2018 22nd International Conference on System Theory, Control and Computing (ICSTCC)

Comparison of State Estimators for a Permanent Magnet Synchronous Generator

Lavinius Ioan Gliga
ACSE, ACS
"Politehnica" University
Bucharest, Romania
lavinius.gliga@acse.pub.ro

Houcine Chafouk
IRSEEM, ESIGELEC, UNIRouen
Normandy University
Rouen, France
houcine.chafouk@esigelec.fr

Dumitru Popescu
ACSE, ACS
"Politehnica" University
Bucharest, Romania
dumitru.popescu@acse.pub.ro

Ciprian Lupu
ACSE, ACS
"Politehnica" University
Bucharest, Romania
ciprian.lupu@acse.pub.ro

***Abstract*—A Permanent Magnet Synchronous Generator is chosen as a case study for this comparison. A continous model with a discrete integrator is shown to be the best solution for the discretization of the nonlinear model. Then, the Kalman Filter (KF), the Extended Kalman Filter (EKF) and the Unscented Kalman Filter are compared. The results are presented and it is shown that the EKF is the best suited for this application. Moreover, a discussion regarding the behaviour of the filters follows, where all are shown to act like proportional controllers.**

***Index Terms*—Nonlinear Model, Discretization, Kalman Filter, Extended Kalman Filter, Unscented Kalman Filter**

I. INTRODUCTION

State estimators are one of the two possible approaches to assure sensor redundancy, the other being state observers. This redundancy is critical for the monitoring of sensors and different equipments, and for constructing residuals which may be later used in fault diagnosis. The purpose of this paper is to lay a foundation for the latter.

State observers and estimators are ideal candidates for software sensors. The main difference between them is that the latter considers the statistic properties of the process. State estimators use the covariance matrices of the states, the process and the measurement noises. They also do not require a priori knowledge of the process uncertainties or the impact of faults. They can also be more insensitive to noises.

The best-known state estimator is the Kalman Filter (KF). It is an optimal estimator for linear systems and is widely used. The Extended Kalman Filter (EKF) is the first nonlinear extension of the classical KF. It is widely used in localization and navigation, being the de facto standard. The Unscented Kalman Filter (UKF) is a further nonlinear extension of the Kalman Filter. It is used in military and aeronautic applications, as it can have superior performance to the EKF, depending on the application [1].

The objective of this paper is to study the differences between these three state estimators. The selected case study is the Permanent Magnet Synchronous Generator (PMSG). These are used in direct drive wind turbines [2], and their motor counterparts are widely used in hybrid electric vehicles [3]. Because the mathematical model is the same, and their construction is similar, the results obtained in this paper can be extended to motors.

The contribution of this paper is the comparison between the three different state observers in the case of the PMSG and the discussion about the behaviour of the KF and EKF, when estimating the states of a process.

This paper is organized as follows: the model of the PMSG will be presented in Section II. The different discretization methods will be shown and compared in Section III. The algorithms of the KF, EKF and UKF will be presented in Section IV. The obtained results will be shown in Section V, and they will be discussed. The conclusions and the perspectives will close this paper.

II. THE MODEL OF THE PERMANENT MAGNET SYNCHRONOUS GENERATOR

The dynamic model of a surface mounted PMSG is [4]

$$\begin{aligned}\dot{I}_d(t) &= \frac{V_d(t) - R_d I_d(t) + n_p \omega_m(t) L_q I_q(t)}{L_d} \\ \dot{I}_q(t) &= \frac{V_q(t) - R_q I_q(t) - n_p \omega_m(t) L_d I_d(t)}{L_q} - \\ &\qquad \frac{n_p \omega_m(t) \phi}{L_q}\end{aligned} \tag{1}$$

where I_d, I_q are the currents (in A), V_d, V_q are the voltages (in V), obtained using through the Park Transform. R_d, R_q are the stator resistances (in Ω) and L_d, L_q are the inductances (in H). ω_m is the angular speed of the rotor shaft (in rpm), ϕ is flux linkage between the rotor and the stator (in Wb) and n_P is the number of pole pairs. For simplicity, it can be considered that $L_d \approx L_q = L_s$ and $R_d \approx R_q = R_s$ where L_s and R_s denote the stator inductance (in H) and resistance (in Ω) [5]. The states are the currents and the inputs are the voltages and the angular velocity of the shaft.

The model is nonlinear, as it contains the product between a state and an input.

The model of the generator is continuous. The goal is to study the difference between the three filters in a discrete simulation, to mimic the behaviour of the algorithms running on a microcontroller. As a microcontroller is a discrete system, the model must be discretized.

III. DISCRETIZATION

It is difficult to discretize a nonlinear function. So, the most appealing option is to use the Taylor Series Expansion (TSE) to obtain a linear model [5]

$$\begin{bmatrix} i_{d_{k+1}} \\ i_{q_{k+1}} \end{bmatrix} = F_k * \begin{bmatrix} i_{d_k} \\ i_{q_k} \end{bmatrix} + G_k * \begin{bmatrix} V_{d_k} \\ V_{q_k} \end{bmatrix} + H_k \qquad (2)$$

where

$$F_k = \begin{bmatrix} 1 - \frac{R_s T_s}{L_s} & T_s n_P \omega_{m_k} \\ -T_s n_P \omega_{m_k} & 1 - \frac{R_s T_s}{L_s} \end{bmatrix}$$

$$G_k = \begin{bmatrix} \frac{T_s}{L_s} & 0 \\ 0 & \frac{T_s}{L_s} \end{bmatrix} and\ H_k = \begin{bmatrix} 0 \\ -\frac{T_s n_P \omega_k}{L_s} \phi \end{bmatrix}$$

and T_s is the sampling period.

However, any linearization might introduce errors in the model. In [6] it is suggested to use the continuous model, but with the following discrete integrator

$$x_k = x_{k-1} + \dot{x} * Ts;$$

where x is the state vector of the process..

However, this integrator differs from the one used in Simulink, in the Power Systems Toolbox. There, when the PMSG is simulated in discrete mode, the continuous model is used but with a Forward Euler Integrator (FEI) [7]

$$y_k = x_k$$

$$x_{k+1} = x_k + T_s * u_k$$

where y is the output of the integrator, x is its internal state and u is its input, i.e. the derivative of the system states.

The simulation results are presented in Table I. The errors obtained with the continuous integrator and the FEI are similar, because the sampling period was chosen to be very small, 10^{-6}. This is to prevent numerical instability in the simulation. In Fig. III and Fig. III, the methods appear to return the same results, but this is out of coincidence. The simulation was checked, but the same results were obtained. Although Fig. III and III are identical, they show that the discrete integrator behaves like the continuous one, for the chosen sampling period (10^{-6} s). All the simulations were done in Matlab, Simulink, using the Simscape/Power Systems toolbox.

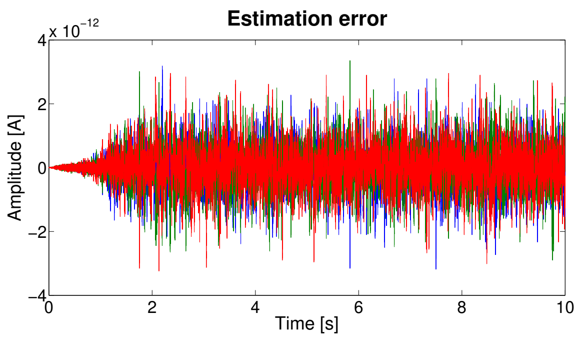

Fig. 1. The continuous model with a continuous integrator

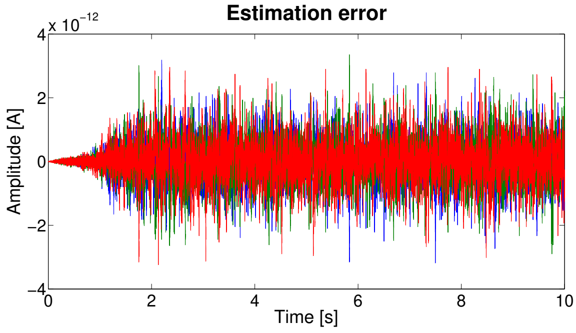

Fig. 2. The continuous model with the discrete FEI

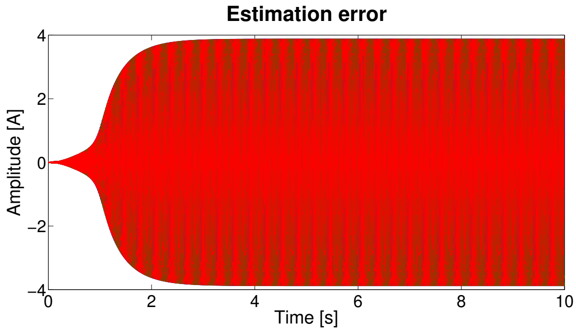

Fig. 3. The continuous model with the discrete integrator from [6]

A small error appears for the continuous model with a continuous integrator because of how Simulink compiles the schematic. Any collection of Power Systems blocks is approximated by a state space model [7].

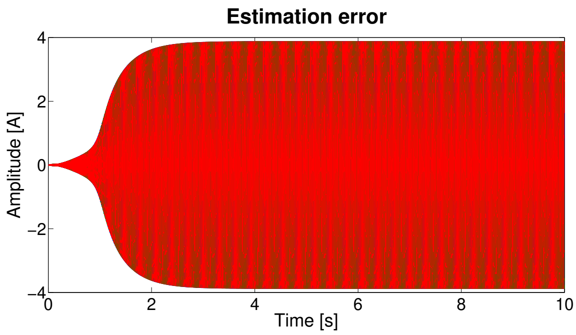

Fig. 4. The linearized model

TABLE I
COMPARISON OF THE DISCRETIZATION METHODS

Model	Integrator type	Order of error
Continuous	Continuous	$\approx 10^{-13}$
Continuous	Discrete - FEI	$\approx 10^{-13}$
Continous	Discrete - from [6]	≈ 4
Linearised	N/A	≈ 4

IV. The State Estimators

A. The Kalman Filter

Although the linear model from (2) introduces significant modelling errors, it would be interesting to see if a KF, which integrates this model, would achieve better results. The Kalman Filter uses a linear model of the form

$$\begin{aligned} \hat{x}_{k+1} &= A_k * \hat{x}_k + B_k * u_k \\ \hat{y}_k &= C * \hat{x}_k \end{aligned} \tag{3}$$

where $x \in \mathbb{R}^{n_x}$ are the states of the process. $u \in \mathbb{R}^{n_u}$ are the inputs and $y \in \mathbb{R}^{n_y}$ are the outputs of the process. $A \in \mathbb{R}^{n_x * n_x}$ is the state matrix, $B \in \mathbb{R}^{n_x * n_u}$ is the input matrix and $C \in \mathbb{R}^{n_y * n_x}$ is the output matrix. The number of states is n_x, the number of inputs is n_u and the number of measurements is n_y. The sampling time is k. The "^" denotes an estimation.

The model from (2) can be put into this form by combining the matrices G and H and using a vector with three elements for the inputs

$$\begin{bmatrix} i_{d_{k+1}} \\ i_{q_{k+1}} \end{bmatrix} = A_k * \begin{bmatrix} i_{d_k} \\ i_{q_k} \end{bmatrix} + B_k * \begin{bmatrix} V_{d_k} \\ V_{q_k} \\ \phi \end{bmatrix} \tag{4}$$

where

$$A_k = F_k = \begin{bmatrix} 1 - \frac{R_s T_s}{L_s} & T_s n_P \omega_k \\ -T_s n_P \omega_k & 1 - \frac{R_s T_s}{L_s} \end{bmatrix}$$

$$B_k = \begin{bmatrix} \frac{T_s}{L_s} & 0 & 0 \\ 0 & \frac{T_s}{L_s} & -\frac{T_s n_P \omega_k}{L_s} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Then, the classical KF algorithm can be used [8]

- Prediction phase

$$\hat{P}_k = A_k \hat{P}_k^* A_k^T + Q_k; \tag{5}$$

- Update phase

$$K_k = \hat{P}_k C^T (C \hat{P}_k C^T + R_k)^{-1} \tag{6}$$

$$\hat{x}_{k+1}^* = \hat{x}_k + K_k (y_k - C \hat{x}_k) \tag{7}$$

$$\hat{P}_k^* = (I - K_k C) \hat{P}_k \tag{8}$$

where $P \in \mathbb{R}^{n_x * n_x}$ is the state covariance matrix, $Q \in \mathbb{R}^{n_x * n_x}$ and $R \in \mathbb{R}^{n_y * n_y}$ are the covariance matrices of the process and the measurement noises. $K \in R^{n_x * n_y}$ is the Kalman gain and $y \in R^{n_y}$ are the measurements acquired from the process. The "*" denotes the corrected estimation.

For all the presented state estimators, the covariance matrices of the process and of the measurement noises are computed as in [9].

B. The Extended Kalman Filter

The EKF also introduces a linearization, through the TSE of the state function. This linearization is used to compute the estimation of the state covariance matrix. However, these linearizations are computed only around the current estimated state, so the introduced error should be smaller.

The EKF uses the most general formulation of a nonlinear model

$$\hat{x}_{k+1} = f(\hat{x}_k, u_k) \tag{9}$$

$$y_k = h(\hat{x}_k) \tag{10}$$

where the state function is $f : \mathbb{R}^{n_x + n_u} \rightarrow \mathbb{R}^{n_x}$ and the measurement function is $h : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_y}$. For the EKF, the model presented in (1) is already written in the required form. The algorithm of the EKF is [10]

- Prediction phase

$$\hat{P}_k = F_k \hat{P}_{k-1} F_k^T + Q_k \tag{11}$$

- Update phase

$$K_k = \hat{P}_k H_k^T (H_k \hat{P}_k H_k^T + R_k)^{-1} \tag{12}$$

$$\hat{x}_k^* = \hat{x}_k + K_k * (y_k - \hat{y}_k) \tag{13}$$

$$\hat{P}_k^* = (I - K_k H_k) \hat{P}_k \tag{14}$$

where $F \in \mathbb{R}^{n_x * n_x}$ and $H \in \mathbb{R}^{n_y * n_x}$ are the Jacobians of the state and measurement functions.

C. The Unscented Kalman Filter

The UKF uses the Unscented Transform (UT) [11] to account for the nonlinearity in the model. The current estimation of the state is treated as the mean value of a probability distribution, which has the same covariance as the states. Depending on the implementation, either $2n_x + 1$ (for a full order UT) or $n_x + 1$ (for a reduced order UT) points are chosen around the current mean. Each sigma point has a certain weight associated with it. There are multiple ways to choose the sigma points [12] [13].

The UKF uses the same model as in (1). The chosen (sigma) points are propagated through the state function. The new points are used to compute the new estimate of the mean, i.e. the state, and its covariance. The new points are also propagated through the measurement function, and their mean is the estimated output of the system [12]. The next steps are somewhat like the algorithm of the EKF.

The classical formulation of the UKF uses the square root of the state covariance matrix to compute the sigma points. To calculate the square root, the covariance matrix must be at least positive semi-definite, which is not guaranteed by the algorithm. A more stable version of the UKF, with a similar degree of complexity is the Square Root UKF (SRUKF). Its algorithm is [12]

- Choose the sigma points
  - Select the weights of the sigma points [13]

$$W_i = \frac{1 - W_0}{2n_x} \tag{15}$$

where W_0 is chosen arbitrarily. A positive value moves the sigma points further away from the previous estimate of the state, while a negative one brings them closer to the previous average. However, the weights must obey the condition

$$\sum_{i=0}^{2n_x} W_i = 1$$

- Compute the scaling parameters

$$\eta_i = \sqrt{\frac{n_x}{1 - W_i}} \tag{16}$$

- Choose the actual sigma points

$$\chi_{k-1_0|k} = \hat{x}^*_{k-1} \tag{17}$$

$$\chi_{k-1_i|k} = \hat{x}^*_{k-1} + \eta_i \hat{S}^*_{k-1} \tag{18}$$

where $i = \overline{1, n_x}$

$$\chi_{k-1_i|k} = \hat{x}^*_{k-1} - \eta_i \hat{S}^*_{k-1} \tag{19}$$

where $i = \overline{n_x + 1, 2n_x}$

- Prediction phase
  - Propagate the sigma points through the state function

$$\chi_{k_i|k} = f(\chi_{k-1_i|k}) \tag{20}$$

  - Compute the new state estimation

$$\hat{x}_k = \sum_{i=0}^{2n_x} W_i \chi_{k_i|k} \tag{21}$$

  - Calculate and then update the square root of the state covariance matrix

$$\hat{S}_{x_k} = qr([\sqrt{W_i}(\chi_{k_i|k} - \hat{x}_k) \quad \sqrt{Q_k}]) \tag{22}$$

for $i = \overline{1, 2n_x}$. ”qr” refers to the QR decomposition.

$$\hat{S}_{x_k} = cholupdate(S_{x_k}, \chi_{k0|k} - \hat{x}_k, sign(W_0)) \tag{23}$$

”cholupdate” is the rank 1 update. The rank update formula is $A1 = A \pm x * x^T$ where A is the matrix obtained through a Cholesky factorization (QR in this case) and x is a column vector. The sign to be used in the update is the one of W_0.
  - Propagate the ”state” sigma points through the measurement function

$$\mathcal{Y}_{k_i} = h(\chi_{k_i|k}) \tag{24}$$

  - Compute the new measurement estimation

$$\hat{y}_k = \sum_{i=0}^{2n_x} W_i \mathcal{Y}_{k_i} \tag{25}$$

- Update phase
  - Compute and then update the square root of the output covariance matrix

$$\hat{S}_{y_k} = qr([\sqrt{W_i}(\mathcal{Y}_{k_i} - \hat{y}_k) \quad \sqrt{R_k}]) \tag{26}$$

for $i = \overline{1, 2n_x}$

$$\hat{S}_{y_k} = cholupdate(S_{y_k}, \mathcal{Y}_{k_0} - \hat{y}_k, sign(W_0)) \tag{27}$$

  - Calculate the covariance between the states and the measurements

$$\hat{P}_{xy_k} = \sum_{i=0}^{2n_x} W_i(\chi_{k_i|k} - \hat{x}_k)(\mathcal{Y}_{k_i} - \hat{y}_k)^T \tag{28}$$

  - Find out the Kalman gain

$$\mathcal{K}_k = (\hat{P}_{x_k y_k}/\hat{S}^T_{y_k})/\hat{S}_{y_k} \tag{29}$$

  - Update the state estimation

$$\hat{x}^*_k = \hat{x}_k + \mathcal{K}_k(y_k - \hat{y}_k) \tag{30}$$

  - Correct the square root of the state covariance matrix

$$S_{x_k} = cholupdate(S_{x_k}, \mathcal{K}_k S_{\hat{y}_k}, -1) \tag{31}$$

V. Results and Discussion

The results of the comparison are shown and Fig. 5 - 7 and they are summarized in Table II.

The noises in the model are introduced by the way of functioning of the Simscape/Power Systems toolbox. Although different electrical components are modelled, they are not simulated as in a Spice-type program. All the electrical blocks and their connections are approximated by a state space model [7]. To assure stability of this model, it is also advised to use a fix-step solver with a very low sampling period.

The initial error of the SRUKF is not zero but is close to 10^{-4}. However, in time it quickly converges to $\approx 10^{-13}$. This is due to improper initialization, so the initial error is ignored.

As complexity, the KF and the EKF are the same. This is because the Jacobian of the state function can be computed in advance. It depends on ω_{m_k}, but so does the state matrix of the linear model. The SRUKF is by far the most complex.

The EKF is the fastest of the three filters, being closely followed by the KF and then, by a large margin, the SRUKF. The sigma point selection, the propagation of the $2n_x + 1$ points through the state function, the QR decompositions and the Cholesky rank update slow it down considerably. The slowdown of the KF might seem surprising. The linear model requires the computation of more mathematical operations - 27, in comparison with the nonlinear one - 21. As the algorithms are the same in rest, the slowdown is due only to the model.

All filter present oscillations. While the EKF and UKF assure a very low modelling error, the KF is plagued by rather

TABLE II
Comparison of the State Estimators

Estimator	Speed [% of EKF]	Maximum error	Complexity
KF	97.5	≈ 194	Low
EKF	100	$\approx 10^{-13}$	Low
SRUKF	40	$\approx 10^{-13}$	High

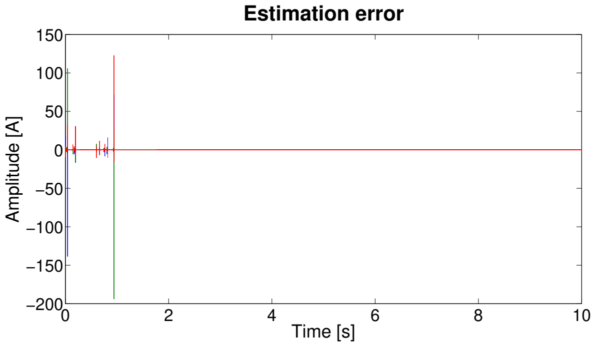

Fig. 5. Estimation error for the Kalman Filter

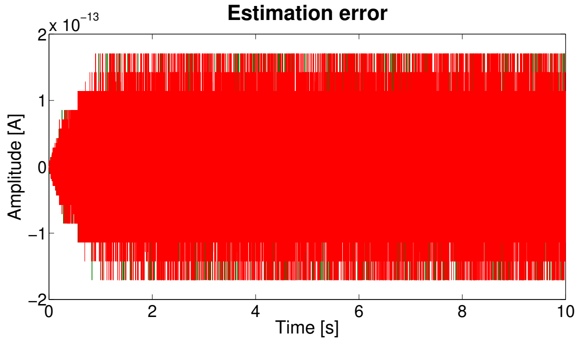

Fig. 6. Estimation error for the Extended Kalman Filter

large spikes. To understand what is happening, two zoomed in views of the estimation error of the KF are used: one before the large spike (Fig. 8) and one after the large spike (Fig. 9).

Before the large spike, the error of the KF oscillates but fairly slow, with certain pauses between each oscillation. In time, due to accumulation of energy, large oscillations appear, like the great spikes. After the large spike, the oscillation frequency has increased, so all the energy causing the previous large spikes is dissipated more quickly. A similar phenomenon can be seen for the EKF and UKF, where very small oscillation

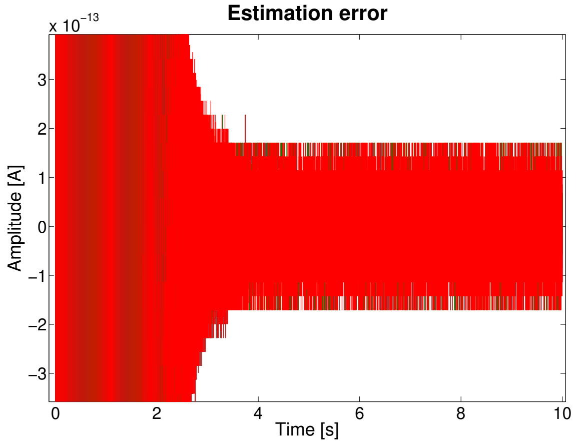

Fig. 7. Estimation error for the Square Root Unscented Kalman Filter

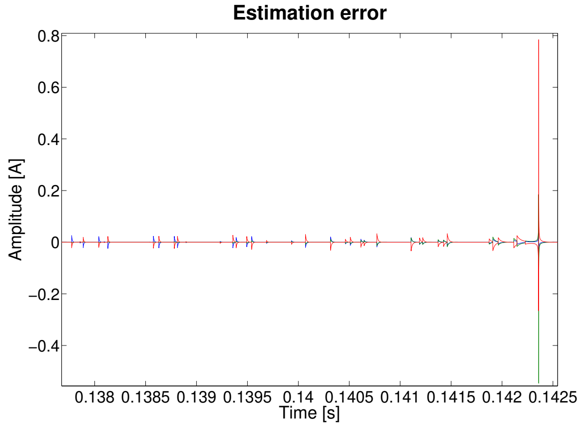

Fig. 8. Zoom in of the estimation error of the KF, before the large spike

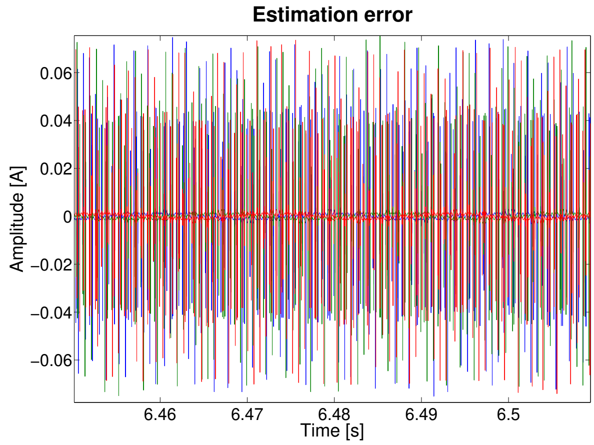

Fig. 9. Zoom in of the estimation error of the KF, after the large spike

are present, but with a very high frequency.

The cause of these oscillations is the approximation made by Simulink. The electrical model of the wind turbine (which includes a generator, a three level back to back converter, and RL filter and a voltage source which represents the grid) was made using the Simscape/PowerSystems toolbox. When the Simulink diagram is compiled, all the electrical model is approximated by a state space model. This introduces differences between the model used in the state estimators and the one used by Simulink. All filters try to compensate for this difference in a similar manner as a P controller. As the KF uses a linearized model which is even further away from the one used by Simulink, it is harder for it to achieve and maintain a null error. However, because both the state matrix used by the KF and the Jacobian of the state function used by the EKF depend on time, together with the intrinsic design (varying amplification and state covariance matrix) of the two filters, they manage to minimize the error. The UT transform helps the SRUKF to minimize the error. The EKF and SRUKF, as they use the nonlinear model, are better.

One might argue that both the EKF and the nonlinear model with a discrete integrator produce a similar estimation error (in the order of 10^{-12} and 10^{-13}), so the added complexity of the EKF is useless. However, when noise is added, the utility of the EKF is obvious (Fig. 10 and Fig. 11). Zero mean noise with a variance of one was added to the measurement of the voltages, which are used as inputs for the model.

VI. Conclusions and Perspectives

A. Conclusions

In the first part of this paper, it was proven the necessity for using a nonlinear model, in the case of a PMSG. A nonlinear model has an insignificant estimation error, while a linear one has an error with a amplitude around four. Moreover, it was shown that using a discrete integrator with a continuous model is the best approach.

Therefore, the resulting system is hybrid, having both a continuous part (the model) and a discrete one (the state estimator).

Three state estimators were compared: the KF, the EKF and the SRUKF. However, the EKF is about 2.5 times faster than the SRUKF and its error is in the order of 10^{-13} as the SRUKF, which can be reasonably approximated by 0. The KF could not compensate completely for the linearization of the model. Because the new model required more mathematical operations, it was also slower than the EKF.

The behavior of the different filters in the presence of the uncertainties generated by the functioning of Simulink and of the Simscape/PowerSystems toolbox was also examined. It was shown that the behaviors of the filters are similar to a proportional controller.

B. Perspectives

An interesting perspective would be the comparison of the presented state estimators with state observers. The best known nonlinear observers are the Nonlinear Unknown Input Observer (NUIO) and the Sliding Mode Observer (SMO). While Kalman-type filters are suitable to estimate the state, their behavior is undefined in fault situations. The NUIO is designed to consider the uncertainties in the process and the faults which can occur, together with their impact on the states. The SMO has very good robustness, which, theoretically, should ensure a good estimation. This comparison will be made in a future paper, while also considering model uncertainties (using a LPV model) and faults in the generator.

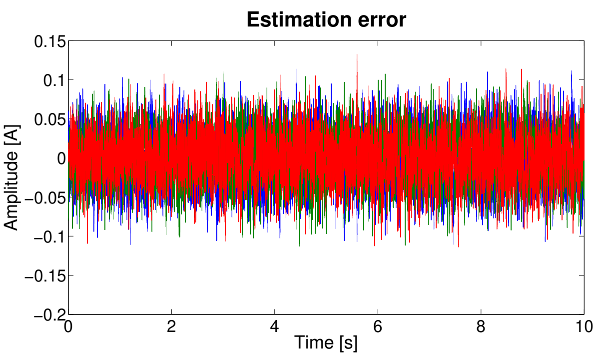

Fig. 10. Estimation error for the nonlinear model, in the presence of noise

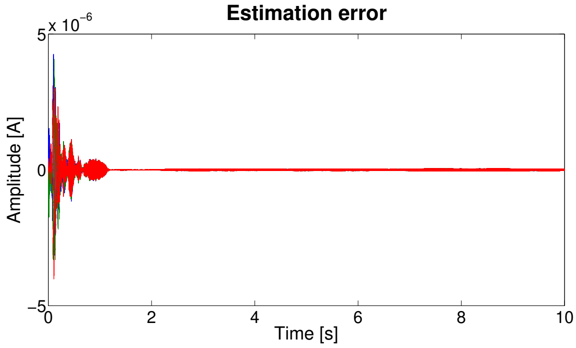

Fig. 11. Estimation error for the EKF, in the presence of noise

References

[1] G. Y. Kulikov and M. V. K., "Do the Cubature and Unscented Kalman Filtering Methods Outperform Always the Extended Kalman Filter?" in IFAC-PapersOnLine, vol. 50, no. 1, pp. 3762-3767, July 2017.

[2] L.I. Gliga, C. Lupu, H. Chafouk and D. Popescu, "Innovations in fault detection and tolerant control for a wind farm, using Wireless Sensor Networks", in Proceedings of the 18^{th} International Carpathian Control Conference, pp. 332-336, Sinaia, Romania, May 2017

[3] K. Alameh, N. Cite, G. Hoblos and G. Barakat, "Vibration-based Fault Diagnosis Approach for Permanent Magnet Synchronous Motors", in IFAC-PapersOnLine, vol. 48, no. 21, pp. 1444-1450, 2015

[4] L. I. Gliga, D. Popescu, H. Chafouk and C. Lupu, "Extended Kalman Filter for a Permanent Magnet Synchronous Generator", in the 14^{th} European Workshop on Advanced Control and Diagnosis, Bucharest, Romania, November 2017.

[5] F. Morel, X. Lin-Shi, J. M. Retif, B. Allard and C. Buttay, " A Comparative Study of Predictive Current Control Schemes for a Permanent-Magnet Synchronous Machine Drive," in IEEE Transactions on Industrial Electronics, vol. 56, no. 7, pp. 2715-2728, July 2009.

[6] S. A. A. Shahriari, M. Raoofat, M. Dehghani, M. Mohammadi and M. Saad, "Dynamic state estimation of a permanent magnet synchronous generator-based wind turbine," in IET Renewable Power Generation, vol. 10, no. 9, pp. 1278-1286, October 2016.

[7] Hydro Quebec, "SimPowerSystems User's Guide", The MathWorks, https://www.mathworks.com/help/releases/R13sp2/pdf_ doc/physmod/ powersys/powersys.pdf, 2003, Accessed 17.05.2018.

[8] D. Stefanoiu and J. Culita, "Kalman Filtering of Distributed Time Series," in Proceedings of the 17^{th} International Conference on Control Systems and Computer Science (CSCS), vol. 1, pp. 101-108, Bucharest, Romania, May 2009.

[9] L.I. Gliga, H. Chafouk, D. Popescu and C. Lupu, "Recursive Covariance Estimation for an Extended Kalman Filter", 2018 unpublished

[10] F. Gilbert, X.N. Zhang, D.M. Vilathgamuwa, "A Sensor Fault Detection and Isolation Method in Interior Permanent Magnet Synchronous Motor Drives based on Extended Kalman Filter", in IEEE Transactions on Industrial Electronics, vol. 60, no. 8, pp. 3485-3495, February 2013.

[11] R. Baesso Althof and M. Ferber, "Efficient uncertainty analysis of wind farms in the time domain using the Unscented Transform", in Proceedings of the 8^{th} International Symposium on Power Electronics for Distributed Generation Systems (PEDG), Florianopolis, Brazil, July 2017

[12] R. Van der Merwe and E.A. Wan, "The square-root unscented Kalman filter for state and parameter-estimation", in Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 6, pp. 3461-3464, Salt Lake City, UT, USA, May 2001

[13] G. Terejanu, "Unscented Kalman Filter Tutorial", Department of Computer Science and Engineering, University at Buffalo, Buffalo, New York, Web-tutorial , 2008.