A is the cross-sectional area, and μ is the magnetic permeability of the air. Then, for the air gap, the integral is simplified to

$$\int_{\delta\Sigma_{\text{gap}}} H \, dl = \mathcal{R}_g(z) \phi. \quad (5)$$

where \mathcal{R}_g is the magnetic reluctance of the gap, which is a function of the position of the mover z .

On the other hand, the relation between ϕ and H in the core presents a hysteresis. Thus, it cannot be accurately modeled with a direct mapping as in Hopkinson's law. For now, the integral expression is simplified to

$$\int_{\delta\Sigma_{\text{core}}} H \, dl = H_c l_c, \quad (6)$$

where H_c is the average magnetic field strength across the core length l_c .

Then, by substituting equations (1), (3), (5) and (6) into (2), the current ι_{coil} is expressed as a function of the other variables,

$$\begin{aligned} \iota_{\text{coil}} &= f_t(z, \phi, H_c, v_{\text{coil}}) \\ &= \frac{N(\mathcal{R}_g(z)\phi + H_c l_c)}{N^2 + R k_{\text{eddy}}} + \frac{k_{\text{eddy}}}{N^2 + R k_{\text{eddy}}} v_{\text{coil}}. \end{aligned} \quad (7)$$

The input of the dynamical system is the voltage v_{coil} , whereas z , ϕ and H_c are variables whose dynamics should be defined. First, the magnetic flux derivative can be obtained directly from (1) and (7),

$$\dot{\phi} = f_\phi(z, \phi, H_c, u) = \frac{v_{\text{coil}} - R f_t(\cdot)}{N}. \quad (8)$$

Secondly, in order to characterize the dynamics of the magnetic field strength in the core, magnetic hysteresis and saturation should be taken into account. Due to its complexity, magnetic hysteresis is most commonly ignored in this class

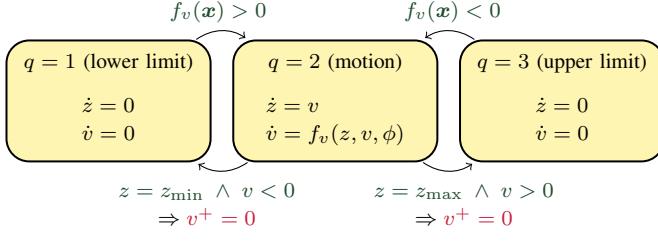


Fig. 2. Diagram of the hybrid automaton modeling the mechanical system.

of actuators, even if its effect is not negligible. Still, there are some proposals for integrating magnetic hysteresis models into the dynamic system of reluctance actuators [22], [23]. Following the approach from [23], the differential permeability $\mu_\Delta = dB_c/dH_c$ —which provides the relation between the core magnetic flux density B_c and field strength H_c —can be expressed as a piecewise function of ϕ , H_c and $\dot{\phi}$ (with different behavior for increasing and decreasing magnetic flux),

$$\mu_\Delta(\phi, H_c, \dot{\phi}) = \begin{cases} \mu_\Delta^+(H_c, \phi), & \text{if } \dot{\phi} \geq 0 \\ \mu_\Delta^-(H_c, \phi), & \text{if } \dot{\phi} < 0 \end{cases}. \quad (9)$$

Then, the differential equation of H_c is defined following the same form as (8),

$$\dot{H}_c = f_{H_c}(z, \phi, H_c, u) = \frac{f_\phi(\cdot)}{A_c \mu_\Delta(\phi, H_c, f_\phi(\cdot))}. \quad (10)$$

Regarding the mechanical subsystem, there are two state variables: the mover position z and its velocity v . The subsystem is modeled as a mass-spring-damper in which the external force is the magnetic force. Then, the differential equation of the velocity is defined as follows:

$$\dot{v} = f_v(z, v, \phi) = \frac{1}{m} (k_{sp}(z_{sp} - z) - c_f v + F_{mag}(z, \phi)), \quad (11)$$

being k_{sp} the spring stiffness, z_{sp} the spring resting position, c_f the damping coefficient, and F_{mag} the magnetic force, which is a function of the position and magnetic flux [24],

$$F_{mag}(z, \phi) = -\frac{1}{2} \mathcal{R}'_g(z) \phi^2, \quad \mathcal{R}'_g(z) = \frac{\partial \mathcal{R}_g(z)}{\partial z}. \quad (12)$$

Note that the stroke of these actuators is limited. Thus, the position dynamics must be static when reaching the lower ($z = z_{min}$) or upper limit ($z = z_{max}$). This is achieved by representing the dynamics of z and v with a hybrid automaton, as depicted in Fig. 2. The discrete state is $q \in \{1, 2, 3\}$, which designates the dynamic mode of the system. Each transition between two modes is accompanied by its guard condition. In the case of transitioning to $q = 1$ or $q = 3$, there is also a reset function: $v^+ = 0$.

B. Estimator model

The previous dynamical model is very accurate. It is also computationally efficient compared to other models of similar accuracy (i.e. including magnetic hysteresis, saturation and eddy currents). However, it is still quite complex, with many

parameters that need to be identified. Thus, introducing some simplifications to the model gives relevant advantages for the position estimation.

To reduce the number of parameters, auxiliary parameters and variables are used. First, the position z and velocity v are normalized,

$$z^* = \frac{z - z_{min}}{l_z}, \quad v^* = \frac{v}{l_z}, \quad (13)$$

where $l_z = z_{max} - z_{min}$. Thus, the new position variable is bounded such that $z^* \in [0, 1]$. Secondly, the flux linkage $\lambda = N\phi$ is used as the third state variable. Then, the current function is simplified by neglecting both magnetic hysteresis and eddy currents,

$$\iota_{coil} = f_\iota(z^*, \lambda, e_\iota) = (\mathcal{R}_g^*(z^*) + \mathcal{R}_c^*(\lambda)) \lambda + e_\iota, \quad (14)$$

where the auxiliary function $\mathcal{R}_g^*(z^*) = \mathcal{R}_g(z)/N^2$ is an scaled reluctance of the gap and $\mathcal{R}_c^*(\lambda)$ is an scaled reluctance of the core. It is modeled based on Fröhlich–Kennelly relation, which takes into account the magnetic saturation,

$$\mathcal{R}_c^*(\lambda) = \frac{\mathcal{R}_{c,0}^*}{1 - \lambda/\lambda_{sat}}, \quad (15)$$

where the constant $\mathcal{R}_{c,0}^*$ is the value of the scaled reluctance when $\lambda = 0$, and λ_{sat} is the saturation value of λ . Note also the addition of e_ι , which considers the modeling errors of the current function. This was previously proposed for an Unscented Kalman Filter [19].

On the other hand, the acceleration is given by the following function,

$$\begin{aligned} \dot{v}^* &= f_v^*(z, v, \lambda, e_F) \\ &= \frac{1}{m^*} \left(k_{sp}^* (z_{sp}^* - z) - c_f^* v^* - \frac{1}{2} \mathcal{R}'_g^*(z^*) \lambda^2 + e_F \right), \end{aligned} \quad (16)$$

where the auxiliary parameters and functions are

$$\begin{aligned} m^* &= m l_z^2, & k_{sp}^* &= k_{sp} l_z^2, & c_f^* &= c_f l_z^2, \\ z_{sp}^* &= (z_{sp} - z_{min})/l_z, & \mathcal{R}'_g^*(z^*) &= \mathcal{R}'_g(z) l_z/N^2, \end{aligned} \quad (17)$$

and the variable e_F is another error term. Note that f_v^* is only valid during motion, so e_F acts as the normal force when the mover is in contact with one of the limits. It is also useful for compensating for modeling errors.

Given the newly defined functions, we express the dynamic system with a discrete-time nonlinear model,

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k, \\ \mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{v}_k, \end{cases} \quad (18)$$

where \mathbf{x}_k , \mathbf{u}_k and \mathbf{y}_k are the state, input and output vectors; \mathbf{w}_k and \mathbf{v}_k are the process and observation noise vectors; and \mathbf{f} and \mathbf{h} are the transition and observation functions. The noise terms \mathbf{w}_k and \mathbf{v}_k are assumed to be independent zero-mean random vectors. Note that the errors e_ι and e_F may be considered part of those noise terms, but treating them as white noise is a very poor approximation. Instead, they are also incorporated into the state vector,

$$\mathbf{x}_k = [z_k^* \ z_{k-1}^* \ \lambda_k \ e_{\iota,k} \ e_{F,k}]^\top. \quad (19)$$

Note also that the previous position z_{k-1}^* replaces the velocity as the second state variable. This allows the numerical resolution of the motion dynamics using Verlet integration [25].

Moreover, the addition of errors in the state vector may seem comparable to an error-state formulation of a Kalman filter, most commonly applied to inertial navigation systems [26]. However, there are very important differences between that state-of-the-art formulation and the proposal. The proposed auxiliary state variables account for errors in the transition and output functions, instead of errors in the nominal state. While the state error formulation is intended to mitigate some linearization issues of nonlinear extensions of Kalman filters, the proposed function errors are formulated to circumvent issues caused by modeling simplifications and inaccuracies.

Then, for clarity purposes, the components of \mathbf{f} and \mathbf{w}_k are represented based on their corresponding state variables,

$$\mathbf{f}(\cdot) = \begin{bmatrix} \mathbf{f}_{z^*}(z_k^*, z_{k-1}^*, \lambda_k, e_{F,k}, q_k) \\ \mathbf{f}_{\lambda}(z_k^*, \lambda_k, e_{\iota,k}, v_{\text{coil},k}) \\ \mathbf{f}_{e_\iota}(e_{\iota,k}) \\ \mathbf{f}_{e_F}(e_{F,k}) \end{bmatrix}, \quad \mathbf{w}_k = \begin{bmatrix} w_{z^*,k} \\ w_{z_{k-1}^*,k} \\ w_{\lambda,k} \\ w_{e_{\iota,k}} \\ w_{e_{F,k}} \end{bmatrix}. \quad (20)$$

To improve the accuracy of the estimator, we propose to use not only the voltage and current, but also the contact information. In that case, the input of the estimator may consist of the voltage and the discrete state of the hybrid automaton (see Fig. 2),

$$\mathbf{u}_k = [v_{\text{coil},k} \quad q_k]^\top. \quad (21)$$

The first element of the state function \mathbf{f} corresponds to the position transition. The discretization is performed using Verlet integration where, in the case of $q_k = 1$ or $q_k = 3$, z_k^* and v_k^* are replaced by their real values. Consequently, \mathbf{f}_{z^*} is defined as a piece-wise function,

$$\mathbf{f}_{z^*}(\cdot) = \begin{cases} 0 + T_s^2 f_v^*(0, 0, \lambda_k, e_{F,k}), & \text{if } q_k = 1 \\ 2z_k^* - z_{k-1}^* + T_s^2 f_v^*(z_k^*, v_k^*, \lambda_k, e_{F,k}), & \text{if } q_k = 2, \\ 1 + T_s^2 f_v^*(1, 0, \lambda_k, e_{F,k}), & \text{if } q_k = 3 \end{cases}, \quad (22)$$

where T_s is the sampling period. Note that, although the velocity is not treated as a state variable in this discrete form, it is still required for computing the acceleration by means of the function f_v^* (16). Thus, the velocity v_k^* is calculated as the average velocity in the previous time interval,

$$v_k^* = \frac{z_k^* - z_{k-1}^*}{T_s}. \quad (23)$$

Note that, as the modeling and discretization errors are accounted for by $e_{F,k}$, there is no need for an additional process noise, i.e. $w_{z^*,k} = 0$, for all k .

The main advantage of the chosen integration method is its simplicity. Specifically, the transition of the previous position is trivial,

$$\mathbf{f}_{z_{k-1}^*}(\cdot) = z_k^*, \quad w_{z_{k-1}^*,k} = 0, \quad \forall k. \quad (24)$$

The third element of the transition function (\mathbf{f}) of the corresponds to the flux linkage, which is defined using the forward Euler method,

$$\mathbf{f}_\lambda(\cdot) = \lambda_k + T_s (v_{\text{coil},k} + R f_\iota^*(z_k^*, \lambda_k, e_{\iota,k})). \quad (25)$$

Analogously to the continuous model, the current is not treated as a state variable, but its value is required for the transition of the flux linkage and calculated via the auxiliary function f_ι^* . While the modeling errors in the current function are already considered through the state variable $e_{\iota,k}$, the truncation errors of the transition function \mathbf{f}_λ are accounted for by the process noise $w_{\lambda,k} \neq 0$.

Regarding the auxiliary variables $e_{\iota,k}$ and $e_{F,k}$, there is no prior information about their dynamics. The best possible assumption for a generalized case, without requiring more model parameters or state variables, is that their expectations are equal to the previous ones,

$$\mathbf{f}_{e_\iota}(\cdot) = e_{\iota,k}, \quad \mathbf{f}_{e_F}(\cdot) = e_{F,k}. \quad (26)$$

Then, the errors introduced by this assumption are accounted for by their process noises $w_{e_{\iota,k}}, w_{e_{F,k}} \neq 0$.

The estimator requires comparing the output of the model with the measured one, and then correcting the state vector accordingly. During motion ($q_k = 2$), the only output is the current. On the other hand, if $q_k = 1$ or $q_k = 3$, the contact information can also be exploited to correct the estimated position. Thus, the observed output vector is

$$\mathbf{y}_k = \begin{cases} [\iota_{\text{coil},k} \quad 0]^\top, & \text{if } q_k = 1 \\ \iota_{\text{coil},k}, & \text{if } q_k = 2, \\ [\iota_{\text{coil},k} \quad 1]^\top, & \text{if } q_k = 3 \end{cases} \quad (27)$$

Correspondingly, the model-based output function must be

$$\mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) = \begin{cases} f_\iota^*(\cdot), & \text{if } q_k = 2 \\ [f_\iota^*(\cdot) \quad z_k^*]^\top, & \text{if } q_k \neq 2, \end{cases} \quad (28)$$

and the observation noise must be

$$\boldsymbol{\nu}_k = \begin{cases} \nu_{\iota,k}, & \text{if } q_k = 2 \\ [\nu_{\iota,k} \quad 0]^\top, & \text{if } q_k \neq 2, \end{cases} \quad (29)$$

where $\nu_{\iota,k}$ is the observation error of the current ι_{coil} (not to be confused with the modeling error $e_{\iota,k}$). It is noteworthy that the dimension of the output terms depends on the input. Apart from that unconventional aspect, it is a standard discrete-time model that can be used to design estimators.

III. STATE ESTIMATOR

Bayesian theory provides a powerful tool for the state estimation of stochastic systems. In particular, Bayesian smoothing is a class of recursive Bayesian estimation in which the state at a given time is reconstructed by using future observation samples [27]. In order to rigorously solve a smoothing problem, the complete probability distributions of the state may be recursively calculated through numerical integration. However, this approach is generally intractable in control applications. Therefore, it is necessary to simplify the Bayesian estimation problem and reduce its computational complexity.

Specifically, we propose to use an extension of the Rauch–Tung–Striebel smoother [28] for nonlinear systems. It is a fixed-interval smoother, suitable for estimating the switching operations of reluctance actuators because their duration is bounded. This estimator consists of two processes: a forward and a backward recursion.

A. Forward pass

The first part of the algorithm is equivalent to an extended Kalman Filter (EKF). It is arguably the most popular stochastic observer for nonlinear systems, due to its simplicity and performance. It may be interpreted as a simplified version of the Bayesian filtering technique, in which the probability distributions are assumed to be Gaussian,

$$\mathbf{x}_k | \mathbf{u}_0, \mathbf{y}_0, \dots, \mathbf{u}_k, \mathbf{y}_k \approx \mathbf{x}_{k|k}^{\text{EKF}} \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k}), \quad (30)$$

and the transition and output functions are linearized,

$$\mathbf{F}_k = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \Big|_{\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}}, \quad \mathbf{H}_k = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \Big|_{\bar{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_k}. \quad (31)$$

The partial derivatives can be solved symbolically, but note that the derived expressions depend on the scaled gap reluctance \mathcal{R}_g^* , and its derivatives \mathcal{R}'_g^* and \mathcal{R}''_g^* . If \mathcal{R}_g^* is approximated with an algebraic expression, it must be twice differentiable with respect to the position z^* . If, instead, look-up tables are used, the derivatives must be included.

The algorithm for the forward filter is divided into three steps: prediction (*a priori*), update (*a posteriori*) and projection (to guarantee state constraints).

- 1) **Prediction.** The k th *a priori* state estimation $(\bar{\mathbf{x}}_{k|k-1}, \mathbf{P}_{k|k-1})$ is derived from the $(k-1)$ th projected *a posteriori* state estimation $(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{P}_{k-1|k-1})$,

$$\bar{\mathbf{x}}_{k|k-1} = \mathbf{f}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}), \quad (32)$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^\top + \mathbf{Q}, \quad (33)$$

being \mathbf{Q} the covariance of the process noise \mathbf{w}_k .

- 2) **Update.** The *a priori* prediction is refined by taking into account the observation \mathbf{y}_k . The updated estimation is the *a posteriori* state $(\bar{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})$,

$$\bar{\mathbf{x}}_{k|k} = \bar{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{y}_k - \mathbf{h}(\bar{\mathbf{x}}_{k|k-1}, \mathbf{u}_k)), \quad (34)$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{H}_k \mathbf{P}_{k|k-1}, \quad (35)$$

where \mathbf{K}_k is the Kalman gain. It is defined as

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^\top (\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^\top + \mathbf{R}_k)^{-1}, \quad (36)$$

being \mathbf{R}_k the covariance of the observation noise \mathbf{v}_k . Note that $\mathbf{R}_k \in \mathbb{R}^{1 \times 1}$ or $\mathbf{R}_k \in \mathbb{R}^{2 \times 2}$, depending on q_k (29).

- 3) **Projection.** The state estimation must be corrected when it is outside bounds. There are multiple ways of enforcing linear constraints in Kalman filters [29]. For this problem, a very simple and effective approach is to project the mean state vector based on its covariance. To obtain the maximum likelihood estimate of the state subject to inequality state constraints, the problem is formulated as

$$\hat{\mathbf{x}}_{k|k} = \arg \min_{\mathbf{x}_k} (\mathbf{x}_k - \bar{\mathbf{x}}_{k|k})^\top \mathbf{P}_{k|k}^{-1} (\mathbf{x}_k - \bar{\mathbf{x}}_{k|k}) \quad (37a)$$

$$\text{subject to } \mathbf{A} \mathbf{x}_k \leq \mathbf{b}. \quad (37b)$$

In general, this is a quadratic programming problem. In our case, only the position needs to be constrained between 0 and 1, which means that

$$\mathbf{A} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ +1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \quad (38)$$

This is easily solved because both constraints—corresponding to each row of \mathbf{A} and \mathbf{b} —cannot be active at the same time. Thus, it can be reformulated as an equality constrained quadratic programming problem,

$$\hat{\mathbf{x}}_{k|k} = \arg \min_{\mathbf{x}_k} (\mathbf{x}_k - \bar{\mathbf{x}}_{k|k})^\top \mathbf{P}_{k|k}^{-1} (\mathbf{x}_k - \bar{\mathbf{x}}_{k|k}) \quad (39a)$$

$$\text{subject to } \mathbf{A}_k \mathbf{x}_k = b_k, \quad (39b)$$

where \mathbf{A}_k and b_k correspond to the active constraint,

$$\mathbf{A}_k = \begin{cases} [-1 \ 0 \ 0 \ 0 \ 0], & \text{if } \bar{z}_{k|k}^* < 0 \\ \mathbf{0}_{0 \times 5}, & \text{if } 0 \leq \bar{z}_{k|k}^* \leq 1, \\ [+1 \ 0 \ 0 \ 0 \ 0], & \text{if } \bar{z}_{k|k}^* > 1 \end{cases} \quad (40)$$

$$b_k = \begin{cases} 0, & \text{if } \bar{z}_{k|k}^* < 0 \\ \mathbf{0}_{0 \times 1}, & \text{if } 0 \leq \bar{z}_{k|k}^* \leq 1, \\ 1, & \text{if } \bar{z}_{k|k}^* > 1 \end{cases} \quad (41)$$

where the position $\bar{z}_{k|k}^*$ is the first element of the state vector $\bar{\mathbf{x}}_{k|k}$. The problem can be solved algebraically, resulting in the following projected state estimation:

$$\hat{\mathbf{x}}_{k|k} = \bar{\mathbf{x}}_{k|k} - \mathbf{P}_{k|k} \mathbf{A}_k^\top (\mathbf{A}_k \mathbf{P}_{k|k} \mathbf{A}_k^\top)^{-1} (\mathbf{A}_k \bar{\mathbf{x}}_{k|k} - b_k). \quad (42)$$

To simplify the algorithm and facilitate its implementation, the covariance matrix is not updated after the projection. It is a conservative approximation because the updated covariance matrix would be smaller.

B. Backward pass

The complete algorithm is the Extended Rauch–Tung–Striebel Smoother (ERTSS). Equivalently to the EKF, the ERTSS is an approximated version of the recursive Bayesian smoothing [30], in which the transition function is linearized (31), and the probability distributions are assumed to be Gaussian,

$$\mathbf{x}_k | \mathbf{u}_0, \mathbf{y}_0, \dots, \mathbf{u}_{k_f}, \mathbf{y}_{k_f} \approx \mathbf{x}_{k|k_f}^{\text{ERTSS}} \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k_f}, \mathbf{P}_{k|k_f}), \quad (43)$$

where k_f represents the final sample of the smoothed interval.

The algorithm includes the previous steps from the forward pass and two additional ones that constitute the backward pass:

- 4) **Smoothing:** As the chosen notation already suggests, the filtered estimation at $k = k_f$ is equal to the smoothed one $(\hat{\mathbf{x}}_{k_f|k_f}, \mathbf{P}_{k_f|k_f})$. Then, starting at k_f , past states are smoothed through a backwards recursion,

$$\bar{\mathbf{x}}_{k|k_f} = \hat{\mathbf{x}}_{k|k} + \mathbf{G}_k (\hat{\mathbf{x}}_{k+1|k_f} - \bar{\mathbf{x}}_{k+1|k}), \quad (44)$$

$$\mathbf{P}_{k|k_f} = \mathbf{P}_{k|k} + \mathbf{G}_k (\mathbf{P}_{k+1|k_f} - \mathbf{P}_{k+1|k}) \mathbf{G}_k^\top, \quad (45)$$

where \mathbf{G}_k is the smoother gain,

$$\mathbf{G}_k = \mathbf{P}_{k|k} \mathbf{F}_{k+1}^\top \mathbf{P}_{k+1|k}^{-1}. \quad (46)$$

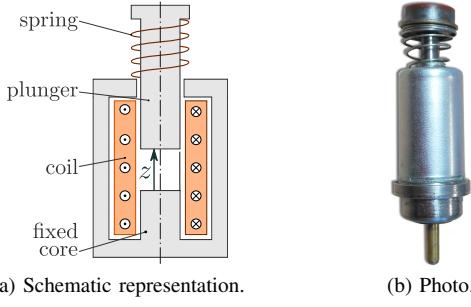


Fig. 3. Plunger-type solenoid valve used to identify the model parameters.

TABLE I
PARAMETERS OF THE COMPLETE MODEL

Parameter	Value	Parameter	Value
m	1.6×10^{-3} kg	k_{eddy}	$1630 \Omega^{-1}$
k_{sp}	61.8 N/m	l_c	0.055 m
z_{sp}	0.0192 m	A_c	$1.26 \times 10^{-5} \text{ m}^2$
c_f	0.8 N s/m	M_{sat}	$1.45 \times 10^6 \text{ A/m}$
z_{\min}	4×10^{-4} m	b	2.45×10^{-3} T
z_{\max}	1.4×10^{-3} m	c	0.736
N	1200	κ	942 A/m
R	48 Ω	α	3.66×10^{-3}

Notice that this process requires the state estimates from the forward pass for all k , so they should be stored or recalculated.

- 5) **Projection:** Equivalently to the third step in the EKF, the smoothed state estimation is projected to satisfy the state constraints. The reasoning is equivalent, so this is performed using (40) to (42), but replacing $\hat{x}_{k|k}$, $\bar{x}_{k|k}$, $P_{k|k}$ and $\bar{z}_{k|k}^*$ with $\hat{x}_{k|k_f}$, $\bar{x}_{k|k_f}$, $P_{k|k_f}$ and $\bar{z}_{k|k_f}^*$, respectively.

The purpose of the smoothing backward recursion is to correct past estimates. Note that it must be computed offline, so it cannot be directly used for feedback control. Nonetheless, it can be exploited for controlling switch-type actuators due to the repetitive operations. Some possible applications are cyclic adaptation strategies of real-time controllers or observers.

IV. ANALYSES

In this section, the performance of the proposed estimator is analyzed via simulations.

A. Reference simulations

The continuous-time model presented in Section II-A includes the most important electromagnetic phenomena. As shown in [23], it matches real measurements with high accuracy. Thus it is considered the reference, emulating the real system, for analysis purposes. The parameters have been estimated through an identification process of a linear travel solenoid valve (see Fig. 3), and their values are presented in Table I. Furthermore, the gap reluctance and its derivatives for different positions are stored in look-up tables, and depicted in Fig. 4.

In Fig. 5, we present an ideal solution of the soft-landing problem for a making (closing) and breaking (opening) cycle, based on the corresponding position trajectories. Note that, the

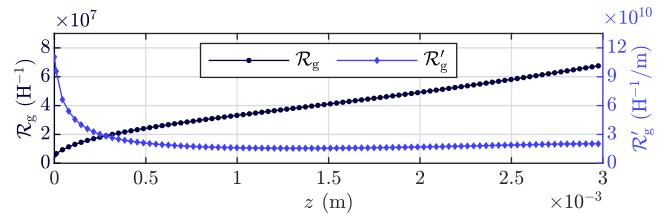


Fig. 4. Gap reluctance and its derivative with respect to the position.

TABLE II
PARAMETERS OF THE REDUCED MODEL

Parameter	Value	Parameter	Value
m^*	$1.6 \times 10^{-9} \text{ kg m}^2$	c_f^*	$8 \times 10^{-7} \text{ N ms}$
k_{sp}^*	$6.18 \times 10^{-8} \text{ N m}$	$\mathcal{R}_{c,0}^*$	1.75 H^{-1}
z_{sp}^*	18.8	λ_{sat}	0.0238 Wb

trajectory design is a critical aspect of the soft-landing control of reluctance actuators. Some works focus on finding feasible solutions [8], [31], [32], while others propose to optimize some particular variables, e.g. transition time [6], mean power consumption [33], or expected contact velocities [7]. In this paper, the position trajectories for both operations are designed with fifth-degree polynomials (see Fig. 5a), but other solutions may also be valid. As boundary conditions the initial and final positions are set to z_{\max} or z_{\min} (depending on the operation type), while the velocities and accelerations are set to zero. The motion duration for each operation is set to 4 ms. Then, Figs. 5b, 5d and 5c depict the required magnetic flux, coil voltage and coil current signals, respectively, for moving the actuator as desired. Note that there is an interval before the start of motion (at $t = 1$ ms and $t = 11$ ms) that allows the coil to energize or de-energize until the magnetic force manages to compensate the spring force. Then, after the landing (at $t = 5$ ms and $t = 15$ ms), the voltage is set to a constant value which guarantees that the mover is fixed at the desired position. Finally, Fig. 5e shows the ϕ - H_c hysteresis curve. As can be seen, this phenomenon is not negligible. The lack of a proper hysteresis model in the estimator model will be a source of estimation errors even when the rest of parameters fit perfectly the ones from the reference model.

The simulations are used to set the estimator constants and analyze the proposal. The parameters of the estimator model (Section II-B) are fitted to the simulated signals and presented in Table II. Moreover, the state vector of the discretized estimator model \mathbf{x}_k (19) is derived as a reference for analysis purposes. To this end, the signals are discretized such that $t_{k+1} = t_k + T_s$, where the chosen sampling period is set as $T_s = 20 \mu\text{s}$. Then, the estimator state variables are calculated based on their definitions,

$$z_k^* = (z(t_k) - z_{\min}) / (z_{\max} - z_{\min}), \quad (47a)$$

$$\lambda_k = N \phi(t_k), \quad (47b)$$

$$e_{\ell,k} = \iota_{\text{coil},k} - f_\ell(z_k^*, \lambda_k, 0, v_{\text{coil},k}), \quad (47c)$$

$$e_{F,k} = m^* (z_{k+1}^* - f_{z^*}(z_k^*, z_{k-1}^*, \lambda, 0, q_k)). \quad (47d)$$

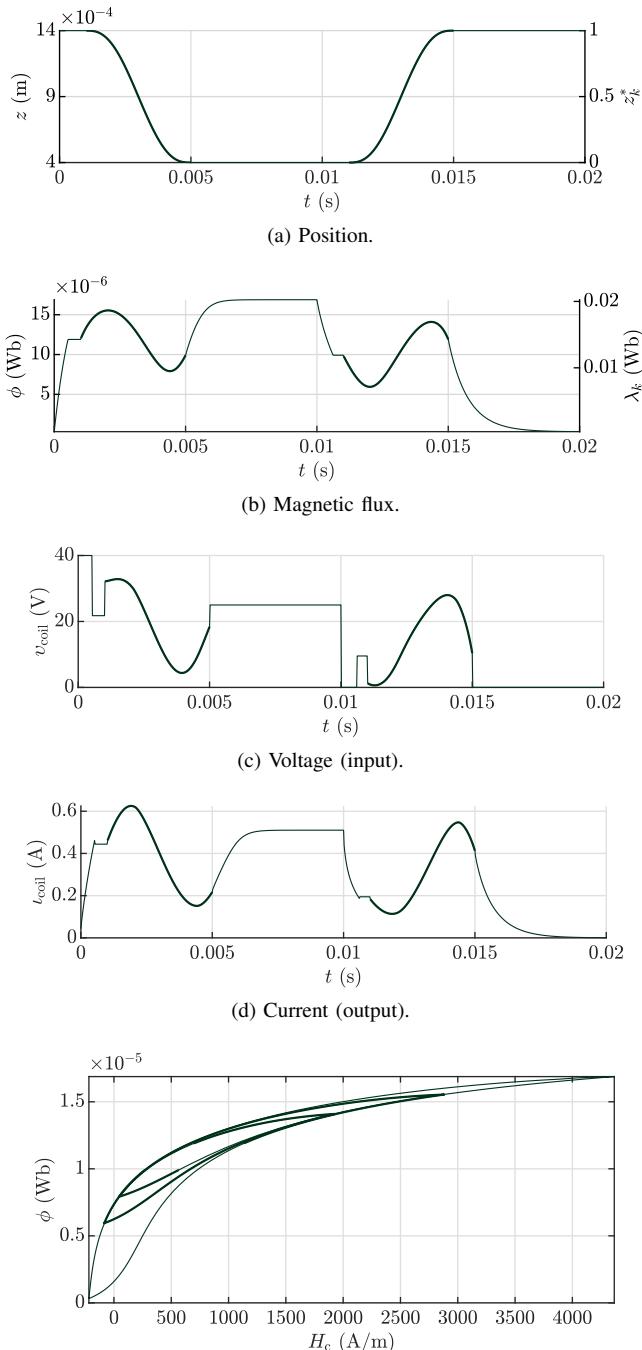


Fig. 5. Simulation results using the complete model. The wide lines correspond to the motion intervals.

B. Observability analysis

In order to check the suitability of the proposed estimator model and position trajectory, an observability analysis is proposed and applied. For this type of analysis, the estimator model is treated as deterministic, i.e. $\mathbf{w}_k = \mathbf{0}$ and $\nu_k = \mathbf{0}$ for all k .

Consider that the intention is to check if a state vector at $k = i$ is observable by measuring the outputs \mathbf{y}_i to \mathbf{y}_j , where $j \geq i$. Note that, when the armature is resting in one of the

limits, it is impossible to estimate the second state variable—i.e. the previous position z_{i-1}^* —from \mathbf{y}_i and any number of subsequent outputs. Nonetheless, in practice, it is not an issue because the second state variable is only necessary during motion. In any case, to analyze the observability of the entire state vector, the previous position z_{k-1}^* is considered an output that is available for the observation of the state vector \mathbf{x}_i . Taking that into consideration, the vector of available outputs is defined as follows:

$$\mathbf{Y}_i^j = \begin{cases} [\mathbf{y}_i^\top \mathbf{y}_{i+1}^\top \cdots \mathbf{y}_j^\top]^\top, & \text{if } q_{i-1} = 2 \\ [z_{i-1}^* \mathbf{y}_i^\top \mathbf{y}_{i+1}^\top \cdots \mathbf{y}_j^\top]^\top, & \text{if } q_{i-1} \neq 2 \end{cases}, \quad (48)$$

where each output \mathbf{y}_k can be expressed in terms of the state vector \mathbf{x}_k , which, in turn, can be expressed in terms of \mathbf{x}_i and a number of consecutive inputs,

$$\mathbf{y}_k = \mathbf{h}\left(\mathbf{f}\left(\mathbf{f}\left(\cdots \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i), \mathbf{u}_{i+1}, \dots, \mathbf{u}_{k-2}), \mathbf{u}_{k-1}\right), \mathbf{u}_k\right)\right). \quad (49)$$

Having defined the vector output \mathbf{Y}_i^n in terms of the state vector \mathbf{x}_i , an observability matrix can be computed as its Jacobian,

$$\mathcal{O}_i^j = \mathcal{O}_i^j(\mathbf{x}_i, \mathbf{u}_i, \mathbf{u}_{i+1}, \dots, \mathbf{u}_j) = \frac{\partial \mathbf{Y}_i^j}{\partial \mathbf{x}_i} \Big|_{\mathbf{x}_i}. \quad (50)$$

If the rank of the observability matrix \mathcal{O}_i^j is equal to the dimension of the state vector, it is observable. Note that, in theory, j may be arbitrarily large. In any case, it is interesting to determine the minimum number of measurements necessary to ensure observability of a certain state. Formally, we define it as follows:

$$\Delta k_{\text{obs}} = \min(j - i + 1) \quad \text{such that } \text{rank}(\mathcal{O}_i^j) = 5. \quad (51)$$

It represents the number of consecutive current measurements required to estimate the state vector \mathbf{x}_i . It can be computed for any state vector and set of consecutive inputs and outputs. In particular, for the reference state, inputs and outputs (Section IV-A), Δk_{obs} is calculated for every sampled instant and depicted in Fig. 6. When the actuator rests in the lower or upper limit, only two current measurements are required for estimating the state, as the position is already known. During motion, on the other hand, five current measurements are required, coinciding with the state vector dimension. There are transitions in which only some outputs include the position due to being in the lower or upper limit and, thus, Δk_{obs} is between two and five. Notice that there is also a sample for each switching operation in which the state is not observable, i.e. Δk_{obs} is infinite. It corresponds to the first sample after the impact or landing. The reason is that the second state variable, the previous position z_{k-1}^* , cannot be inferred from electrical current measurements after the contact instant. Nevertheless, pragmatically, this is not problematic because it can be estimated from previous data.

C. Model perturbation and specification

For the following analyses, the system stochasticity is considered. Specifically, numerous simulations are performed, in

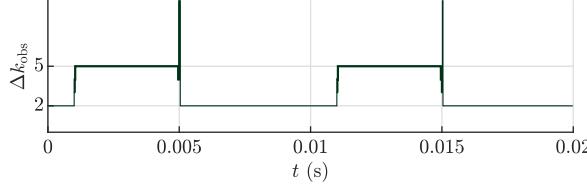


Fig. 6. Minimum number of consecutive samples to ensure observability.

which the model parameters for the estimators are perturbed. Consider \mathbf{p} to be the vector of perturbed parameters,

$$\mathbf{p} = [m^* \ k_{\text{sp}}^* \ z_{\text{sp}}^* \ c_f^* \ \mathcal{R}_{c,0}^* \ \lambda_{\text{sat}} \ \text{mean}(\mathcal{R}_g^*) \ \text{mean}(\mathcal{R}'_g^*)]^T. \quad (52)$$

Note that the average of both \mathcal{R}_g^* and its derivative is included. Formally, their mean values are calculated with respect to the position,

$$\text{mean}(\mathcal{R}_g^*) = \int_0^1 \mathcal{R}_g^*(z^*) dz^*, \quad \text{mean}(\mathcal{R}'_g^*) = \int_0^1 \mathcal{R}'_g(z^*) dz^*. \quad (53)$$

Then, the parameter values are perturbed in each run. Formally,

$$p_j = p_j^{\text{nom}} (1 + \epsilon_j), \quad (54)$$

where p_j denotes the j th component of \mathbf{p} , p_j^{nom} is the corresponding nominal parameter values and ϵ_j is the relative error. On the one hand, the nominal parameters are fitted to the simulation results from Section IV-A and presented in Table II. On the other hand, every relative error is randomly selected in each run according to a continuous uniform probability density distribution with an interval length $\Delta\epsilon$,

$$\epsilon_j \sim \text{unif}(1 - \Delta\epsilon/2, 1 + \Delta\epsilon/2). \quad (55)$$

The errors ϵ_7 and ϵ_8 serve to modify the mean values of \mathcal{R}_g^* and \mathcal{R}'_g^* (52). Then, with some simple manipulations, it is possible to derive their perturbed values for any position,

$$\mathcal{R}_g^*(z^*) = \mathcal{R}_g^{*\text{nom}}(z^*) + \text{mean}(\mathcal{R}_g^{*\text{nom}}) \epsilon_7 + (z^* - 1/2) \text{mean}(\mathcal{R}'_g^{*\text{nom}}) \epsilon_8, \quad (56)$$

$$\mathcal{R}'_g(z^*) = \mathcal{R}'_g^{*\text{nom}}(z^*) + \text{mean}(\mathcal{R}'_g^{*\text{nom}}) \epsilon_8. \quad (57)$$

Apart from the model parameters, the estimators also require setting the covariance of the process and observation noises. On the one hand, the observation covariance \mathbf{R}_k (36) depends only on the measurement errors of the current. To emulate this type of error, white noise is added to the current i_{coil} every sample of every run, with a standard deviation of $\sigma_i = 2 \times 10^{-3} \text{ A}$. Then, according to (29), the covariance \mathbf{R}_k is defined as

$$\mathbf{R}_k = \begin{cases} \sigma_i^2, & \text{if } q_k = 2 \\ \begin{bmatrix} \sigma_i^2 & 0 \\ 0 & 0 \end{bmatrix}, & \text{if } q_k \neq 2. \end{cases} \quad (58)$$

On the other hand, the process noise covariance \mathbf{Q} (33) is highly dependent on the errors of the parameters, so it is calculated for each $\Delta\epsilon$. First, based on the definition of the state transition (18), each process noise is determined as

$$\mathbf{w}_k^n = \mathbf{x}_{k+1} - \mathbf{f}^n(\mathbf{x}_k, \mathbf{u}_k), \quad (59)$$

where \mathbf{x}_{k+1} is the reference state vector (47) and $n \in \{1, 2, \dots, N_{\text{sim}}\}$ represents the distinction between simulation runs of the Monte Carlo method, in which the model parameters are perturbed. For this and the following sections, the number of simulation runs is set as $N_{\text{sim}} = 10\,000$. Then, as the process noise is assumed to be zero-mean, its covariance is defined as

$$\mathbf{Q} = \frac{1}{N_{\text{sim}}(k_f + 1)} \sum_{n=1}^{N_{\text{sim}}} \sum_{k=1}^{k_f} \mathbf{w}_k^n (\mathbf{w}_k^n)^T. \quad (60)$$

As explained in Section II-B, the first two components of the process noise vector are zero by definition. Moreover, in order to further simplify the matrix, the process noises are assumed uncorrelated, so every entry outside the diagonal is set to zero. Thus, the number of nonzero elements in the matrix is reduced to only three. Note that this is advantageous for the covariance estimation and fine-tuning in real applications.

Furthermore, the initial state is initialized in each simulation. It is assumed that both current and flux are zero at the start, so $e_{t,0}$ is also zero. As the coil is de-energized, the mover is initially in the upper limit, i.e. $z_0^* = 1$. The initial acceleration is zero, so $e_{F,0}$ must compensate exactly the estimated force of the spring (based on the perturbed parameters k_{sp}^* and z_{sp}^*). Ultimately, the mean state and covariance is initialized as

$$\mathbf{x}_{0|0} = [1 \ 1 \ 0 \ 0 \ -k_{\text{sp}}^*(z_{\text{sp}}^* - 1)]^T, \quad \mathbf{P}_{0|0} = \mathbf{0}. \quad (61)$$

Note that the initial magnetic flux is not exactly zero in the reference (see Fig. 5e). Technically, its initial variance should be positive to account for that discrepancy. Nonetheless, for the sake of simplicity, we have decided to set it to zero.

D. Consistency analysis

The proposed estimator is a suboptimal extension of a linear fixed-interval smoother (see Section III). In addition, several model simplifications have been proposed (see Sections II-B and IV-C) to further facilitate its initialization and implementation. Due to these reasons, the consistency of the estimated covariance matrices cannot be theoretically guaranteed.

A consistency analysis is provided based on the presented Monte Carlo method, in which the interval length of the parameter errors is set to $\Delta\epsilon = 1\%$. The position from Fig. 5a is estimated by using the electrical signals from Figs. 5c and 5d. The chosen sampling period is $T_s = 20 \mu\text{s}$. As only the motion operations are of interest, the smoothing algorithm is applied only on those intervals. In that regard, for each operation, the last motion sample (in which $q_k = 2$) is treated as $k_f - 1$, marking the start of the backwards smoothing. Then, the algorithm stops when $q_k \neq 2$.

The normalized estimation error squared (NEES) of the entire state vector is calculated for every simulation run and sample k as follows:

$$\text{NEES} = (\hat{\mathbf{x}}_{k|k_f} - \mathbf{x}_k)^T \mathbf{P}_{k|k_f}^{-1} (\hat{\mathbf{x}}_{k|k_f} - \mathbf{x}_k), \quad (62)$$

where $\hat{\mathbf{x}}_{k|k_f}$ and $\mathbf{P}_{k|k_f}$ are the estimated expectation and covariance of the state vector, while \mathbf{x}_k is the corresponding reference vector (47).

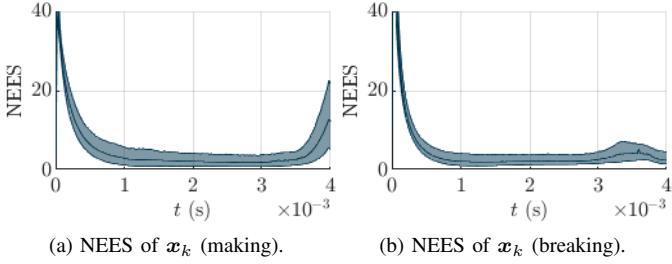


Fig. 7. Mean and 5–95th percentile intervals of the normalized estimation errors squared.

The mean and 5–95th percentile intervals of NEES are represented in Fig. 7 as functions of time for the making and breaking operations. Note that, under the initial hypothesis that the estimation errors are realizations of a zero-mean random vector, NEES should be chi-square distributed with 5 degrees of freedom. Fittingly, 89.56 % of all values are inside the 90 % confidence interval, which is approximately [1.145, 11.070]. Moreover, the overall average of NEES is 4.9477, very close to its expectation, which is exactly 5. In that regard, the estimation errors are very consistent with the true covariance matrix. Nonetheless, there is a noticeable dependency on time and type of operation. The NEES are considerably larger in average at the start of each operation, i.e. the estimations are initially overconfident.

Another useful metric for analyzing the consistency is the normalized estimation error (NEE) of each state variable $x_{k,i}$. Formally, it is calculated for each simulation run and sample k as

$$\text{NEE}_i = (\hat{x}_{k|kf,i} - x_{k,i}) / \sqrt{P_{k|kf,ii}}. \quad (63)$$

The mean and 5–95th percentile intervals of every NEE are represented in Fig. 8 for both operation types. In theory, NEE values should be consistent with a standard normal distribution. However, there is a noteworthy time-dependent bias for each state variable, as the mean values of NEE indicate. Furthermore, focusing on the position estimations, most NEE_1 values are inside the 90 % confidence interval for a standard normal random variable, which is approximately $[-1.655, 1.655]$. This means that the ERTSS estimation of the position variance is overly conservative or, in other words, the position estimation errors are smaller than expected.

In any case, note that the estimation consistency is not as critical for the intended application as in scenarios in which the state of the system is estimated indefinitely. The estimator is a fixed-interval smoother intended for determining the position trajectories during the motion intervals. Moreover, given the repetitive nature of the switching operations, the estimation can be reset after each operation (or making and breaking cycle), so there is effectively no divergence risk due to estimation inconsistencies.

E. Simulated comparison

The most important performance indicators are the estimation errors. Based on those, the proposal is analyzed

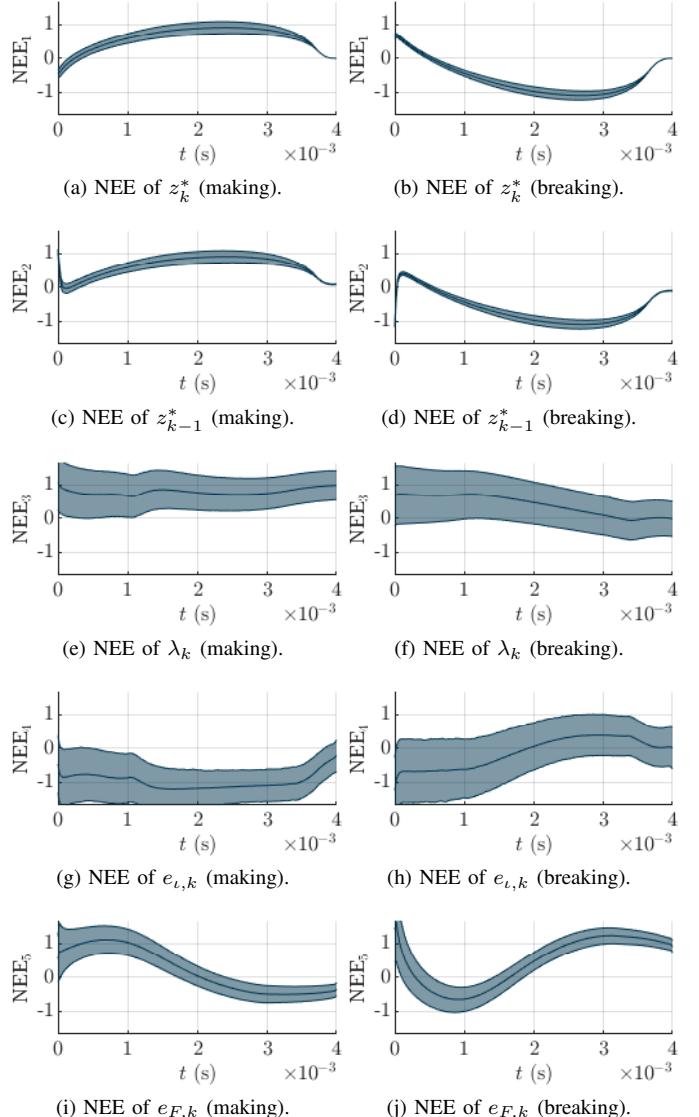


Fig. 8. Mean and 5–95th percentile intervals of the normalized estimation errors for each state variable.

and compared with three other alternatives under the same conditions:

- **EKF:** The second estimation algorithm is an extended Kalman filter, as described in Section III-A.
- **ERTSS\mathbf{\lambda}_t:** The third estimator uses the same procedure as the proposed ERTSS, but with a slightly simplified model: the auxiliary state variable e_t is set to zero, i.e.

$$f_{e_t}(\cdot) = 0, \quad w_{e_t,k} = 0. \quad (64)$$

- **ERTSS\mathbf{e}_F:** Similarly, a fourth ERTSS is designed by changing the model. In this case, the other auxiliary variable e_F is set to zero:

$$f_{e_F}(\cdot) = 0, \quad w_{e_F,k} = 0. \quad (65)$$

Note that e_F does not only represent force deviations due to parametric or discretization errors, but also accounts

for the normal forces during contact. As such, the transition function of the position (22) must also be tweaked to circumvent the removal of e_F ,

$$f_{z^*}(\cdot) = \begin{cases} 0, & \text{if } q_k = 1 \\ 2z_k^* - z_{k-1}^* + T_s^2 f_v^*(\cdot), & \text{if } q_k = 2. \\ 1, & \text{if } q_k = 3 \end{cases} \quad (66)$$

The comparison between EKF and ERTSS is useful for quantifying the accuracy increase of the smoother. On the other hand, the results from ERTSS\(e_t and ERTSS\(e_F serve to determine the overall improvement of the estimations due to these proposed additions.

For each one of the estimators, and for different error interval lengths $\Delta\epsilon$, 10 000 Monte Carlo simulations are performed. To summarize and compare the results, normalized root-mean-square errors (NRMSE) of the position are calculated for each operation,

$$\text{NRMSE}_z = \sqrt{\frac{\sum_n \sum_k (\hat{z}_k^{*n} - z_k^*)^2}{N_{\text{sim}} \sum_k (z_k^*)^2}}, \quad (67)$$

where \hat{z}_k^{*n} is the estimated normalized position for each sample k and Monte Carlo run n , and z_k^* is the normalized reference position (47a). Note that the position errors are zero outside the motion intervals, so only the samples k in which $q_k = 2$ are used to compute each NRMSE_z .

The errors are represented in Figs. 9a and 9b for the making and breaking operations, respectively. The advantage of the proposed ERTSS over a more traditional EKF is quite evident, with smaller errors for every $\Delta\epsilon$. There is also a notable improvement over ERTSS\(e_t and ERTSS\(e_F , which justifies the addition of the auxiliary variables e_t and e_F . On the one hand, e_F mainly accounts for modeling errors in the force characterization, so its usefulness is clearer for larger parameter errors. On the other hand, the main motivation for adding e_t is to account for current errors due to the lack of a hysteresis model. Thus, as the graphics confirm, it is useful even for negligible parameter errors.

As the goal is to reduce impact velocities, the estimation of the velocity is also very important for soft-landing control. Then, in an equivalent fashion, the velocity estimation errors NRMSE_v are calculated and presented in Figs 9c and 9d. The velocity errors are consistently larger, but for $\Delta\epsilon < 10\%$ the proposed ERTSS is still very accurate.

To better understand why the fixed-interval smoother (ERTSS) is so much better than its filter counterpart (EKF), Fig. 10 depicts the full position evolution of the worst-case scenario for $\Delta\epsilon = 10\%$ (of all 10 000 ERTSS runs with 2 operations each, the one with the maximum NRMSE_z is selected). As explained in Section III, the forward filter recursion of the ERTSS is precisely the EKF. The real advantage of the ERTSS is that, once the landing occurs, the position estimate can be corrected. Then, the backwards smoothing recursion serves to correct past data based on the position correction at the contact instant.

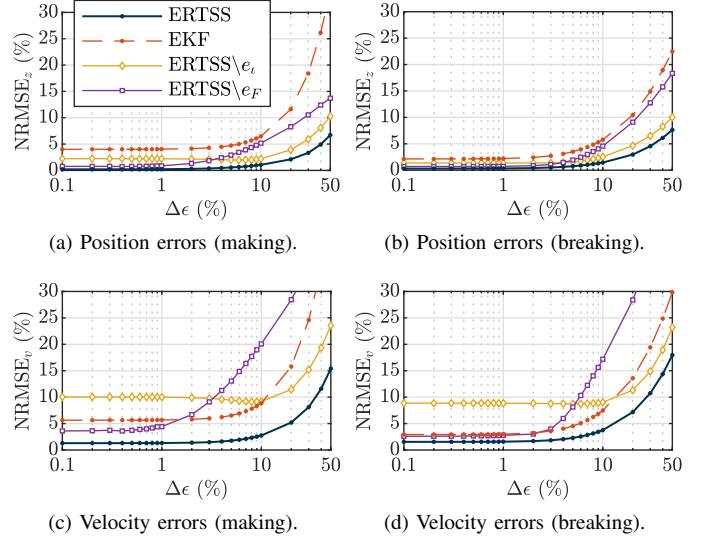


Fig. 9. Estimation errors in the making and breaking operations, for different interval lengths $\Delta\epsilon$. Notice the logarithmic scale of the horizontal axis.

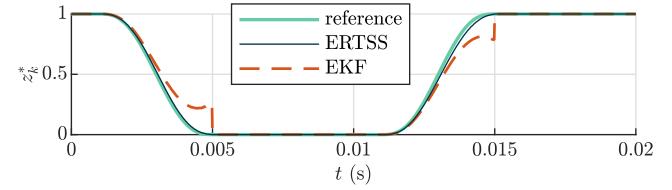


Fig. 10. Worst-case result regarding the position estimation, for $\Delta\epsilon = 10\%$.

F. Experimental validation

To further validate the proposal, the estimators are tested and compared using an experimental setup, which is displayed in Fig. 11. The device is a plunger-type reluctance actuator, of the same type as the one shown in Fig. 3. The movement of the plunger is measured with a high-speed and high-accuracy laser displacement sensor (Keyence LK-G5001P controller with LK-H082 sensor head). On the other hand, the coil voltage and current are measured with a high-precision PC oscilloscope with an arbitrary waveform generator (PicoScope 4824).

For the first experimental validation, a single making and breaking cycle is tested. The voltage signal is displayed in Fig. 12a. A constant voltage is applied for each operation (30 V for the making and 0 V for the breaking). Note that there are small intervals in which the alternative voltage value is applied (0 V for the making and 30 V for the breaking), intended for slowing down the actuator and decreasing the impact velocities. The measured current signal is also displayed in Fig. 12a, whereas the measured position is presented in Fig. 12b. Instead of measuring the contacts, the discrete state of the hybrid automaton (Fig. 2) is directly derived from the position measurements as follows:

$$q = \begin{cases} 1, & \text{if } z \leq z_{\min} \\ 2, & \text{if } z_{\min} < z < z_{\max}, \\ 3, & \text{if } z \geq z_{\max} \end{cases} \quad (68)$$

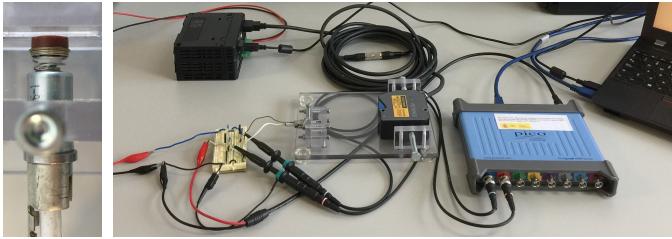


Fig. 11. Plunger-type actuator (left) and experimental setup with the actuator, a laser displacement sensor and a PC oscilloscope (right).

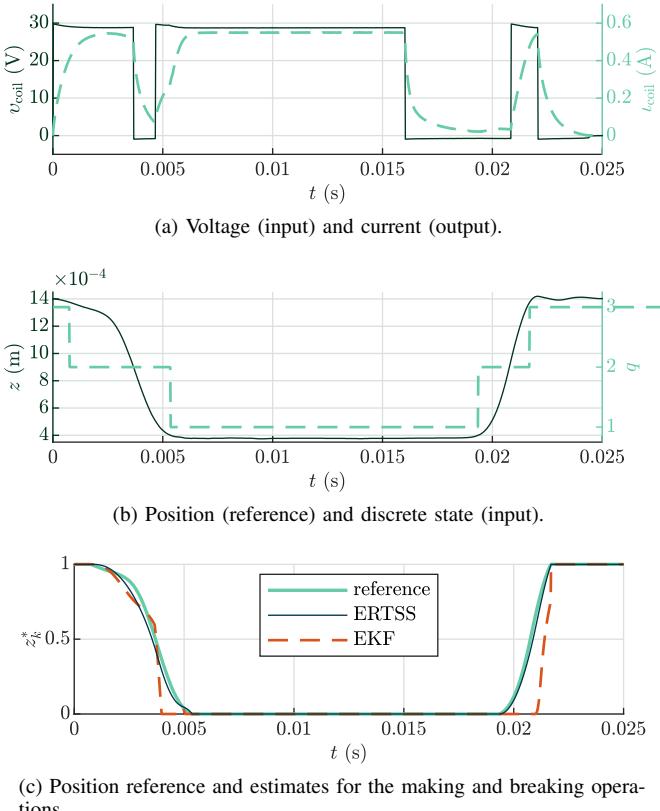


Fig. 12. Experimental test of the proposed estimator.

where the minimum and maximum position values are calculated with a 2.5 % tolerance in order to filter out measurement noise and the deformations of the rubber at the end of the plunger when starting and ending the motion,

$$z_{\min} = \min(z) + 0.025(\max(z) - \min(z)), \quad (69a)$$

$$z_{\max} = \max(z) - 0.025(\max(z) - \min(z)). \quad (69b)$$

Then, the position is estimated using the proposed ERTSS, with the voltage and discrete state as inputs, and current as output. The parameters and variance values of the estimator model are taken directly from the simulated cases in which $\Delta\epsilon = 10\%$. Note that one of the main features of the proposal is its robustness against significant modeling errors. Thus, to emphasize this, the model parameters are not fine-tuned.

The estimated position for both making and breaking operations are displayed in Fig 12c. The EKF estimates correspond to the forward pass of the proposal, whereas the ERTSS estimates correspond to the complete proposal, with the smoothing

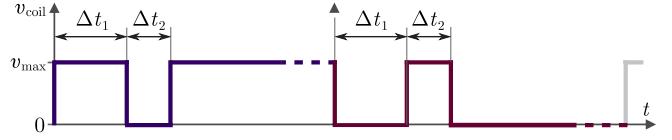


Fig. 13. Profile of the voltage applied to the actuator.

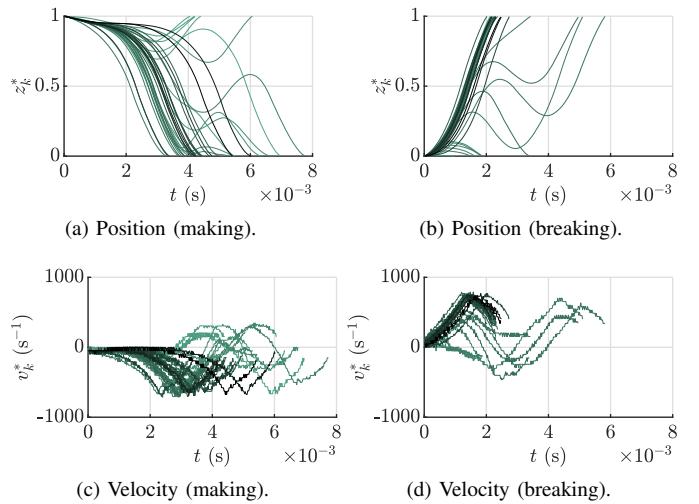


Fig. 14. Measured position and velocity trajectories during switching operations.

backward pass. The effect of the modeling errors are readily apparent from the EKF results, as the position estimation errors are very large. ERTSS, on the other hand, manages to estimate the position much better in both operations.

For certain control scenarios, the voltage applied to the device would be modified in each iteration. For every case, the position estimates should be as accurate as possible so that they can be exploited by a control strategy. On that basis, for a more thorough validation, the same four estimators compared in Section IV-E are tested for multiple making and breaking cycles. The voltages signals are constructed following the profiles presented in Fig. 13, where the maximum voltage is set as $v_{\max} = 30$ V. In addition, the time intervals are modified such that $\Delta t_1 \in \{3, 3.5, \dots, 6\}$ and $\Delta t_2 \in \{1, 1.25, \dots, 2\}$. From every one of the possible 35 combinations, the position is measured and the corresponding velocity is derived. They are presented in Fig. 14. Then, the estimators are used to calculate the position trajectories based on the corresponding measured inputs and outputs. The absolute errors of the position and velocity estimations are condensed in the box plots from Fig. 15. Focusing on the proposal, most estimation errors are fairly small (approximately 75 % of the position and velocity errors are smaller than 0.1 and $130 s^{-1}$, respectively). Moreover, as in the simulated cases, the benefits of the smoothing backward pass and the incorporation of e_L and e_F into the model are clear by comparing the ERTSS results with EKF, ERTSS\(e_L , and ERTSS\(e_F , respectively.

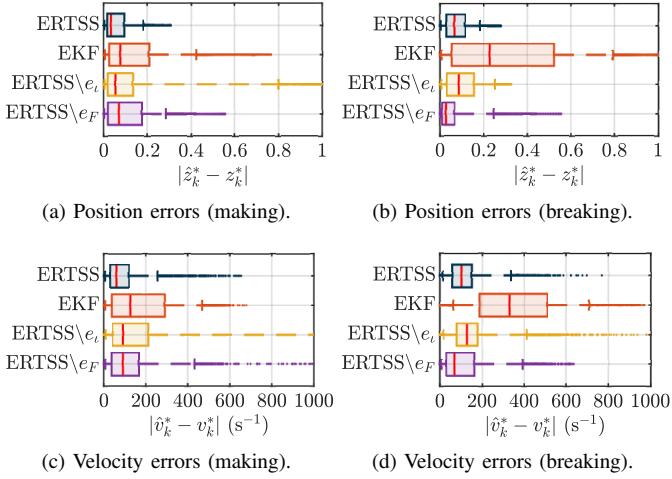


Fig. 15. Box plots of estimation absolute errors of position and velocity for making and breaking operations. The boxes correspond to the 25–75th intervals, the central red marks indicate the median values, the left and right whiskers extend to the 5th and 95th percentiles, and the points represent outliers.

V. CONCLUSIONS

This paper proposes a position estimator for soft-landing control of short-stroke single-coil reluctance actuators. It is a Rauch–Tung–Striebel fixed-interval smoother, which uses the electrical signals and contact information as observable variables. To facilitate its implementation, it also uses a deliberately simple model. Despite that, the simulated and experimental results show that the estimation of both position and velocity are very accurate, even with large parameter errors. This is especially evident when compared to other alternatives, proving the advantage of the novel ideas. On the one hand, the incorporation of error terms as state variables serves to estimate and correct modeling errors. On the other hand, the backward recursion exploits the information gained after the contact instant and corrects past estimates correspondingly.

We have proposed to use the discrete state of the hybrid system—i.e. the dynamic mode of the system—as a second input of the estimator model. This requires a sensor to know if the mover is in contact with one of its limits. This is a compromise between measuring only electrical signals and measuring directly the position. On the one hand, if only the current and voltage are measured, the estimator must be based on a more complex and accurate model, such as the ones presented in [22] or [23]. This makes the model identification and estimator implementation much more challenging and expensive. On the other hand, a full position sensor would not be affordable, especially for low-cost solutions with simple reluctance actuators.

ACKNOWLEDGMENT

E. Moya-Lasherás would like to express his great appreciation to Prof. E.A. Lomonova, Prof. Dave Krop and the staff of the Electromechanics and Power Electronics group for their hospitality and assistance during his research stay at the Eindhoven University of Technology. E. Moya-Lasherás and

C. Sagües would also like to express their gratitude to Sergio Llorente and BSH Home Appliances Spain for their support.

REFERENCES

- [1] S. K. Chung, C. R. Koch, and A. F. Lynch, “Flatness-Based Feedback Control of an Automotive Solenoid Valve,” *IEEE Transactions on Control Systems Technology*, vol. 15, no. 2, pp. 394–401, mar 2007.
- [2] R. R. Chladny and C. R. Koch, “Flatness-Based Tracking of an Electromechanical Variable Valve Timing Actuator With Disturbance Observer Feedforward Compensation,” *IEEE Transactions on Control Systems Technology*, vol. 16, no. 4, pp. 652–663, jul 2008.
- [3] R. Gill, A. Wahrburg, O. Craciun, K. Listmann, and C. Reuber, “Model Based Landing Control for a Bistable Electromagnetic Actuator with Discontinuous Dynamics,” in *2015 IEEE Conference on Control Applications (CCA)*, Sydney, Australia, sep 2015, pp. 1479–1485.
- [4] X. Zhao, L. Li, J. Song, C. Li, and X. Gao, “Linear Control of Switching Valve in Vehicle Hydraulic Control Unit Based on Sensorless Solenoid Position Estimation,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 7, pp. 4073–4085, jun 2016.
- [5] E. Moya-Lasherás, E. Ramírez-Laboreo, and C. Sagües, “Model-Free Sliding-Mode Controller for Soft Landing of Reluctance Actuators,” in *21st IFAC World Congress 2020*, Berlin, Germany, jul 2020.
- [6] T. Glück, W. Kemmetmüller, and A. Kugi, “Trajectory optimization for soft landing of fast-switching electromagnetic valves,” *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 11532–11537, sep 2011.
- [7] E. Moya-Lasherás, E. Ramírez-Laboreo, and C. Sagües, “Probability-Based Optimal Control Design for Soft Landing of Short-Stroke Actuators,” *IEEE Transactions on Control Systems Technology*, vol. 28, no. 5, pp. 1956–1963, sep 2020.
- [8] T. Braun, J. Reuter, and J. Rudolph, “Flatness-Based Feed-Forward Control Design for Solenoid Actuators Considering Eddy Currents,” in *8th IFAC Symposium on Mechatronic Systems*, vol. 52, no. 15, Vienna, Austria, sep 2019, pp. 567–572.
- [9] M. Benosman and G. M. Atıç, “Multi-Parametric Extremum Seeking-based Learning Control for Electromagnetic Actuators,” in *2013 American Control Conference*, Washington, DC, USA, 2013, pp. 1914–1919.
- [10] E. Ramírez-Laboreo, C. Sagües, and S. Llorente, “A New Run-to-Run Approach for Reducing Contact Bounce in Electromagnetic Switches,” *IEEE Transactions on Industrial Electronics*, vol. 64, no. 1, pp. 535–543, jan 2017.
- [11] W. Hoffmann, K. Peterson, and A. G. Stefanopoulou, “Iterative learning control for soft landing of electromechanical valve actuator in camless engines,” *IEEE Transactions on Control Systems Technology*, vol. 11, no. 2, pp. 174–184, mar 2003.
- [12] J. Tsai, C. R. Koch, and M. Saif, “Cycle Adaptive Feedforward Approach Controllers for an Electromagnetic Valve Actuator,” *IEEE Transactions on Control Systems Technology*, vol. 20, no. 3, pp. 738–746, may 2012.
- [13] E. Moya-Lasherás and C. Sagües, “Run-to-Run Control With Bayesian Optimization for Soft Landing of Short-Stroke Reluctance Actuators,” *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 6, pp. 2645–2656, dec 2020.
- [14] T. Glück, W. Kemmetmüller, C. Tump, and A. Kugi, “A novel robust position estimator for self-sensing magnetic levitation systems based on least squares identification,” *Control Engineering Practice*, vol. 19, no. 2, pp. 146–157, feb 2011.
- [15] M. F. Rahman, N. N. C. Cheung, K. W. Lim, and Khiang Wee Lim, “Position estimation in solenoid actuators,” *IEEE Transactions on Industry Applications*, vol. 32, no. 3, pp. 552–559, may 1996.
- [16] R. R. Chladny and C. R. Koch, “A Magnetic Flux-Based Position Sensor for Control of an Electromechanical VVT Actuator,” in *2006 American Control Conference*, Minneapolis, Minnesota, USA, jun 2006, pp. 3978–3983.
- [17] T. Braun, J. Reuter, and J. Rudolph, “Observer Design for Self-Sensing of Solenoid Actuators With Application to Soft Landing,” *IEEE Transactions on Control Systems Technology*, vol. 27, no. 4, pp. 1720–1727, jul 2019.
- [18] P. Mercorelli, “A Hysteresis Hybrid Extended Kalman Filter as an Observer for Sensorless Valve Control in Camless Internal Combustion Engines,” *IEEE Transactions on Industry Applications*, vol. 48, no. 6, pp. 1940–1949, nov 2012.
- [19] E. Moya-Lasherás, C. Sagües, E. Ramírez-Laboreo, and S. Llorente, “Nonlinear Bounded State Estimation for Sensorless Control of an Electromagnetic Device,” in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, Melbourne, Australia, dec 2017, pp. 5050–5055.

- [20] F. Straußberger and J. Reuter, "Position Estimation in Electro-Magnetic Actuators Taking into Account Hysteresis Effects," in *7th IFAC Symposium on Mechatronic Systems*, vol. 49, no. 21, Loughborough University, Leicestershire, UK, sep 2016, pp. 206–212.
- [21] E. Ramirez-Laboreo, "Modeling and Control of Reluctance Actuators," Ph.D. dissertation, Universidad de Zaragoza, 2019.
- [22] E. Ramirez-Laboreo, M. G. L. Roes, and C. Sagües, "Hybrid Dynamical Model for Reluctance Actuators Including Saturation, Hysteresis and Eddy Currents," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 3, pp. 1396–1406, jun 2019.
- [23] E. Moya-Lasheras, C. Sagües, and S. Llorente, "An efficient dynamical model of reluctance actuators with flux fringing and magnetic hysteresis," *Mechatronics*, vol. 74, 2021.
- [24] E. Ramirez-Laboreo, C. Sagües, and S. Llorente, "A New Model of Electromechanical Relays for Predicting the Motion and Electromagnetic Dynamics," *IEEE Transactions on Industry Applications*, vol. 52, no. 3, pp. 2545–2553, may 2016.
- [25] L. Verlet, "Computer Experiments on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules," *Physical Review*, vol. 159, no. 1, pp. 98–103, jul 1967.
- [26] P. S. Maybeck, "Design and performance analysis of Kalman filters," in *Stochastic Models, Estimation, and Control*. Academic Press, 1979, vol. 1, ch. 6.3, pp. 291–297.
- [27] S. Särkkä, *Bayesian filtering and smoothing*. Cambridge, UK: Cambridge University Press, 2013, vol. 3.
- [28] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum likelihood estimates of linear dynamic systems," *AIAA Journal*, vol. 3, no. 8, pp. 1445–1450, aug 1965.
- [29] D. Simon, "Kalman filtering with state constraints: a survey of linear and nonlinear algorithms," *IET Control Theory and Applications*, vol. 4, no. 8, pp. 1303–1318, aug 2010.
- [30] H. Cox, "On the Estimation of State Variables and Parameters for Noisy Dynamic Systems," *IEEE Transactions on Automatic Control*, vol. 9, no. 1, pp. 5–12, jun 1964.
- [31] C. Robert Koch, A. F. Lynch, and R. R. Chladny, "Modeling and Control of Solenoid Valves for Internal Combustion Engines," *IFAC Proceedings Volumes*, vol. 35, no. 2, pp. 197–202, dec 2002.
- [32] M. Montanari, F. Ronchi, and C. Rossi, "Trajectory Generation for Camless Internal Combustion Engine Valve Control," in *IEEE International Symposium on Industrial Electronics*. Rio de Janeiro, Brazil: IEEE, jun 2003, pp. 454–459.
- [33] A. Fabbrini, A. Garulli, and P. Mercorelli, "A Trajectory Generation Algorithm for Optimal Consumption in Electromagnetic Actuators," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 4, pp. 1025–1032, jul 2012.