Efficient calibration of transient ECU functions through system optimization

André Sell, Frank Gutmann, Tobias Gutmann

Abstract

The necessity of the recurring calibration of engine control functions for a large number of variants opens up potential for increasing efficiency. In this context, transient functions are moving more into focus due to legislative changes in the form of more dynamic certification cycles and the consideration of real driving. Since the well-known stationary DoE and MBC methods cannot be used directly for transient calibration tasks, methods have already been proposed for automated and efficient processing of such tasks. In this paper, some potentials of these methods are pointed out and suggestions for further development are made. These concern the amount of data to process and parameters to be optimized, the determinacy of the optimization problem and the plausibility of the optimized parameters.

The SGE *system optimization* method applies these suggestions and addresses the potential by automatically calibrating ECU functions based on measurement or modeled data by simultaneously optimizing all calibration parameters. In this way, also transient systems can be calibrated efficiently, as the example of an exhaust gas temperature function shows.

Kurzfassung

Die Notwendigkeit der wiederkehrenden Applikation von Motorsteuerungsfunktionen für eine Vielzahl von Varianten eröffnet Potenzial zur Effizienzsteigerung. In diesem Zusammenhang rücken transiente Funktionen aufgrund gesetzlicher Änderungen in Form von dynamischeren Zertifizierungszyklen und der Berücksichtigung des realen Fahrens stärker in den Fokus. Da die bekannten stationären DoE- und MBC-Methoden nicht direkt für transiente Applikationsaufgaben eingesetzt werden können, wurden bereits Methoden zur automatisierten und effizienten Bearbeitung solcher Aufgaben vorgeschlagen. In diesem Beitrag werden einige Potenziale dieser Methoden aufgezeigt und Vorschläge zur Weiterentwicklung gemacht. Diese betreffen die Menge der zu verarbeitenden Daten und zu optimierenden Parameter, die Bestimmtheit des Optimierungsproblems und die Plausibilität der optimierten Parameter.

Der SGE Ansatz der *Systemoptimierung* wendet diese Vorschläge an und adressiert damit das vorhandene Potenzial, indem sie Steuergerätefunktionen basierend auf Mess- oder modellierten Daten automatisch durch gleichzeitige Optimierung aller Parameter appliziert. Auf diese Weise können auch transiente Systeme effizient bearbeitet werden, wie das Beispiel einer Abgastemperaturmodells zeigt.

1 Introduction

1.1 Motivation

The calibration effort of parameters in present vehicle ECUs is growing due to the increasing diversification of drive, vehicle and country variants. In addition, changes in legislation in the form of more dynamic certification cycles and the consideration of real driving bring the transient behavior of the vehicle, especially with regard to emissions and diagnostic functions, more into focus. This further increases the calibration effort, since the known stationary DoE and MBC methods cannot be applied [1] and manual calibration is very time consuming [2]. Therefore, methods are required to make the calibration of transient functions more efficient and accurate in the daily routine of a calibration engineer.

1.2 State of the Art

Proposals exist for the automated calibration of transient functions [2]. The SGE approach that does also apply to transient functions is the so-called *system optimization.* It permits the automated calibration of ECU functions by simultaneously optimizing all calibration parameters. The aim is to minimize the deviation of the ECU system behavior from a reference behavior [3]. Problems and limitations arising in this context are discussed in the following and potentials are pointed out.

2 Challenges and Concepts

2.1 Large amounts of data to be processed

Transient control functions contain elements whose behavior is not only determined by the current state of the system inputs, but also depends on the previous state - such as filters and integrators. Therefore, it is not possible to completely calibrate such systems by setting combinations of the inputs e.g. according to a DoE plan and measuring the outputs stationary. If the dynamic behavior of a control function is to be calibrated, time traces of the input and output signals of the system must be applied. This makes it necessary to process large amounts of data during optimization - regardless of whether these originate from measurements, simulations or a dynamic model. This places increased demands on the plausibility of the data and the performance of the optimization [2]. One proposed solution is to reduce the time data to scalar describable properties (KPI values = "Key Performance Index" values) [1]. However, since these are limited to fixed maneuvers, they must be selected very carefully in order to cover the relevant operating range. In addition, the optimized calibration must be tested and confirmed for an extended operating range.

The time-based data of the input and output signals required for optimization can be reduced by converting them from their acquisition or calculation grid to a dynamic grid derived from the course of the signals without exceeding a defined deviation. As a result, fewer points are placed in stationary phases than in dynamic phases. The ECU function to be optimized may have to be adapted if it was previously calculated in a constant grid, for example. Since the grid generation has to be done only once but saves runtime in each calculation run of the optimization, a clear overall advantage of optimization time results.

The following figure shows an example of the reduction of a signal by more than 80% of the points without the deviation from the output signal exceeding 1%.

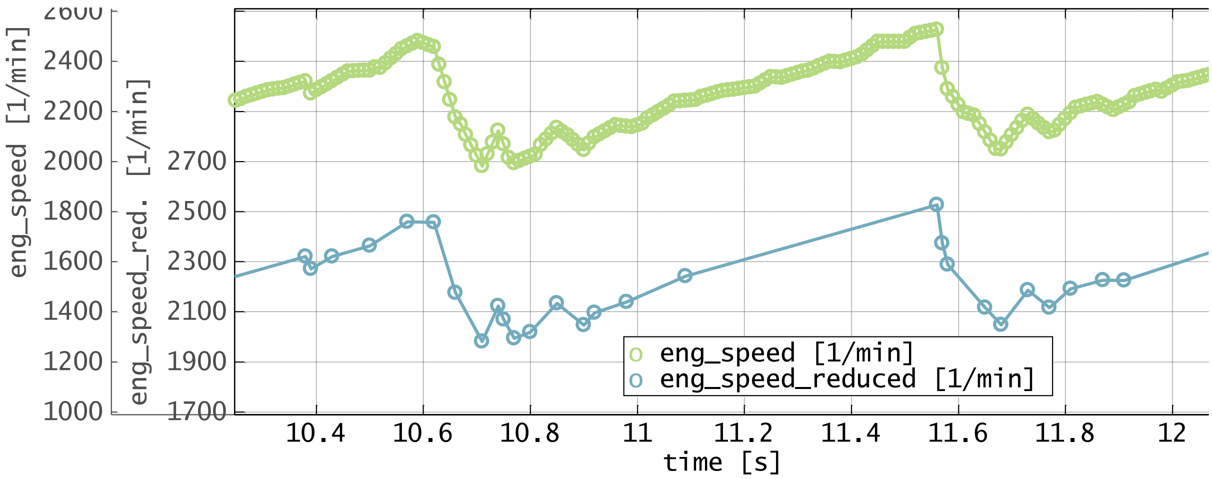


Figure 1: 80% signal reduction with less than 1% deviation

If further runtime optimizing measures are added, such as the integration of the ECU function as a Simulink Host-Based Shared Library, typical calibration tasks such as the dynamic exhaust gas temperature model can be optimized overnight, which is usually sufficient for the calibration engineer as a typical user.

2.2 Large number of parameters to be optimized

Although methods exist to directly map models in the ECU [4], most of the functions to be calibrated are traditionally implemented as a combination of maps, curves and scalar parameters. Since the *system optimization* optimizes all parameters of a function simultaneously and curves and maps consist of several to many individual parameters, the optimization must handle hundreds to thousands individual parameters. This results in two problems. On the one hand, the calculation time required for the optimization increases and on the other hand, flawed input data cause the optimization to use the high number of parameters to minimize the objective function by generating implausible and wavy maps [2]. The latter problem is discussed in the section 2.4.

To compensate the long calculation runtimes that occur, it is proposed to replace the parameters with constructs that can be described with fewer parameters, such as polynomials [2] [5], approximation by individual cells [6] or LoLiMoT networks [7]. All of them have in common that restrictions of the form capabilities of the ECU parameters are made and thus the behavior of the ECU function cannot be mapped exactly if interpolation grid, incrementation or the interpolation routine deviate. When using polynomial models, another disadvantage is that the expected optimized parameter shape must be known in order to determine the polynomial order. Thus the utilization is more effortful when new or changed functions are to be calibrated because usually several iterative optimizations are necessary before suitable settings were found.

Therefore, an algorithm is proposed which allows runtime advantages despite exact mapping of the ECU behavior and without prior knowledge. In this procedure called *initial estimation*, similar to the approximation by single cells [6], a map/curve is first divided into a smaller number of single surfaces, so that the number of parameters to be optimized is reduced. This division is not static, but adapts itself in the course of the optimization. As the optimization progresses, the cells are divided repeatedly until, at the end of the optimization, the map again corresponds to the ECU state and thus the optimization result is not subject to any restrictions. As an example, the following figure illustrates the initial reduction of a map from 144 to 16 individual parameters.

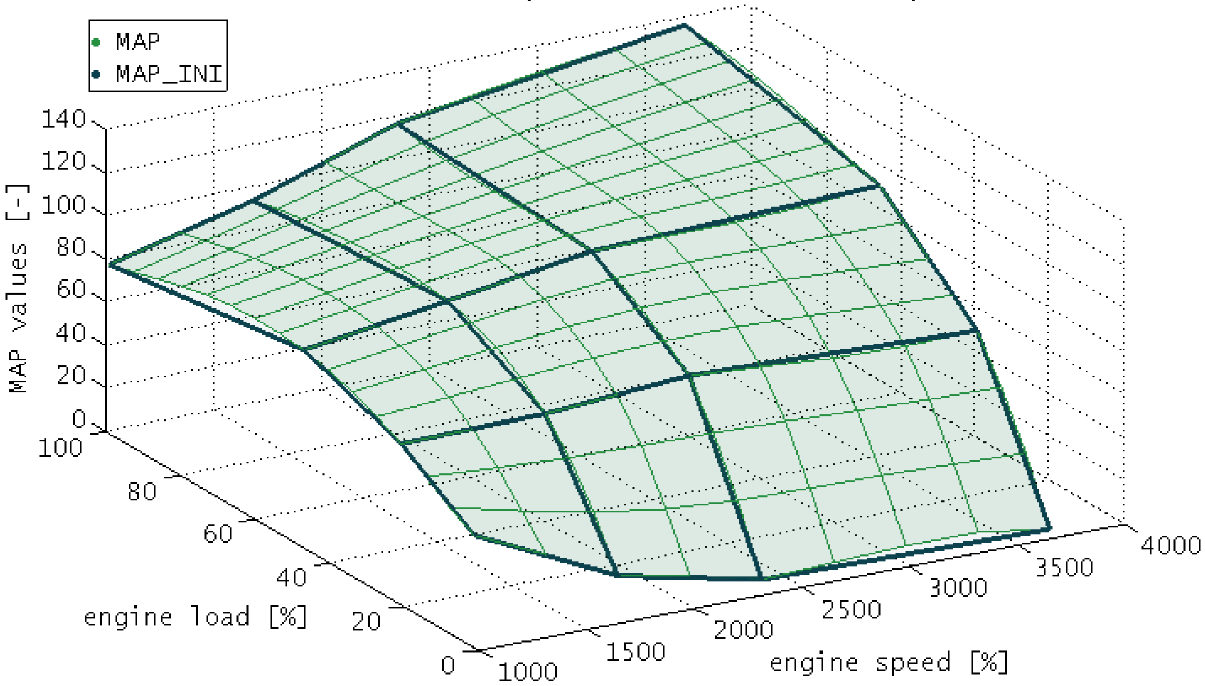


Figure 2: Initial estimation map reduction

The following figure shows the runtime reduction that can be achieved for the function of a gasoline engine load detection with five maps. The course of the objective criterion of two optimizations can be seen, which differ only in the use of the *initial estimation*. For *initial estimation*, the 768 individual parameters were reduced to 80 (-90%). In this way, the optimization result is completed in 20min instead of 120min runtime (-83%).

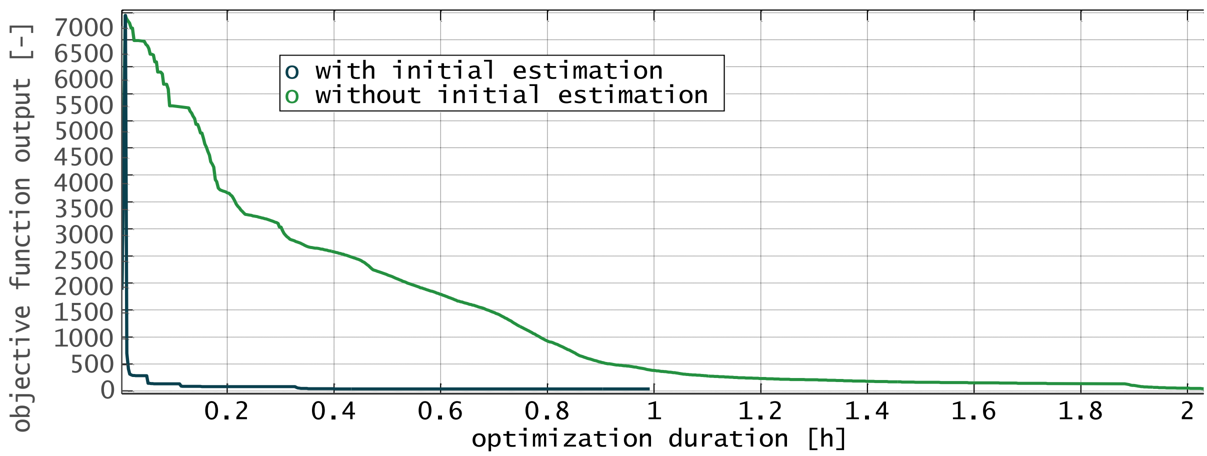


Figure 3: Optimization performance due to initial estimation feature

2.3 Underdetermined optimization task

The optimization task can be underdetermined if, for example, individual parameters exist in the case of maps for which there are no input data points in any of the four neighboring map sections. Then this map parameter will not have any influence to the objective criterion and therefore cannot be determined in a unique way during optimization. There are suggestions to consider a penalty term depending on the smoothness in this case [8], which will be dealt with in section 2.4.

There is also a common case of underdeterminedness when parameters of a function are summed or multiplied and there is an infinite number of value combinations that produce the same result. As described in [2], this case can be made unique by constraints of the optimization. In the following example of determining the optimal ignition angle of a torque model as the sum of two parameters, the lambda correction in CURVE_ZWOPTLAM for lambda 1, for example, can be set to zero.

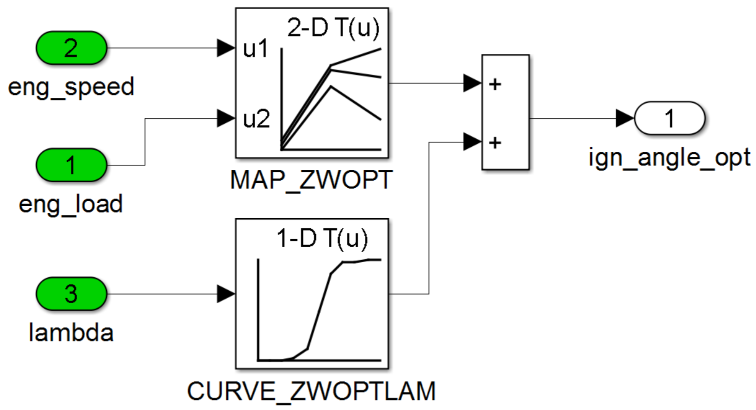


Figure 4: Underdetermined optimization task

2.4 Smoothness / plausibility of the optimized parameters

An important criterion for the evaluation of the optimization, besides the accuracy, is the plausibility and smoothness of the resulting parameters. Due to an underdetermined optimization task or flawed measurement data, the results of the optimization can be unsatisfactory and considerable postprocessing is required [2]. Two approaches exist here. On the one hand the mentioned reduction of the parameters to e.g. polynomial models leads to restrictions of the degrees of freedom, so that the results are necessarily smooth - however with the discussed disadvantages. On the other hand, smoothness in the form of constraints or penalty terms [8] can be considered during optimization.

Constraints limit the permissible gradients and curvatures. Within these limits, however, they do not influence the optimization, so that the result will not be smoother than these limits. In other words, it is very difficult for the user to define these limits in such a way that they provide plausible smooth results without worsening the accuracy to an undesirable extent. Similarly is behaves with the quantitative adjustment of penalty terms to take smoothness into account. These always deteriorate the result of the optimization and also require careful tuning.

Both smoothness criteria as a constraint and a penalty term must be individually adapted to each parameter to be optimized if the shapes of the parameters significantly differ. They also strongly depend on the quality of the data for the input and output signals. Flawed data requires stronger smoothness criteria than error-free data. Experience has shown that both methods require an iterative adjustment over several optimizations, which eliminates part of the efficiency gain through optimization.

An additional problem of penalty terms is that they are also applied to parameter sections for which no data is available. Since in these sections the parameter values have no influence on the optimization result, these sections will usually be wavy and implausible after optimization [2]. If one then applies a penalty term that evaluates the smoothness of the entire parameter as a whole, the smoothing of the underdetermined sections will unnecessarily worsen the optimization result, since the determined sections will also be smoothed at cost of the objective.

As already mentioned in [8], it is proposed not to consider a smoothness criterion as a penalty term but to use it as a further criterion for optimization. This then becomes a multi-criteria optimization and avoids the problem of worsening of the result due to the smoothing of underdetermined sections. However, there is still a need to adjust the weighting of the smoothness criteria of the individual parameters.

As part of our *system optimization*, we have developed a procedure for taking smoothness into account that does not require manual adjustment. The optimization algorithm does not consider the smoothness directly. Instead parallel to the optimization, a smoothing algorithm operates, which considers all parameters simultaneously, analogous to the optimization, and minimizes the gradient and curvature of the parameters using a criterion similar to [8]. It is limited by a maximum permissible worsening of the objective function caused by the smoothing. Optimization and smoothing algorithms are regularly exchanging data and integrate the respective progress. What's new is that the smoothness is not used directly as a penalty or constraint, but only the worsening of the objective function caused by the smoothing is taken into account.

There are some advantages to this approach. On the one hand, no manual parameterization of a smoothing criterion is necessary. Each parameter is smoothed individually up to the permitted threshold of the objective function. Thus, sections where only little error occurs due to smoothing (e.g. overdetermined areas due to multiple data) are strongly smoothed, while other sections are only slightly adjusted if much error would occur due to smoothing. Underdetermined sections are smoothed even completely. Furthermore the smoothing can compensate roughness between the parameters by the simultaneous processing of all parameters. This is very relevant when optimizing underdetermined functions, which contain multiplication or summation (see section 2.3). In such sections of the parameters that are not defined by limits, large smoothing advances without loss of quality are usually possible, since any number of combinations of a multiplication and sum provide the same result as described above.

Since the smoothing algorithm evaluates the objective function, this procedure results in an additional runtime compared to pure optimization without considering smoothness. However, in our experience, this procedure supports a plausible progress of the optimization and avoids local optima. In addition, the increase in runtime is within a range that allows overnight processing for typical calibration tasks, which is usually sufficient for a calibration engineer as a typical user.

As part of the *system optimization*, the user is provided with a comprehensive graphical user interface for postprocessing after completion of the described combined optimization and smoothing. There it is possible to perform manual or automatic postprocessing. The time related output of the objective function is available for comparison of all settings at any time. The smoothing algorithm already used during optimization is also available. In postprocessing, the constraint error threshold is adjustable which enables the user to conveniently weight accuracy and smoothness. This is a great advantage especially for flawed data and is much easier to handle than making fixed adjustments at the pre-processing of the optimization.

3 Application example

3.1 Calibration of a Transient Exhaust Temperature Function

The use of the *system optimization* with the features described before is explained now using a typical ECU function of an exhaust gas temperature model, which is somewhat outdated. Although there are more modern functions for mapping the physical behavior, this is a good example of a typical work package of a calibration engineer who has to work on existing functions that cannot completely map the physical behavior.

The function is illustrated in the following figures. It maps the gas temperature before the catalyst and the material temperature of the catalyst depending on 9 input signals. In a first step the stationary exhaust gas temperature before catalyst is calculated. Afterward the transient behavior before catalyst is applied and finally the exothermic and transient behavior of the catalyst is modeled.

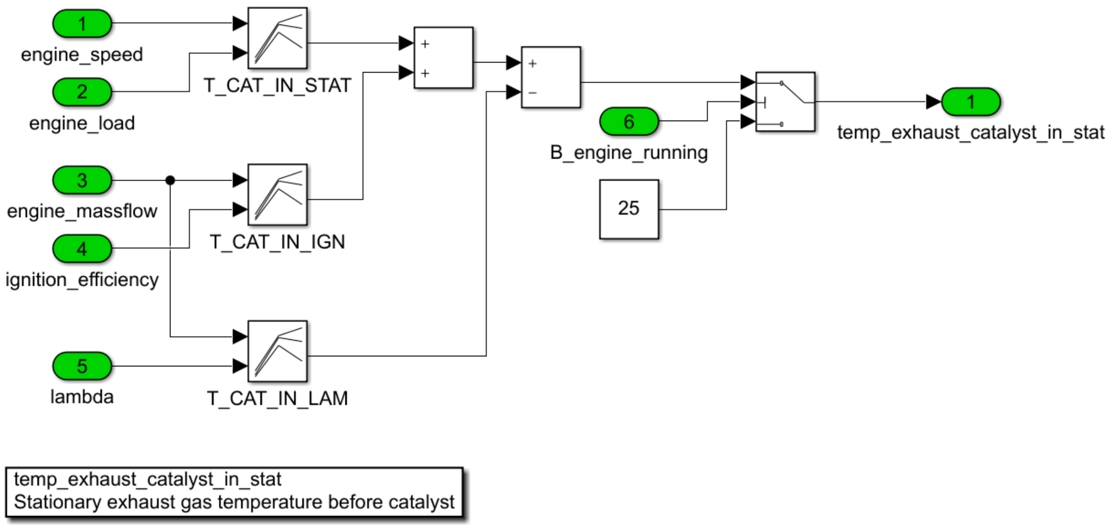


Figure 5: Stationary part of the ECU function calibrate

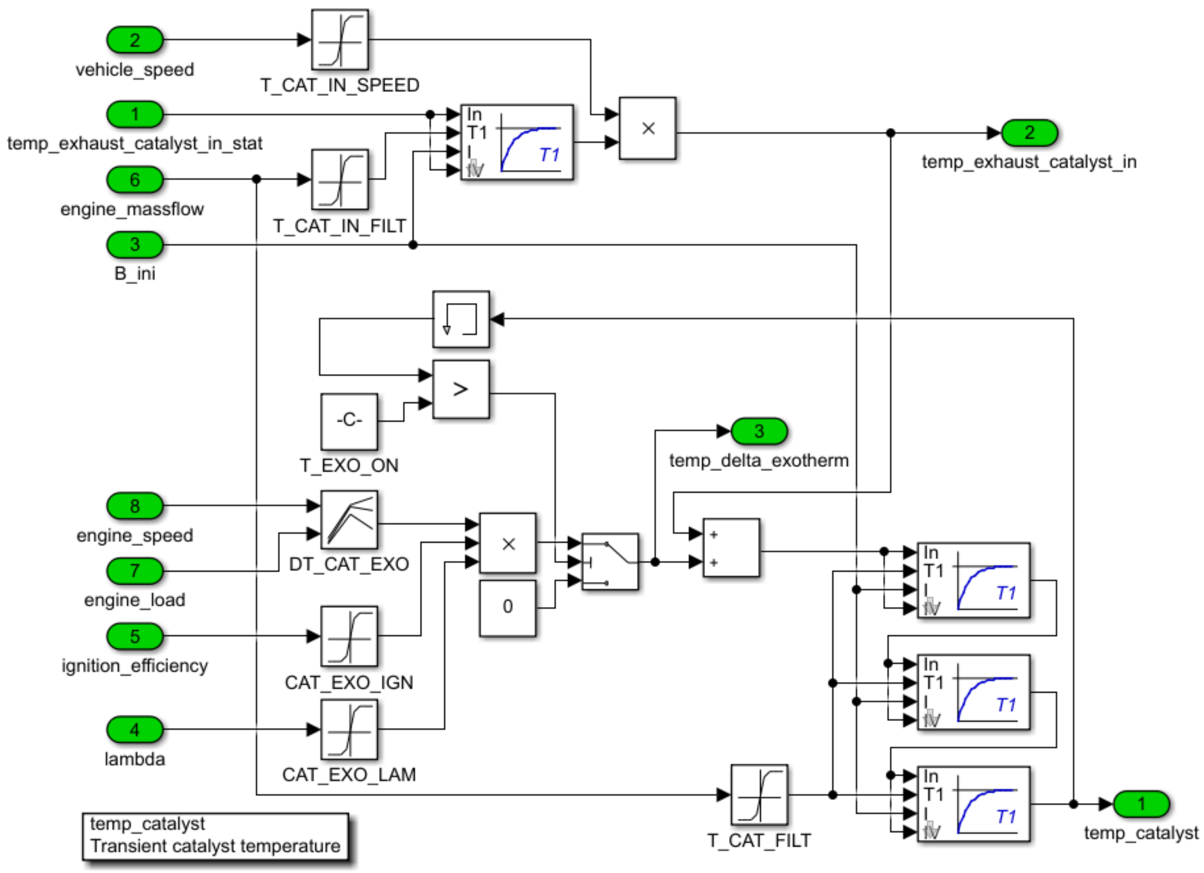


Figure 6: Transient part of the ECU function calibrate

Time-based measurement data containing all input signals and the two measured temperatures - a total of 30000 data points - are available as a reference. The data is derived from chassis dynamometer measurements and thus allows the calibration of parameters that are depending on the input signals being varied during data recording. These are 6 maps, curves and scalars consisting of 142 single parameters. Further parameters describing the ignition angle and lambda dependence were not varied in the measurements and were therefore taken from an existing dataset, which was determined in advance on the engine test bench.

The objective criterion was implemented by integrating a Simulink Host-Based Shared Library of the ECU function and calculating the deviation from the reference data for both temperatures. Since a certain degree of temperature deviation is less relevant, a final deviation weighting has been introduced to place more emphasis on minimizing high deviations.

Here the user has all options to apply his experiences and priorities in order to guide the optimization to his desired direction. This allows to define a compromise if an ECU function cannot exactly map the physical behavior.

The resulting signal is converted to a scalar quality criterion by the optimization. The calibration parameters are adjusted in such a way that this criterion is minimized.

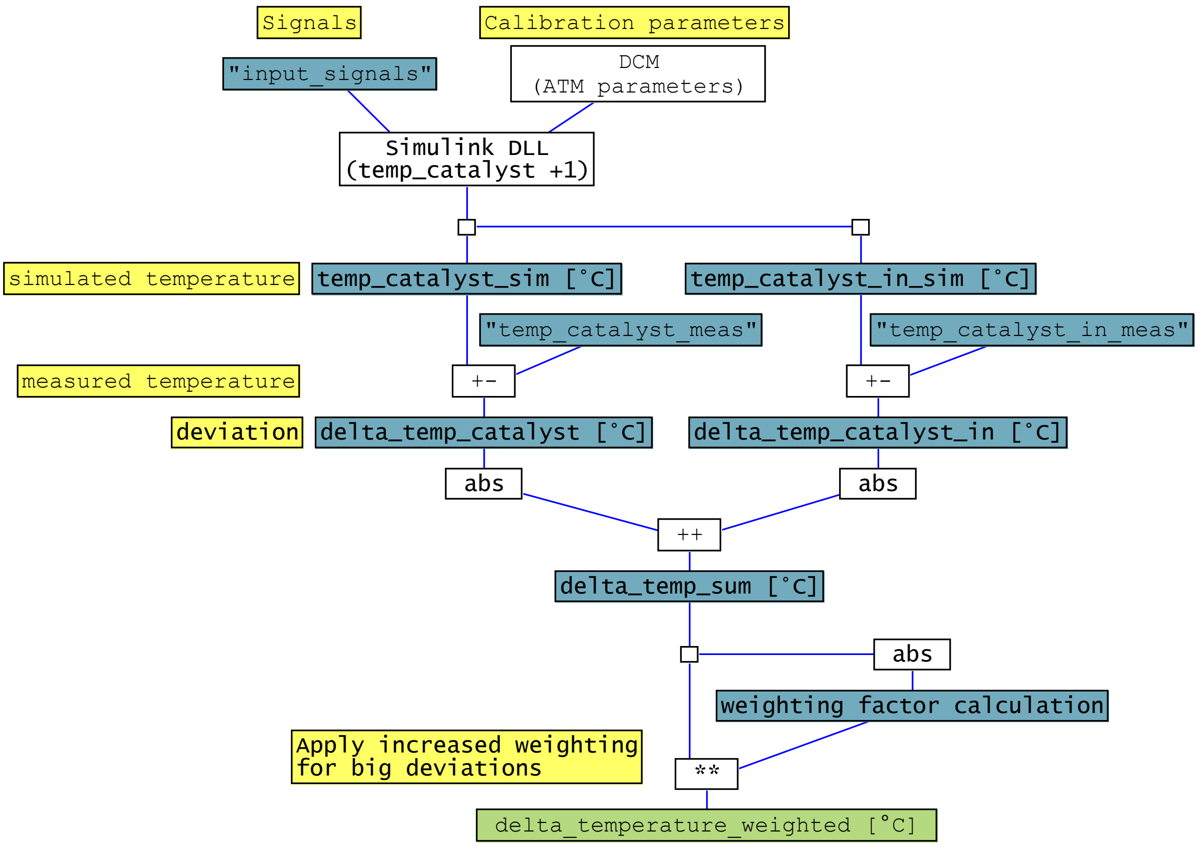
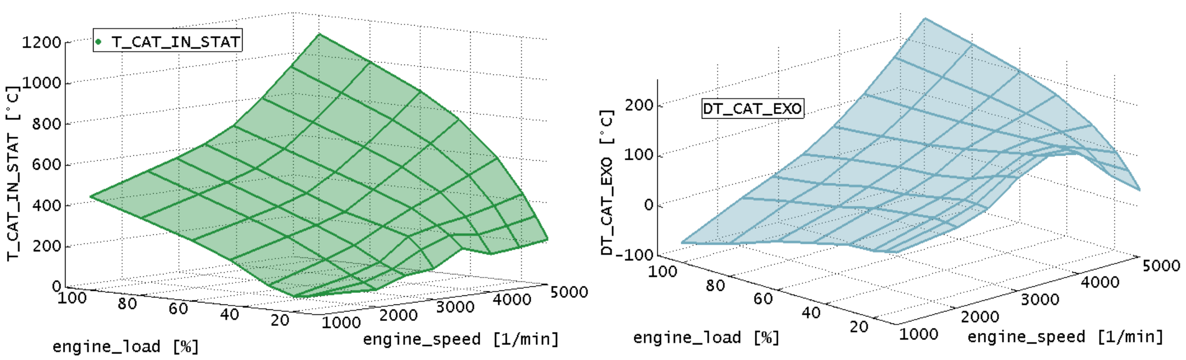


Figure 7: Objective function

For this function the optimization needs 2 hours of computing time on a standard computer.

The postprocessing takes about 15 minutes. As described, the user can view the results of the optimization in postprocessing and, if necessary, adjust and further smooth them. Graphically guided, a continuous selection of a state between the raw result of the optimization and a very smooth variant with a significant increase of constraint error is available as well as manual editing and data exchange with calibration data files. This enables a comparison with existing calibrations. Afterwards the parameters can be transferred directly into the ECU. See the following figures for the optimized calibration parameters of the exhaust gas temperature function after postprocessing.



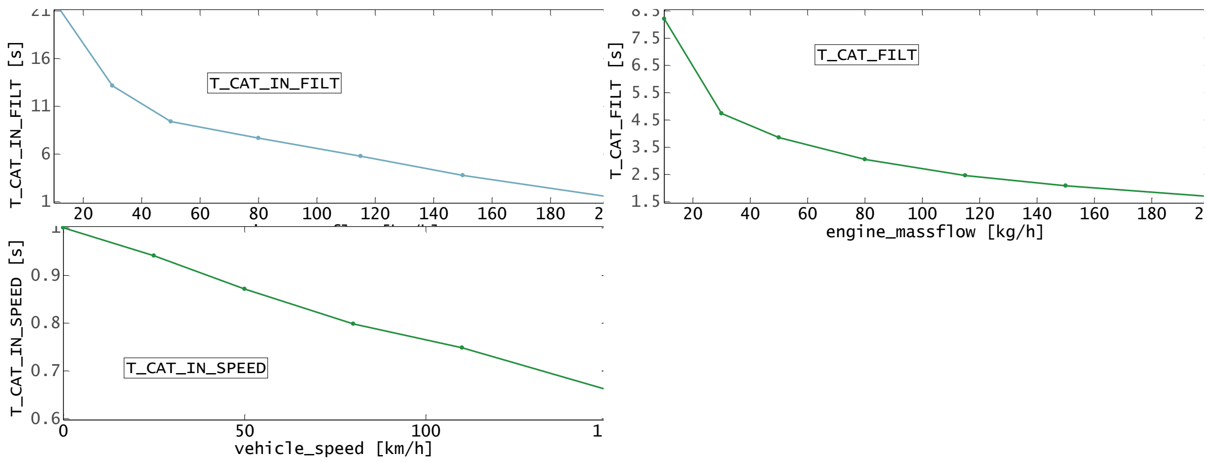


Figure 8: Calibration parameters after postprocessing

During postprocessing, the time-related objective function output as well as input and interim signals are always available for a comparison of all variants in a so called *system signal view*. In this way, the effects of postprocessing and smoothing can be evaluated in relation to the time data and thus the actual deviations of the properties to be optimized.

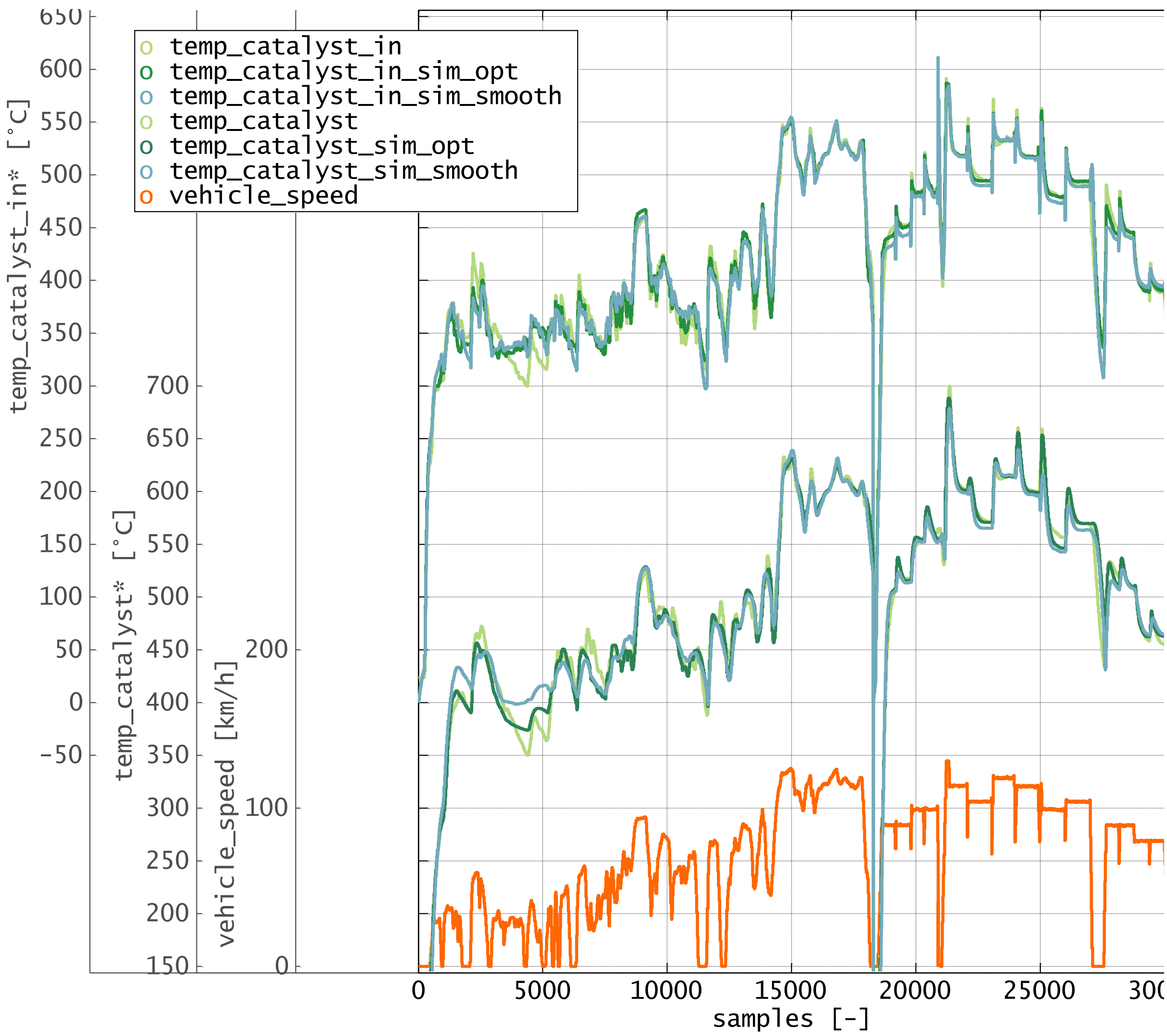


Figure 9: System Signal View – measured signals, optimized and smoothed result

As you can see from the previous illustration, this simple ECU function provides a good representation of the physical behavior with plausible shapes of the parameters to be optimized. The optimized signals (*_opt) map the measured ones well and there is only little deterioration by the postprocessing (*_smooth). Only at the beginning of the measurement during warm-up there are relevant deviations, since the function does not consider the temperature as a input variable.

4 Summary

In this paper some potentials of the already known methods for automated calibration of transient ECU functions were pointed out and suggestions for further development are made.

Transient control functions must be calibrated based on time related data of the input and output signals. This makes it necessary to process large amounts of data, which reduces the performance of the optimization. To avoid some limitations of the known „Key Performance Index" approach [1], it is proposed to reduce the time-based data to a dynamic grid derived from the course of the signals without exceeding a defined deviation. The reduced amount of data results in an advantage of optimization time without significant deviation of the optimization result.

Traditional approaches implement ECU functions as a combination of maps and curves consisting of several hundred to thousand individual parameters. This results in a significant performance loss of the optimization and in case of flawed data also to implausible and wavy parameter shapes. Derived from existing approaches to reduce maps and curves to constructs that can be described with fewer individual parameters, a new dynamic reduction mechanism called *initial estimation* is proposed to avoid the drawbacks resulting from the necessity of prior knowledge and a modified implementation of the ECU function.

To ensure plausible and smooth shapes of the parameters to optimize even for underdetermined optimization tasks and flawed measurement data, approaches already exist to use smoothness criteria in the form of optimization constraints or penalty terms in the objective function. This results in some disadvantages for the optimization and usually requires an iterative adjustment, which eliminates part of the efficiency gain through optimization. Therefore, a new proposal was made for considering smoothness without the need for manual adjustment. In parallel to the optimization, a smoothing algorithm is operating that does not influence the optimization through a penalty or constraint, but only regards the worsening of the objective function caused by the smoothing. In this way, an individual smoothing of the parameters is made possible based on the objective and no assumptions about the shape of the parameters are required. The capabilities of smoothing are further enhanced by the options available during postprocessing.

Finally, the suggestions were applied to the automated calibration of an exhaust gas temperature function resulting in a good representation of the physical behavior with plausible shapes of the parameters to be optimized, which is supported by extensive and comfortable features during postprocessing for influencing and evaluating the result.

5 Reference

[1] Stefan Scheidel, Marie-Sophie Gande – AVL List GmbH, "DOE-based transient maneuver optimization," in *International Symposium On Development Methdology*, Wiesbaden, 2017.

[2] Daniel Rimmelspacher, Dr. Wolf Baumann, Ferhat Akbaht, Dr. Karsten Röpke – IAV GmbH; Prof. Dr. Clemens Gühmann – Technische Universität Berlin, "Usability of computer-aided ECU calibration," in *International Symposium On Development Methdology*, Wiesbaden, 2017.

[3] A. Sell, F. Gutmann, T. Gutmann - SGE Ingenieur GmbH, System optimization for automated calibration of ECU function; Automotive Data Analytics, Methods, DoE, Renningen: expert Verlag, 2016.

[4] M. G. Stefan Angermaier, Implementation of data-based models using dedicated machine learning hardware (AMU) and its impact on function development and the calibration processes; Automotive Data Analytics, Methods, DoE, Renningen: expert Verlag, 2017.

[5] I. Brahma, The Challenges of an Emirical Model Based Transient Calibration Process; Design of Experiments (DoE) in Engine Development, Renningen: Expert Verlag, 2011.

[6] D. S. H. L. M. C. S. M. Grégory Font, Derivative Free Optimization Method and Physical Simulations Coupled with Statistical Models for Transient Engine Calibration; Design of Experiments (DoE) in Engine Development, Renningen: expert Verlag, 2011.

[7] O. Nelles, "Lokale, lineare Modelle zur Identifikation nichtlinearer, dynamischer Systeme," in *at - Automatisierungstechnik, Band 45, Heft 4*, ISSN (Online) 2196-677X, ISSN (Print) 0178-2312, DOI: https://doi.org/10.1524/auto.1997.45.4.163. , 1997, pp. 163-174.

[8] R. L. Y. D. F. K. T. K. Jan-Christoph Goos, Computing Optimized Calibration Maps including Model Prediction and Map Smoothness; Design of Experiments (DoE) in Engine Development, Renningen: expert Verlag, 2015.

The Authors:

Dipl.-Ing. André Sell
SGE Ingenieur GmbH
Gräfelfing

M.Eng. Frank Gutmann
SGE Ingenieur GmbH
Gräfelfing

Dipl.-Ing. (FH) Tobias Gutmann
SGE Ingenieur GmbH
Gräfelfing