Simulation-based Approach to Application Fitness for an E-Bike

Alexandra Iosub[1,2], Jerome Kirscher[2], Monica Rafaila[2], Andi Buzo[2], Georg Pelz[2], Liviu Goras[1]
[1] 'Gheorghe Asachi' Technical University of Iasi, Romania,
[2] Infineon AG Neubiberg, Germany
Alexandra.Iosub-EE@infineon.com

Abstract* — Apart from fulfilling component electrical specifications, electronics needs to fit into the target application, composed of electronics components as well as mechanical parts. In complex applications such investigations can be made by simulations in which relevant application performances are checked under any allowed variation of operating conditions, design parameters and noise factors. In this paper, we investigate verification methods for application fitness, on top of specification compliance. These are supported by a flow, which supports to assess a component's impact on the application performances. This flow is applied on an E-Bike application modelled in SystemC-AMS. The objective is to answer whether the assessed component reasonably serves the application in an environment with uncertainties. To accomplish this, concepts of experiment planning, metamodeling and sensitivity analysis are applied.Keywords: SystemC-AMS, PMSM, PI, E-Bike, DoE, Metamodeling, ANOVA, design robustness, Simulation

I. Introduction

Nowadays electronic components assessment tends to be extended towards a validation under real-life application scenarios.

The assessment of application fitness of an electronic component is not trivial especially in applications with control loops, where interaction of the mission profiles and all the component's properties play a significant role. We investigate an Electric Bicycle application (see the block diagram in Fig.1). An electric bicycle is a bicycle with auxiliary electric motor which supports the propulsion. A real setup is often inflexible and costly to analyze, therefore, a behavioral model of the system is used for investigations.

The goal is to determine if a candidate position sensor satisfies application requirements. To accomplish this, we investigate the impact of component performances onto application properties, but also assess the interactions between performances of the component and other application factors to predict the performances at the application level at different values of the factors i.e. variations under study.

In the following, we start with assessing if the component under study – the angle sensor – satisfies the system requirements for any allowed angle error combined with a specified variation of load inertia. DoE (Design of Experiments) methods are used to this purpose. Then, because in the first iteration the component doesn't satisfy all desired requirements, we implement an optimization step, making use of the two degrees of freedom of the PI controller in the speed loop. At this step metamodels are employed (mathematical approximations of the performances versus influencing factors) to express the relationships between the desired system performances, factors under optimization and factors expressing undesired variations and to predict optimal values.

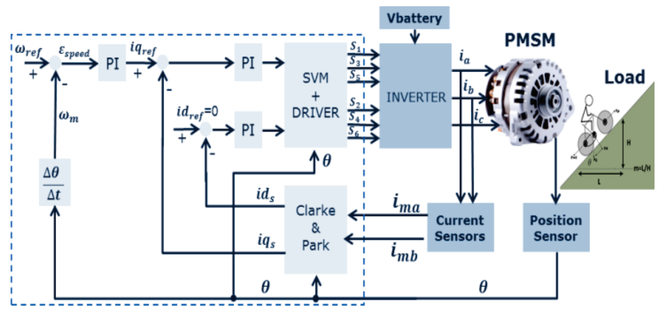

Figure 1. Diagram of an E-Bike

II. Related Work and Concepts

In this work we propose an overall approach for a motor drive application by modeling in SystemC-AMS [10] a Permanent Magnet Synchronous Motor (PMSM), driver circuits, sensors, and a microcontroller, where, some limitations and non-ideal effects are captured, e.g. saturation of the Pulse-Width Modulation (PWM) signals, Analog to Digital Converter (ADC) resolution, time frames of the motor currents sampling, interruption events, synchronization between the ADC and PWM blocks, main back EMF harmonics.

As far as modelling is concerned, investigations in this direction have been announced at various conferences in [1-4]. In [1], the authors introduce a novel method for modeling of a PMSM drive system based on Space Vector Pulse Width Modulation, where the simulation model of the complete system is built in Matlab/Simulink.

With respect to investigations of component impact, a hierarchical approach to analyze the effects of component parameters is presented in [2], where effects of parameter variations in one level of design hierarchy on those of the next are mapped through linear and piecewise linear sensitivity functions. The data is used to determine the critical circuit specifications that must be measured and those that may be eliminated from the testing process. In [3] the effect of parameter variation on the behavior of analog circuits at the transistor level is studied. There, the authors manage to characterize a safe subset of the parameter space for which the circuit can be guaranteed to satisfy the design specifications. They present a statistical model inference approach that exploits recent advances in statistical verification techniques and the approach uses extensive circuit simulations to deduce polynomials that approximate

978-1-4799-7250-0/16/$31.00 ©2016 IEEE

the behavior of the circuit. Paper [4] discusses verification and optimization of complex systems with respect to a set of specifications under stochastic parameter variations. Here the authors also refer to a system for a DC motor drive, where the best values for coefficients from controller should be found in order to fit the system performances. They introduce a simulation-based statistically sound model inference approach that considers systems whose responses depend on a few design parameters and many stochastic parameters.

Our approach uses concepts and methods as Design of Experiments (DoE) and Metamodeling [12]. Metamodels [11] are predictive models of the performances versus the parameters variation. When needed, a linear regression or extensions [13] can be applied, depending on the complexity of the estimated performance. The metamodels can be optimized in order to find regions of the design space which meet specific constraints or which are global extremes of some given objective functions. In this work, metamodels are used also for sensitivity analysis and optimization. The resulting metamodels provide useful information to predict values for some controllable factors, i.e. parameters which can be set during production, in order to fit the requirements into desired values while some uncontrollable factors i.e. variations which cannot be controlled during production or product use, are set to their worst case values, which are predicted using the metamodel.

III. System Description and Requirements

The system under study is an electric bicycle. Apart from the electric motor, the electrical part of an E-bike contains: a controller block, a power stage consisting of a transistor driver and an inverter circuit, sensors for electrical signals (three phase currents) and mechanical signals (motor position). All these components are presented in Fig.1.

The PMSM motion is controlled by applying the Field-Oriented Control (FOC) algorithm [15]. These control strategies use almost pulse width modulated (PWM) switching strategies that look to produce a precisely controlled current to the windings of the motor. The FOC aims to control both torque and flux to force the motor to accurately track the desired values. This control is performed by regulating the motor current, and it is mandatory to have the angle information for this. To apply the FOC algorithm the electrical equations are projected from a 3 phase non-rotating frame into a 2 co-ordinate rotating frame by using the mathematical transformations (Clarke and Park).

There are three PI regulators in the control system: one is for the mechanical loop (the loop that regulates the angular speed) and two are for the electrical loops (the loops that regulate the d and q currents). The corresponding outputs of the PI controllers (that represent the voltages to be applied to the motor, referenced to the stationary frame) are then passed through an inverse Park and Clarke transformation to convert the variables back to the 3-phase stator reference. The 3-phase voltages (which are continuous sine waves) are finally converted to PWM signals using the Space Vector Modulation (SVPWM) technique [15]. The main objective is to approximate the reference voltage vector utilizing eight switching patterns.

For motor position's detection a sensor based on the giant magnetoresistance (GMR) effect is used. The sensor converts the motor angle to two voltages that represent the sine and the cosine of the current motor angle.

The angle sensor modeling focuses only on the sensor parameter characteristics, which could have impact on the control of the motor, such as: offsets in the sensor output voltages V_X, V_Y, the synchronicity error, and mechanical misalignment and inorthogonality of the GMR structures. After analog to digital conversion, the angle information is reconstructed inside the control algorithm by using the arctangent function which calculates the digital angle value as: $\theta = tan^{-1}(\frac{V_Y}{V_X})$.

The motor position is required for the SVPWM technique and also for the motor speed reconstruction. The angular speed is calculated as the derivative of the angle with respect to time: $\omega = \frac{d\varphi}{dt}$

We focus on the angle error impact on the torque response while the torque depends on the second derivative of the motor angle (Eq.1). As it is already known, numerical differentiation is an unstable operation, and should be taken with great caution because it can amplify noises.

$$T_e(t) = T_L(t) + J \cdot \theta''(t) \tag{1}$$

where J is the total system inertia (load inertia plus motor inertia) which behaves as an amplifier for the error induced by the angle sensor. We noticed here that a large variation of load inertia is caused by the weight of the cyclist and the bicycle or/and the possibility to use a power transmission equipment such a gear box. Therefore, we assess here two possibilities: one is the use of a gearbox, case where the total system inertia will be reduced ($J_r = \frac{J_l}{N^2}$, J_r is the reflected inertia, J_l is the load inertia and N is the gearbox reduction ratio); The second case addresses new trends in such applications which consists of using a direct driven motor and the system has to behave properly with the total amount of inertia.

For an E-Bike application, some of the system responses i.e. performances under optimization, are defined as follows:

1. Ripple in torque response:

$$\mathrm{T_{e_{Ripple}}}[\%] = \frac{\mathrm{T_{e_{max}}} - \mathrm{T_{e_{min}}}}{\mathrm{T_{e_{avg}}}} \cdot 100 \tag{2}$$

where, $T_{e_{Ripple}}$ is the electrical torque ripple. $T_{e_{max}}$, $T_{e_{min}}$ and $T_{e_{avg}}$ are the maximum, minimum and average value of electrical torque signal. All these quantities are measured in steady state operation.

2. Acceleration time:

$$tRWm = t_2|_{W(t_2)=0.95 \cdot W_{m_{avg}}} - t_1|_{W(t_1)=0.1 \cdot W_{m_{avg}}} \tag{3}$$

$tRWm$ is the acceleration time (or rising time for speed signal) and is the elapsed time, measured in transient, between two pre-defined values of the speed signal. $W_{m_{avg}}$ is the average value of mechanical speed, measured in steady state operation. The limits (TableI) for the responses defined above are established in the context of high performances of the application. Those requirements are introduced at the beginning of assessment phase of a given component, as how the flow from Fig.2 indicates.

TABLE I. PERFORMANCES LIMITS

Ripple in Torque	0% < TeRipp ≤ 20 %
Acceleration time	2.2 [s] ≤ tRWm ≤ 4.5 [s]

For an E-Bike, high performances are: fast acceleration time and low noise and vibration, whose main cause is the level of ripple in the electrical torque [17].

IV. ASSESMENT FLOW

The proposed assessment flow using DoE methods and Metamodels is illustrated in Fig.2. The flow starts from the Experiment Plan phase. At this phase the experimental factors and responses (the application performances) and the objectives for them, are set. Then, an appropriate design of experiment based on factors number and resources is chosen. The design table can be a full factorial design but the number of simulations will be large (n^k simulations for k factors with n levels). Therefore, we apply a combination

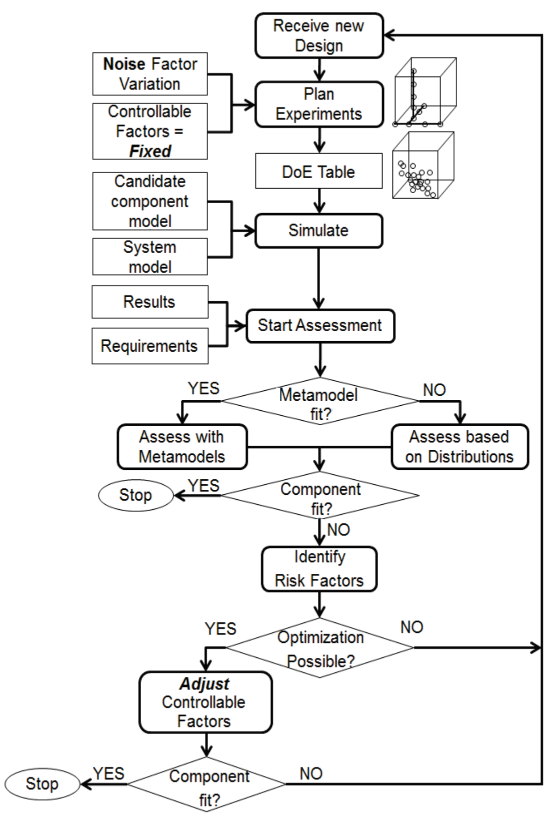

Figure 2. Assessment Flow

between One Factor at a time design for identification of high individual effects, combined with a Latin Hypercube sampling, for a higher coverage.

The number of runs is chosen high enough to account for all factors interactions. The experiment is applied on the simulation model. After the experiment planning, all simulations are done and the results are collected. Next, in the first step of the assessment phase, a metamodel for the responses is built on the main factors identified from the sensitivity analysis step, which is an ANOVA based method to indicate factors probable to have high contribution in the output variance.

We use a regularized linear regression augmented with non-linear basis-functions and interactions, because not only it is able to handle complex shapes, but is also fast to learn and evaluate [16]. First, the factors x_i are re-scaled to the range $[-1; 1]$ in order to give them similar weight and to simplify the following steps: $x_i = 2\frac{x_i - x_{i,min}}{x_{i,max} - x_{i,min}} - 1$.

The basis for the metamodel is a standard least-squares linear regression:

$$\hat{y} = \beta_0 + \sum_{i=1}^{n_F} \beta_i x_i + \sum_{j=1}^{n_F} \sum_{i=1}^{n_F} \beta_{ij} x_i x_j + \varepsilon_i \quad (4)$$

where $\hat{y}$ is the regression model (dependent variable), x_i, x_{ij} are the design factors (independent variables), β_i, β_{ij} are the regression coefficients and ε_i is the model error.

However, instead of just using the factors as predictors, we also add simple non-linear transformations of the factors:

$$x_i \to \{x_i, x_i^2, \sqrt{x_i - 1}, \sqrt{1 - x_i}, e^{x_i}, e^{-x_i}, \cdots\} \quad (5)$$

Additionally, we add all interactions between two non-linear predictors to the regression equation:

$$\{x_i, x_j\} \to \{x_i, x_j, x_i \cdot x_j\}.$$

The resulting metamodel is still linear in the coefficients β and thus a least-squares linear regression quickly fits the metamodel to the samples [16]. The regression coefficients are given by:

$$\beta = (X'X)^{-1}X'y, \text{ where } X \text{ is the design matrix.}$$

If the metamodel is fit the component assessment can be completely done based on this metamodel. In our analysis, we obtained a fit metamodel for each response. Therefore, next assessment is done by applying the step from left side of the flow - "Assess with Metamodels". The right step - "Assess based on Distributions" will not be discussed here.

The next step is to check if the component under study satisfies the system requirements. If this is accomplished, then, the assessment flow can finish. If not, a next step will be to identify the risk factors (factors which cannot be constrained). In the “Adjust Controllable Factors” step the controllable factors are adjusted in order to satisfy the desired requirements. Here, the risk factors must be set at their worst case level (a level which causes the worst case values of the system performances). After the optimization step we have to check again if the candidate component satisfies the system requirements. If it does, the assessment flow finishes, otherwise, we readjust the ranges of the factors.

V. EXPERIMENT RESULTS

We show that it is possible to model, validate and optimize a complex system using only one description language, SystemC-AMS (an open-source C++ library). The simulation of one run in SystemC-AMS is between 5-10 min compared to about 1h in VHDL-AMS. Also, we are able to simulate both digital and analog blocks simultaneously (because in Matlab/Simulink, for example, is more difficult to perform microcontroller models).

A. Assessment of angle error impact on electrical torque with load inertia variations

First simulations are run at the nominal case, where all components are ideal (no sensor errors). This ideal test scenario is applied here in order to check if the control algorithm is working properly. A step is applied in the speed reference signal from 0 to 24 $[\frac{rad}{s}]$, while the human torque is kept constant at 20 $[Nm]$. The load inertia is set at $J_L = 6$ $[Kgm^2]$ and the values of the control factors are chosen to ensure stability, small rising time and small overshoot. The obtained electrical torque response is a smooth waveform and the acceleration time is fast (2.2 [s]). From first simulations the individual effect of the system inertia (*fJ*) was identified to have the most important individual effect, when the rest is fixed.

Next, we focus only on the impact of the error from the angle sensor and load inertia variation onto application performances. First, we want to assess if the component – angle sensor- satisfies the system requirements (Table 1), for any angle error $\leq 0.9^{\circ}$ combined with a variation of load inertia from 0.01 to 5 $[Kgm^2]$. DoE methods are used for this purpose. In addition to the experimental results, the theoretical equation (Eq.1) confirms the dependency between torque ripple and error in the angle (the torque depends on the second derivative of the motor angle).

An experiment with uniform distribution variations is made for inertia and the angle error. In this first assessment the control factors (coefficients of PI controller adjusted through *fWlpf* and *fDamp*) were kept constants (at their first tuning values from ideal case). The factor denoted *fWlpf* is the bandwidth of a digital low pass filter, while the factor *fDamp* is called “damping factor” because has the effect on system phase margin. The load torque, reference speed and human torque are kept constant (with values corresponding to the normal environmental conditions). Fig.3 is a matrix plot which shows the values as scatter plot of each variable against each other, in each box. The main diagonal is the distribution of the parameter. A strong correlation is indicated by a pattern in the scatter plot. It can be seen that ripple in torque response is strongly affected by the angle error (*AErrDeg*), while the inertia (*fJ*, in $[Kgm^2]$) has an slight exponential effect only for low values (up to 2) and for values higher than 2 the effect is saturated. A clear direct relationship between acceleration time (denoted with *tRWm*) and inertia can be seen also in Fig.3. Metamodels are built

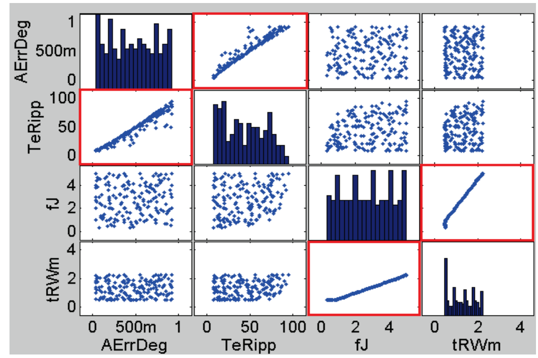

Figure 3. Matrix of scatter plot of the responses and factors

in order to predict deviations from the target requirements compared to the initial tuning of the control i.e. optimized for the case with no deviations of other parameters.

For fitting, the General Regression method is used. Fig. 4 is a 2D representation of the metamodels, where the factors *AErrDeg* and *fJ* are set to their worst case values. These snapshots provide an understanding of the first order effects but have the risk of missing higher order interactions. Therefore interactive plots are used that update

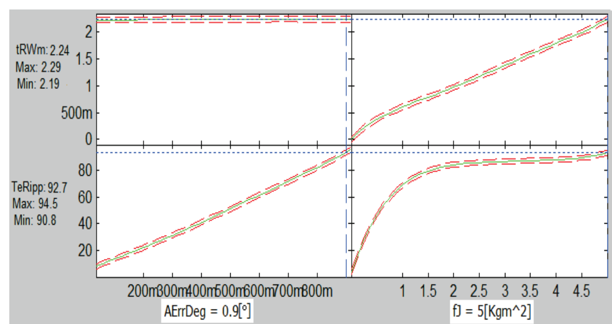

Figure 4. 2D plot of Acceleration time and Torque Ripple Metamodels

response-factor dependency according to the setting of the rest of the factors. The metamodel fitness is validated by a 10-fold cross validation (only 90% of the samples are used for training, rest of 10% for validation). The maximum normalized to the response residuals are: 0.065 for *TeRipp* and 0.016 for *tRWm*. The overall R squared is 0.998 for *TeRipp* and 0.999 for *tRWm*, meaning that about 99% of the overall variance in the data is described by the metamodels. Therefore, the metamodels can be used for predictions, sensitivity analysis and optimization.

We can assess the impact of inertia combined with the maximum angle error in at least two conditions: first, the application will not use a gear box and inertia is high (e.g. $fJ=5[Kgm^2]$). According to the metamodel, torque ripple can be up to 90% and it can be fit into application requirements for an angle error $< 0.2^{\circ}$, but such constraint for an angle error will lead to an expensive angle sensor.

If there is a low inertia mismatch for the application (the load is coupled by a belt or a chain drive and *fJ* $= 0.01[Kgm^2]$) and the torque ripple is about 60% and for an angle error $< 0.6^{\circ}$ the ripple can be reduced to 20%. The prediction of $tRWm$ response emphasizes that for any values of inertia or of angle error, only the upper limit requirement (4.5[s]) is satisfied, however, the lower limit (2.2[s]) not. Since the torque ripple and acceleration time performances are not satisfactory, the next question is if the optimization is possible, according to the flow presented in Section IV.

Further analysis is focused on the first case: an E-Bike with a direct driven motor.

B. Optimization by tuning of the PI's coefficients

In the next step, the PI's coefficients are tuned considering variations of two factors: *fWlpf* and *fDamp,* which directly lead to a change in *Kp*, *Ki* control factors (proportional and integral gains) of the speed loop, in order to fit the torque ripple within the maximum limit. A Full Factorial design is applied in order to see also higher order interactions. The factors used in design are shown in Table II. These values can be optimized by an optimization applied on the metamodel.

TABLE II. FACTORS TYPES

Controllable Factors	Uncontrollable Factors
fWlpf	*fJ*
fDamp	*AErrDeg*

TABLE III. FACTOR RANGES

fJ [Kgm^2]	min = 4.5	max = 5.8
fWlpf [rad/s]	min = 20	max = 110
fDamp[-]	min = 100	max = 150
AerrDeg [deg]	min = 0.001°	max = 0.9°

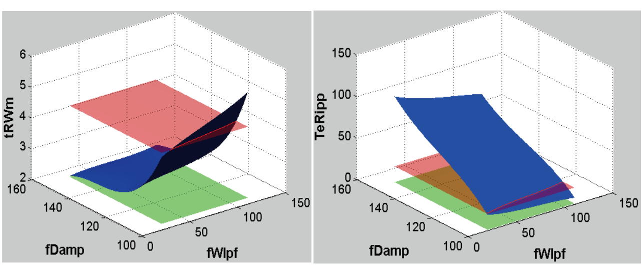

Figure 5. 3D plot of Acceleration time and Torque Ripple Metamodels: Blue- the response; Green- lower limit; Red- upper limit

For this design 3 levels for each of the 4 factors (Table II) are used: *fJ, fWlpf*, *fDamp* and *AErrDeg*. Now, the inertia was varied only between 4.5 and 5.8[Kgm^2] (direct driven motor). The control factors reveal a trade-off between torque ripple and the acceleration time (Fig.6): when *fWlpf* increases, $TeRipp$ increases and $tRWm$ decreases. Fig. 5 shows the performances versus factors in a 3D representation as surface plots. Both responses ($TeRipp$ and $tRWm$) are represented in the control factors plane (*fDamp* and *fWlpf*) while the angle error and inertia are kept constant at their worst case values (in order to ensure that at the worst case situation the requirements are satisfied): $AErrDeg$ $=0.9^{\circ}$, *fJ*$= 5[Kgm^2]$. The metamodels are built by using the General Regression method. For each factor two terms are found: linear and quadratic terms. The maximum normalized residuals are: 0.059 for $TeRipp$ and 0.013 for $tRWm$, while the R squared is 0.994 for $TeRipp$ and 0.999 for $tRWm$.

For the 2D representation (Fig. 6), the control factors are set to the levels indicated in the text box and by the dotted vertical slide bar. The torque ripple is lower than 20% when *fDamp* is higher than 130 and *fWlpf* is lower than 25, but at this values of the control factors also the acceleration time achieves its limit (4.5 [s] for inertia fJ=5 [Kgm^2]). Based on the fitted metamodels two contour plots are drawn in Fig. 7 which are the intersection with the spec limit planes ($TeRipp$, $tRWm$) in a 2-dimensional plane ($AErrDeg$, *fJ*).

In Fig.7, the regions under the continuous lines are safe subsets of factor values predicted by metamodels. Fig.7 is an example that shows how the tuning can change the performance and fit to the specification limits. At the left side, the performances TeRipp and tRWm are visualized for controllable factors *fWlpf*=100 and *fDamp*=110 and the right side shows the case after tuning, where control factors were fixed at *fWlpf*=63 and *fDamp*=130.

It can be seen that, on the left side, only for a certain combination of uncontrollable factors (AErrDeg and fJ) we can satisfy both performances. However, the safe (green) area is too small (top left corner) and the acceleration time response is close to its lower limit (2.2 [s]) and far away from the upper limit (4.5 [s]).

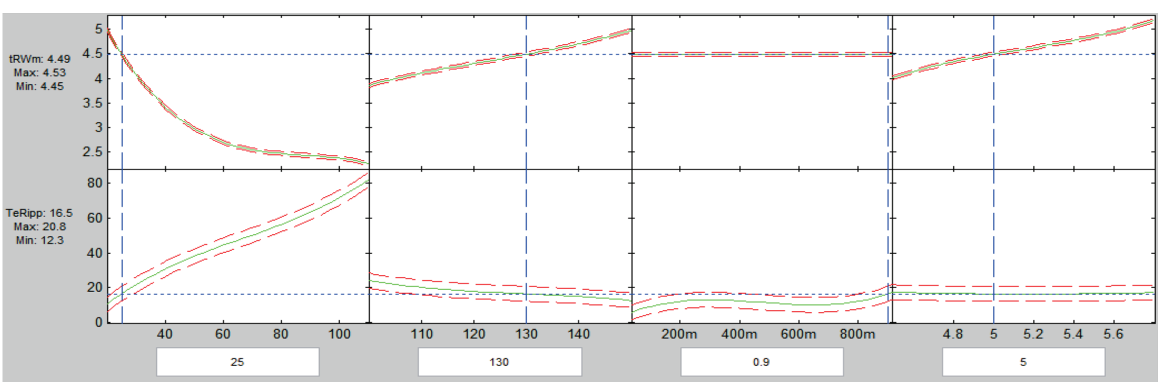

Figure 6. 2D plot of Acceleration time and Torque Ripple Metamodels

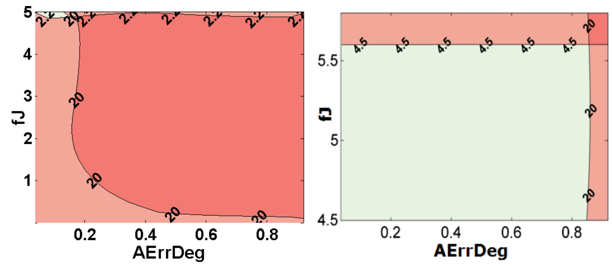

Figure 7. Contour plots of a feasible region in the controllable factors space for *TeRipp* and *tRWm* before tuning (on the left) and after tuning (on the right). It can be observed that after tuning requirements are satisfied at a larger region of the factors' space (AErrDeg and fJ).

After tuning of PI's coefficients, a large number of combinations between the uncontrollable factors gives a safe region (where both of performances are satisfied) for *TeRipp* and *tRWm.* (right side of Fig.7).In this case, the acceleration time reaches the upper limit and is far away from the lower one. The results show that the component under study (Angle Sensor) has a great impact on the system response, namely ripple in torque. Moreover, the interaction between these two factors (system inertia and angle error) becomes significant when both of them have large values. The metamodels' fitness and prediction power justify their use. As a risk factor we have identified the system inertia (factor which can be constrained only if an additional component is added in the application – a gear box, and; to limit its impact, its value must be significantly reduced). The optimization is done by using new metamodels where the system control factors have been included. After this optimization, the system requirements are satisfied for the full range of the angle error at the high level of the system's moment of inertia.

When it comes to the number of the simulations required to obtain the presented results, our approach based on metamodels required only 81 runs (= 3^4, i.e. 3-level full factorial of four factors). Without using the prediction of the metamodels, in order to obtain a comparable assessment and optimization quality, one would need 10000 runs if a full grid of 10 levels with 4 factors is used for verification. Other analysis types such as Monte Carlo, which do not do prediction, but only estimate the variance of the response, would still need hundreeds of runs.

VI. CONCLUSIONS

Design of Experiments and Metamodeling concepts are applied in order to answer if the component – angle sensor - satisfies the system requirements for a variable angle error within a given range combined with a variation of load inertia, in the case of an E-Bike. The proposed flow ends with a positive answer only after the optimization step, where, by using the metamodels, we can predict the values of the application performances as function of controllable ($fWlpf$, $fDamp$ which set the mechanical loop PI's coefficients) and uncontrollable (*AErrDeg, fJ*) factors. The feasible space of performances is qualitatively and quantitatively described under different system variations and the preferred trade-off should be mapped to the system settings and expected behavior. The metamodels reveal strong interaction effects, as the effect of one factor depends on the other. The factors and their interactions play a role as it can also be seen from the analysis. The metamodels are also used for optimization ($TeRipp$, $tRWm$). For the given modeled application and for the desired requirements, an angle sensor with error not higher than 0.9° can be used.

REFERENCES

[1] L.Ting-Ting et al. "Simulation of PMSM vector control system based on Matlab/Simulink." Measuring Technology and Mechatronics Automation, 2009. ICMTMA'09. International Conference on. Vol. 2. IEEE, 2009.

[2] H.Yoon, P.Variyam, A.Chatterjee, and Nagi, N. "Hierarchical statistical inference model for specification based testing of analog circuits." VLSI Test Symposium, 1998. Proceedings. 16th IEEE. IEEE, 1998.

[3] D.Zhang, Y.Zhang, et al. "From statistical model checking to statistical model inference: Characterizing the effect of process variations in analog circuits." Proceedings of the International Conference on Computer-Aided Design. IEEE Press, 2013.

[4] D. Zhang,Y.Zhang, et al. "Statistically Sound Verification and Optimization for Complex Systems." Automated Technology for Verification and Analysis. Springer International Publishing, 2014. 411-427.

[5] Jingyu Liu and Xueli Li and Hui Liu and Zhong Hu and Muyuan Gao, "Handling stability simulation analysis of Electric Power".

[6] Lee Stephens, Senior Motion Control Engineer Kollmorgen, "The Significance of Load to Motor Inertia Mismatch", Available: www.kollmorgen.com

[7] Dave Wilson, Motion Products Evangelist, Texas Instruments, "Teaching Your PI Controller to Behave". Available: https://e2e.ti.com/blogs_/b/motordrivecontrol/archive/2015/07/20/teaching-your-pi-controller-to-behave-part-i

[8] Jingyu Liu, Xueli Li, et al; "Steering (EPS) system based on ADAMS", Electric Information and Control Engineering (ICEICE), 2011 International Conference, 2011.

[9] C. Grimm, M. Barnasconi, A. Vachoux, K. Einwich, An Introduction to Modeling Embedded Analog/Mixed-Signal Systems using SystemC AMS Extensions, System C, 2008, p. 12, Available: http://www.systemc.org/

[10] Fan, Xuan; "Intelligent Power Assist Algorithms for Electric Bicycles." (2010).

[11] T. W.Simpson, J.D.Peplinski, et al, "Metamodels for Computer-based Engineering Design: Survey and recommendations. In: Engineering with Computers" 17 (2001), p. 129-150.

[12] D. Montgomery; "Design and analysis of experiments"'; John Wiley and Sons, 2005.

[13] Myers, R.H., D.C. Montgomery, G.G. Vining, and T.J. Robinson: Generalized linear models: with applications in engineering and the sciences, volume 791. Wiley, 2012.

[14] "SystemC-AMS"; Available: http://www.systemc-ams.org

[15] A.Veltman, D. Pulle and de Doncker, R. (2007), "Fundamentals of Electrical Drives", Springer Netherlands.

[16] Markus Dobler, Wolfgang Rosenstiel, Martin Bogdan, Eberhard Karls; "Rapid Design Space Exploration of a State-of-the-art PSI 5 Controller"; Universität TübingenTübingen

[17] Bosch-eBike-Manual-Active-Line-MY15WEU.pdf; Available: http://www.bosch-ebike.de/media/pdf/bedienungsanleitungen_0315/Bosch-eBike-Manual-Active-Line-MY15WEU.pdf