Proceedings of the 2010 Forum on specification & Design Languages

Southampton, UK
September 14th-16th, 2010

General Chair:

Prof. Tom Kazmierski
School of Electronics and Computer Science
University of Southampton

FDL is an ecsi event!

FDL 2010

Table of Contents

Design of Experiments for Reliable Operation of Electronics in Automotive Applications

Monica Rafaila, Christian Decker
Automotive Power, Infineon AG
Neubiberg, Germany
Email: monica.rafaila@infineon.com

Christoph Grimm
Vienna University of Technology
Vienna, Austria
Email: christoph.grimm@ict.tuwien.ac.at

Jerome Kirscher, Georg Pelz
Automotive Power, Infineon AG
Neubiberg, Germany

***Abstract*—Starting from pre-sillicon verification steps, automotive electronics demand for the assurance of a high degree of reliability. A dependable system must comply with requirements even in the presence of sources of variation, which can come from internal or external influences. Therefore, one must study the impact of such variations on the responses of the system and find safe margins for them. The problem amplifies when the effects of variations and the conditions which drive the system into a worst-case change in time.**

Statistical Design of Experiments and metamodelling strategies are adopted and extended to cope with the problem. A reduced number of simulation runs is found sufficient to characterize time-variant effects and to predict the system's behaviour between the simulated points. Then, the bounds of the signals of interest are estimated and confirmed. The proposed methods are demonstrated on a window lifting automotive system. [1]

I. Introduction

The ever increasing complexity of automotive ECUs (electronic control units) introduces numerous sources of variation with potential impact on the behaviour of the system. There are no levels guaranteed for the parameters of components, either internal of external to the system, or for environment conditions. The existence of tolerances, safety margins, or simply the room left for later design decisions introduces the need to validate in an early stage the system against requirements in the presence of the above mentioned influences [11]. They vary in pre-specified windows and form a multi-dimensional, continuous verification space, which must be optimally covered before moving on to the next design step.

Great effort is invested into model-based verification and into solutions which speed-up simulation, e.g. modelled using the SystemC specification language. But this is just a first step towards more efficiency in covering the verification space. Models can and should be resimulated until the conformance of the system is validated under any factor condition, be it an internal parameter variation or a external influence. New methods are needed, which start from the basic simulation flow, and extend it to perform multivariable sensitivity and worst-case analyses. A reduced number of simulation runs must be invested to characterize the time-variant impact of variations and extreme system behaviour.

[1]This research project is supported by the German Government, Federal Ministry of Education and Research under the grant number 01M3178*. The authors are responsible for the context of the paper.

In the context of the present work, the factors are input variables in the simulation process (operating conditions, parameters of functional parts). The response is a quantity of interest for system verification, which can be extracted after each simulation. Experiments are sets of simulation runs which apply variations in the factors to identify reasons for changes in the response.

Design of Experiments (DoE) is an approach to plan and analyze real life as well as simulated experiments [5]. DoE can handle the complexity of a highly dimensional factor space by investing a reasonable number of tests. DoE is applied here for deterministic simulation experiments, where all factors are controllable and under investigation, to study the impact functional blocks and their interdependencies have on the system outputs.

Since simulation tests the response in time of the system under predefined stimulation, it is desired to extend the DoE approach for responses which vary in time. The target system behaviour is quantified in signals i.e. transient responses. Carefully designed experimentation is followed by a postprocessing step which extracts a predictive response metamodel. It characterizes effects of factors on the response, and how they vary in time. After validating the metamodel as a response prediction, response extremes are estimated for the simulated time frame.

An ECU designed for window lifting applications, within its application context, presents numerous factors which vary in specified ranges. Applying the proposed flow shows it is a time effective, still reliable, multi-run strategy to discover response bounds and understand the main causes for worst-case behaviour.

II. Related work

Previous work addresses similar issues, by worst-case and sensitivity analyses. Extended Monte-Carlo methods such as [10] use Importance Sampling to explore the verification space in search for the worst-case. However, for a highly dimensional search space they require a significant amount of runs ($> 1k$).

Approaches based on Affine Arithmetic e.g. [3] compute bounds for system outputs. These are safe, but over-pessimistic, when dealing with complex factors effects. Moreover, documentation and software implementations are hardly available. [2] presents methods to compute inner and outer uncertainty bounds, using the Genetic Algorithm and Affine Arithmetic. It deals, however, only with a reduced set of factors. Assumptions on the system response need to be made and checked, to reduce the complexity of the problem.

DoE is used as starting point in the approach, to improve the trade-off invested effort versus extracted information. Concepts from the DoE methodology are introduced next [5].

DoE assumes that the response dependency on the factor set f can be approximated with a multivariate low-order polynomial metamodel $R(f)$:

$$R(f) = c_0 + \sum_{o=1}^{2}\sum_{i=1}^{n} c_i^{(o)} \cdot f(i)^o + \sum_{j=1}^{n-1}\sum_{k=j+1}^{n} c_{jk} \cdot f(j) \cdot f(k)$$

Each $f(i)$ is normed and centered on 0, so the verification space is $[-1; 1]^n$. The coefficients quantify the effects of factors on the response: $c_i^{(1)}$ - main effects, $c_i^{(2)}$ - quadratic effects, c_{ik} - 2-factor interaction effects. The assumption is based on the sparsity of effects principle which often applies in practice, i.e. that the system is likely to be driven primarily by some main effects and low-order interactions.

Statistical DoE plans experiments with minimum runs, in order to find factor effects and how they interact, with maximum statistical confidence. Regression extracts the coefficients which best fit simulation data. Classical DoEs allow optimal regression by extracting results which enable decoupling the factor effects. They work in the assumption that estimated effects are significantly higher then the not accounted effects:

- Fractional Factorial DoEs consist of selected factors' corners, and are widely used to investigate low-order factor effects. The basic ones are Resolution 3 (R3FF), which estimates all main effects, and Resolution 5 (R5FF), which can extract main and interaction effects.
- Response Surface DoEs additionally estimate 2^{nd} order effects. Central Composite DoE (CCD) adds to Fractional Factorials the center and 2 axial points per factor.

Space filling DoEs can improve the metamodel fitness at the cost of more simulation runs. E.g. a Latin Hypercube Sampling (LHS) generates random factor levels, e.g. normally distributed, and maximize the minimum distance between points in the factor space. Software packages generate the tests required by such DoE [4].

To evaluate the fitness of the metamodel as a response estimate, residuals, i.e. differences between results in the simulated points and estimates, are analyzed. To conform, they must be small enough, approximately normally distributed with mean zero and not correlated to the response values.

DoE has been successfully applied to control simulations of parameterized systems, for various verification purposes: screening [13], sensitivity analysis [6], robust design [1], or multi-objective optimization [12]. Previous work found DoE able to detect important effects and response extremes, for static responses in automotive applications [7], [8]. For more details on metamodelling methods, [9] provides an extensive review.

III. EXTENDED EXPERIMENT FLOW

A. Problem description

The concept of a response must be extended to the set of time-value pairs of the signal of interest. The problem can be formally described as: given a vector of factors f of length n and a response signal $r(f, t)$ $(t = time)$, the dependency on f and bounds of r with respect to f must be estimated. The effects of f can vary in time, and so can the factor sets which determine global extremes of the response. For each simulation, f is kept fixed and $r(f, t)$ can be recorded, at least for a set of time samples S.

It is assumed that the dependency of r on f can be approximated by a polynomial metamodel $R(f, t)$:

$$R = c_0(t) + \sum_{o=1}^{2}\sum_{i=1}^{n} c_i^{(o)}(t) \cdot f(i)^o + \sum_{j=1}^{n-1}\sum_{k=j+1}^{n} c_{jk}(t) \cdot f(j) \cdot f(k)$$

The assumption will be checked for correctness after processing the simulation results, in the Metamodelling step. With respect to t, the sampled r represents a set of highly correlated static responses, because they are values of the same signal, influenced by the same set of factors. Ideally, the factors with major impact remain the same over the time interval under study. But since different events occur at different times, not only the constant term c_0, i.e. the value when all factors are zero, modifies in time, but in reality so can the effects of factors and their interactions.

For simplification, the sample points are equidistant: $S = \{t = p \cdot T; p = 1...s\}$, where s is the number of samples (the simulation duration is at least $s \cdot T$).

Metamodelling must estimate for each time sample the set of coefficients $c(t) = \{c_0(t); c_i^{(o)}(t); c_{jk}(t)\}$, where $o = 1, 2; i = 1...n; j = i...n-1; k = j+1...n$. The metamodel must be validated as described above, but for each time sample. The relative residuals, i.e. normed to the maximum response variability, are analyzed. They also vary in time:

$$\epsilon(f, t) = (r(f, t) - R(f, t))/(max_f R(f, t) - min_f R(f, t))$$

B. Flow

Figure 1 shows the steps of the approach.

A prerequisite of the experiment is that the DUT model is available and all potential factors are variables coded in the DUT model (as internal parameters) or its test bench (as external blocks' parameters or stimuli). They are run-time configurable by a controller, which also reports the results.

Given a response of interest, a set of factors under variation and their ranges are first extracted from the spec. and provided to the controller. The experiment planning and analysis is performed on factors normed to the mid value of their ranges and centered on 0.

1) Design the experiment: A DoE is generated, depending on the number of factors n which must be handled and the number of runs m which can be invested. A DoE matrix d of size $m \times n$ is loaded to be simulated. The first run is always the center point of the factor space ($f = 0$). This run provides a reference for each time sample.

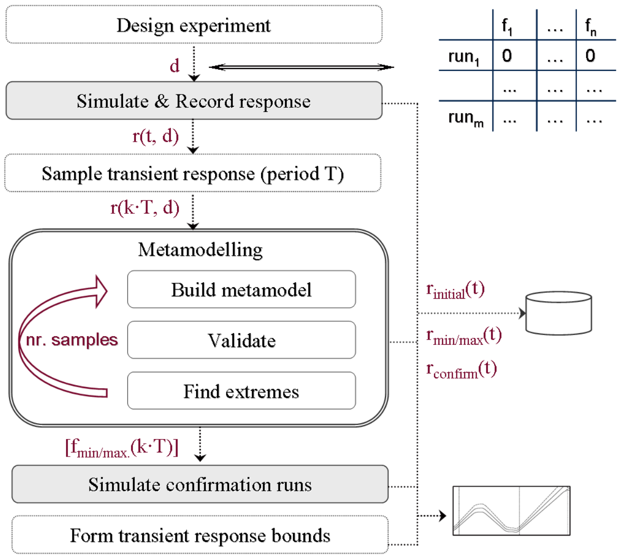

Fig. 1. Transient response experiment flow

2) Simulate the experiment: The set of runs is simulated. The response signal is recorded for the complete duration of each simulation run, with a predefined time resolution.

3) Sample the response: The traces are loaded into the postprocessing algorithm and a matrix of samples is formed: $[k \cdot T; r(k \cdot T, d(1,:))...r(k \cdot T, d(m,:))], k = 1...s$, where $r(k \cdot T, d(i,:))$ is the response result for run i, at time sample $k \cdot T$. $d(i,:)$ is the row i of DoE matrix, i.e. with the factor settings from run i.

The failed runs are filtered out and the outliers are replaced with NaNs. When too many runs failed, the controller aborts the analysis and optionally reruns with reduced factor ranges. The outliers, on the other hand, are removed only for the respective time samples. For instance, a response signal which ramps-up and is triggered at different points in time can present such outliers for sample times around the trigger event. This restriction is necessary to keep the response assumptions valid (discontinuities are removed).

4) Metamodelling: For each time sample, the vector of results $r(k \cdot T, d)$ and the matrix d are used to:

Build the metamodel: Regression extracts least square estimates for the coefficient set $c(k \cdot T)$.

Validate the metamodel: The metamodel must be characterized in terms of fitness on data, in order to be used for interpretation of effects and response prediction. For this, the residuals are analyzed according to the criteria defined in the Concepts. Optionally, some runs can be invested to optimize the coefficient set, by performing iterative regression and validation. Samples where the metamodel is unfit are removed from the analysis. When they are too many, the controller aborts the analysis and optionally redesigns the experiment for a higher metamodel accuracy. Important factors and interactions, and the way they vary in time, can be identified by the estimated coefficients.

Find the metamodel extremes: Optimization strategies vary depending on the type of effects. Convergence problems can occur when many factors present interaction and quadratic effects. In that case, the metamodel can be evaluated on a grid to reduce the search area. The output of this step is the pair of estimated factor sets for extreme values of the metamodel:

$$f_{min/max}(k \cdot T) = f(k \cdot T)_{R(f, k \cdot T)=min/max}$$

5) Simulate confirmation runs: Simulations can be run to confirm the predictions with respect to the response's extremes, for time samples of interest. Only the significantly different predictions are run, i.e. only those runs, for which factor sets are far enough from each other.

Results will show that the impact of factors and, consequently, the factor set which determines a response extreme value, vary in time.

6) Form bounds for the response: Finally, bounds for the transient response are "assembled". While metamodelling interpolates the response with respect to the factor set, the sampling in time and assembling of the response bounds performs a time interpolation of the response dependency on the factors.

The confirmation runs completed by the predictions are used for the samples $k \cdot T$. Between these samples, the response can be predicted by extending the dependency on the factors from the closest time sample. Still, the constant term is adjusted using the value $r(f = 0, t)$. Safer response margins can be obtained by corrections based on the residuals (as center value or maximum).

Finally, the controller must report the estimated metamodels as well as the predictions for extreme response values. Assembled response bounds and results of initial or confirmation simulation runs are also stored. Additionally, the resulting performance as described in what follows is reported.

C. Performance and accuracy considerations

The experiment time is

$$\tau = (m + conf) \cdot \tau_{sim} + s \cdot \tau_{regress\_validate} + s_{fit} \cdot \tau_{predict}$$

where s is the number of time samples, m is the number of runs of the DoE. $conf$ is the number of confirmation runs. s_{fit} is the number of time samples where the metamodel was fit. The dependency of m on the number of factors n, which is a measure of the response complexity, is visible in Figure 2. The plot corresponds to classical DoEs, which require minimum runs for the respective metamodel complexity [5]. The error rate of the experiment is given mainly by the results of residuals' evaluation. The number of significantly different sets of effects and extreme response predictions shows how complex the response dependency in time is.

D. Framework

Figure 3 show the basic experiment setup. The modelling and simulation are handled with SystemC/SystemC-AMS.The controller (in MATLAB) realizes the experiment planning, control of simulations and analysis of results. Extensive data handling is also required and each experiment must be properly reported by the controller.

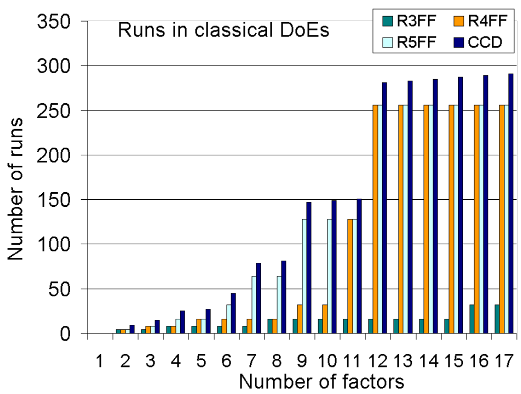

Fig. 2. Number of runs in classical DoEs

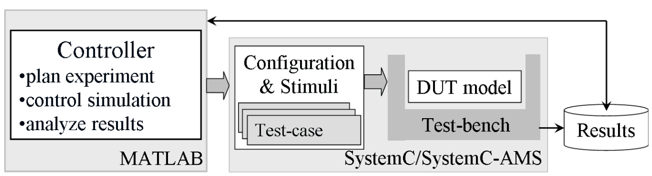

Fig. 3. Steps

IV. Experimental results

A. System, responses and factors

The system subject to experiment is an ECU for car window lift applications, with the structure visible in Figure 4. The system is heterogeneous (includes mechanics, analog electronics, digital electronics, software) and presents multi-nature parameters with variations.

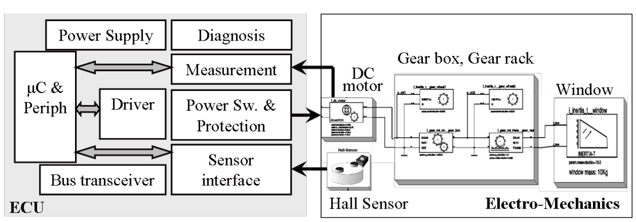

Fig. 4. ECU structure

The central element of the electro-mechanical subsystem, driven and supervised by the ECU is a DC motor. A Hall sensor provides the ECU with the speed and direction of the motor, as electric signals, for window position track. The main digital element is the microcontroller subsystem (MCU) compatible to the standard 8051 core. A LIN transceiver is the interface for commands to control the mechanical load. Two low-side switches (LSS) are dedicated to DC motor control, by relays. A measurement block includes an interface (MI) with 8-bit ADCs, and a diagnosis unit, which signals the MCU when error conditions occur. An amplifier feeds the MI with amplified value of the motor current.

B. Experiment

The simulated test case is visible in Figure 5. For each window move command, the LSSs are controlled to drive the DC motor. At time 1.8 seconds, an obstacle in the window determines increase in the force developed by the DC motor, thus in its current. The measurement block senses over-current and the MCU reacts by switching the LSSs off.

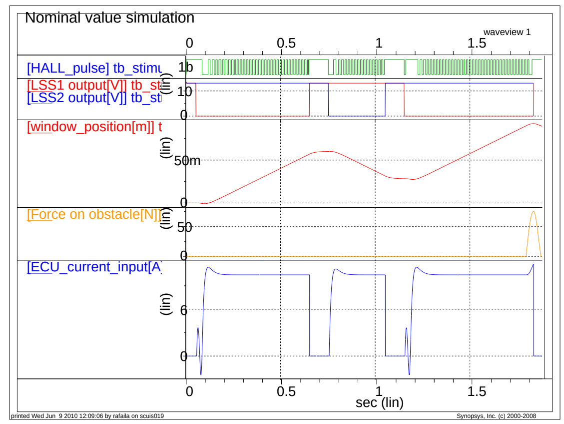

Fig. 5. Simulated testcase

To demonstrate the approach, 2 responses are analyzed: the current at the ECU input and the window position (r1=current[A]; r2=window position[m]). Sources of variability involved in this scenario introduce several factors in the experiment (Table I).

Their ranges are extracted from the spec. and transferred to a DUT configuration file, which is parsed for experiment and simulation setup. For a set of 10 factors, a CCD DoE is augmented by a LHS DoE for more metamodel accuracy and for later validation against separate runs. A total of 200 runs are initially performed.

TABLE I
FACTORS

Source of variation	Factors
Environment	Supply voltage
Measurement	Amplifier gain, Shunt resistance
Power switch	Slew rate, ON-resistance
Protection	Thresholds (e.g. current)
Transceiver	Delay
Actuator	DC motor params. Gear box, Gear rack params. Window mass

C. Metamodelling

The initial response traces are sampled in $s = 100$ equidistant points. Data for each of the two responses is fitted in each time sample and the metamodel is validated.

Figure 6 exemplifies a set of conforming residuals for r2, in an arbitrary time sample. The hypothesis of fit to a normal distribution of mean zero is passed. The correlation to the response and the maximum value are also tested. Samples where the metamodel is concluded unfit are removed from the analysis, and only the simulated points are used. This way, no extra effort is spent to predict the extremes for such samples.

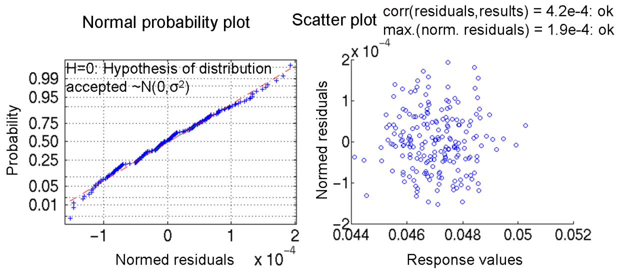

Fig. 6. Residual analysis

To study factor effects, coefficients of the metamodels are studied at sample times of interest. Figure 7 shows the metamodel of response r2, plotted for a subset of 4 factors, when the others are set to 0, for 3 time samples. It can be concluded that the impact of factors varies in time.

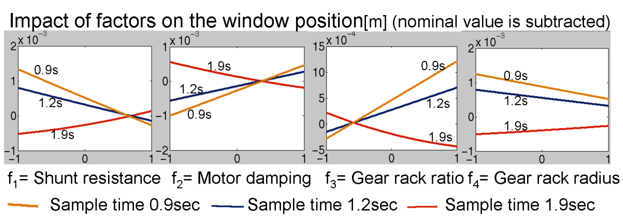

Fig. 7. Factor effects at different time samples

Different time samples correspond to different factor effects, and consequently, to different predictions for the factor sets which determine response extremes. Some are exemplified in Table II, for the window position (r2).

The factor sets which were found are not only corners, i.e. where factors are either maximum (1) or minimum (-1). This is justified by the presence of quadratic effects (Figure 7).

TABLE II
WORST-CASE PREDICTIONS FOR R2

			$f(1)$	$f(2)$	$f(3)$	$f(4)$
Time	Min.	2.57cm	1	1	1	1
0.38s	Max.	2.82cm	-1	-1	-1	-1
Time	Min.	4.27cm	-1	-1	-1	-1
0.90	Max.	5.28cm	1	1	1	1
Time	Min.	7.84cm	0.35	-1	1	-1
1.76s	Max.	9.52cm	-1	1	-1	0.84
Max. residual				0.05		

These steps are useful to identify and interpret effects which are otherwise hard to track, especially when they change in time. For instance, the effect of the LSS delay depends on the simulation time: the faster the switch is, the higher the position is at time 0.38 seconds. At time 0.9 seconds, because of the 2 switching events which occurred until then, the effect becomes positive, i.e. the switch determines an increase in the response. Therefore, the accumulation of the factor influences in time, combined with the interaction and quadratic natures of effects which can occur, becomes rather complex.

D. Analysis of bounds for the responses

To analyze the worst-case over the complete simulation duration, the bounds for the two responses are formed as explained in the Approach. They are plotted in Figure 8. The figure also plots the resulted bounds after a similar analysis, but in fewer time samples ($s = 10$), and without interpolation.

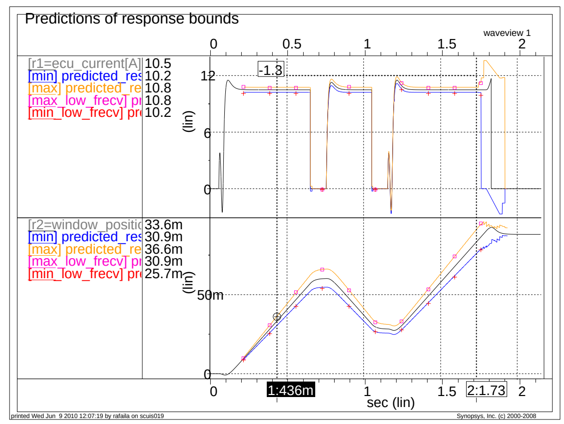

Fig. 8. Predictions on transient responses

The large response metamodel variance to the end of the test is caused by various switch-off times. When the obstacle occurs, factors which impact the DC motor speed, the window inertial effects and ECU block delays affect the reaction delay. Thus, they introduce discontinuities in the response sampled at these times. This explains the outliers and lack of metamodel fit around the obstacle detection time.

Confirmation simulations are run for different time samples. Only the significantly different worst-case predictions are run, i.e. only those runs, for which factor sets are far enough from each other. A minimum distance between factor sets of 1% is used to determine whether to run confirmations. Figure 9 shows results of runs with the worst-case factor sets predicted in different time samples, for the window position.

They confirm that the worst-cases originate from different factor sets at different time samples, because the traces are intersecting. This also happens for the current, as factor sets which determine higher overshoots in current correspond to smaller settle values.

E. Performance

The performance is evaluated on the criteria introduced in the Approach. The experiment included a 200-runs initial experiment ($m = 200$), at a simulation duration of $\tau_{sim.}$ ≈45 seconds per run (for 2 seconds of real time simulated in this testcase). None of them failed the functional test i.e. the response could be recorded for each.

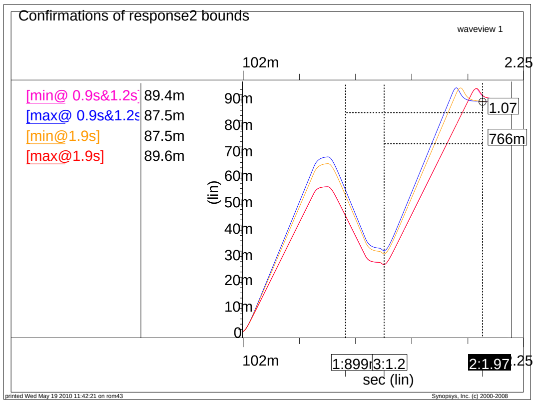

Fig. 9. Confirmation runs

Out of a total of $s = 100$ time samples, r1 results presented too many outliers in 5 time samples, while for 7 samples of r1 and for 8 of r2 a metamodel could not be fitted. These were caused mainly by the responses' discontinuities: different switch-off times to the end of the simulation. For response r2, sets of different factor effects determined $conf = 18$ different worst-case predictions, both for min. and max. These were run for confirmation, and output residuals smaller than 5%.

The postprocessing durations become significant only for a high number of samples: $\tau_{regress\_validate}$ ≈4 seconds per sample, while $\tau_{predict}$, of 8 seconds per sample, was invested only for the well fitted metamodels. When the metamodels are unfit for many samples, either a resampling or additional runs should be performed.

F. Discussion

A more complex system is likely to require a higher τ_{sim} (take longer time to simulate) and have more factors. Similar problems occur when the DUT model is more accurate: more details e.g. in the function or structure of a model adds factors and increases the simulation time. But a higher number of factors n determines a higher number of experiment runs m, in order to fit a reasonable metamodel (Figure 2).

$\tau_{regress\_validate}$ and $\tau_{predict}$ also increase with the number of factors to consider. With respect to the number of time samples, a higher s reflects into bigger postprocessing time.

Tracing should be done as little as possible, especially when the simulated duration is high, because it slows down simulation and will consume too much memory e.g. for more than 100 runs. Only the time samples of interest from the traced responses are loaded during postprocessing, because of the limitations in the run-time memory.

To avoid oversampling or undersampling the simulated traces before the metamodelling step, an adaptive sampling step dependent on the rate of change in time of the response would also be an option.

It is therefore important to monitor each step closely, in order to identify and control potential bottlenecks of simulation or processing. However, the trade-off between the overall time versus the responses' complexity which was successfully handled recommends such extended DoE flows for future use.

V. Conclusion

The presented work addresses the multivariate sensitivity analysis and search for the bounds of signals, necessary when multiple sources of variation influence the system behaviour. The DoE planning methods support in generating minimum-sized experiments, even for large factor sets, which present 2^{nd} order and interaction effects.

The approach starts with designing experiments to fit a satisfactory regression metamodel on simulation results. The metamodel bounds and corresponding factor sets can be estimated for each time point of interest and confirmed by simulation. Analyzing the effects and how they change in time achieves a better understanding of the worst-case behaviour.

This extended DoE and metamodelling flow is applied on a case-study of an automotive window lifter ECU. No output of the simulation experiment is wasted and the postprocessing effort is controlled and small enough. Results show that the statistical methods under evaluation form an efficient solution to cope with variations in complex automotive systems.

References

[1] M. Ayeb, H. Theuerkauf, and C. Winsel. Robust identification of nonlinear dynamic systems using design of experiment. In *IEEE Computer Aided Control System Design*, pages 2321–2326, 2006.

[2] N. Femia and G. Spagnuolo. True worst-case circuit tolerance analysis using genetic algorithms and affine arithmetic. In *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 2000.

[3] C. Grimm, W. Heupke, and K. Waldschmidt. Refinement of mixed-signal systems with affine arithmetic. In *DATE '04: Proceedings of the conference on Design, automation and test in Europe*, page 10372, Washington, DC, USA, 2004. IEEE Computer Society.

[4] MathWorks. *MATLAB Statistics toolbox online documentation, regression statistics documentation*.

[5] D. Montgomery. *Design and analysis of experiments*. John Wiley & Sons, 2005.

[6] V. Nookala, C. Ying, D. Lilja, and S. Sapatnekar. Microarchitecture-aware floorplanning using a statistical doe approach. In *Design Automation Conference*, pages 579–584, 2005.

[7] M. Rafaila, C. Decker, C. Grimm, and G. Pelz. Design of experiments for effective pre-silicon verification of automotive electronics. In *Advances in Design Methods from Modeling Languages for Embedded Systems and SoC's - Selected Contributions from FDL'09*. Springer, 2009.

[8] M. Rafaila, C. Decker, C. Grimm, and G. Pelz. Sequential design of experiments for effective model-based validation of electronic control units. In *Mikroelektroniktagung ME10*, 2010.

[9] T. W. Simpson, J. D. Peplinski, P. N. Koch, and J. K. Allen. Metamodels for computer-based engineering design: Survey and recommendations. *Engineering with Computers*, 17:129–150, 2001.

[10] A. Singhee and R. Rutenbar. Statistical blockade: a novel method for very fast monte carlo simulation of rare circuit events and its application. In *Design, Automation and Test in Europe*, pages 1–6, 2007.

[11] R. Stone and J. Ball. *Automotive Engineering Fundamentals*. SAE International, 2004.

[12] J. Tate, B. Woolford-Lim, I. Bate, and Y. Xin. Comparing design of experiments and evolutionary approaches to multi-objective optimisation of sensornet protocols. In *Congress on Evolutionary Computation*, pages 1137–1144, 2009.

[13] L. Trocine and L. Malone. An overview of newer, advanced screening methods for the initial phase in an experimental design. In *Winter Simulation Conference*, pages 169–178, 2001.