Dossier

This paper is a part of the hereunder thematic dossier published in OGST Journal, Vol. 70, No. 1, pp. 3-211 and available online here

Cet article fait partie du dossier thématique ci-dessous publié dans la revue OGST, Vol. 70, n°1, pp. 3-211 et téléchargeable ici

DOSSIER Edited by/Sous la direction de : **B. Leduc et P. Tona**

IFP Energies nouvelles International Conference / Les Rencontres Scientifiques d'IFP Energies nouvelles
E-COSM'12 — IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling
E-COSM'12 — Séminaire de l'IFAC sur le contrôle, la simulation et la modélisation des moteurs et groupes moto-propulseurs

Oil & Gas Science and Technology – Rev. IFP Energies nouvelles, Vol. 70 (2015), No. 1, pp. 3-211

Oil & Gas Science and Technology – Rev. IFP Energies nouvelles, Vol. 70 (2015), No. 1, pp. 31-39

DOI: 10.2516/ogst/2013202

OPEN ACCESS

IFP Energies nouvelles International Conference
Rencontres Scientifiques d'IFP Energies nouvelles

E-COSM'12 - IFAC Workshop on Engine and Powertrain Control, Simulation and Modeling
E-COSM'12 - Séminaire de l'IFAC sur le contrôle, la simulation et la modélisation des moteurs et groupes moto-propulseurs

The Art of Control Engineering: Science Meets Industrial Reality

Urs Christen* and Rainer Busch

Research and Advanced Engineering, Ford of Europe, Aachen – Germany
e-mail: uchriste@ford.com - rbusch1@ford.com

* Corresponding author

Abstract — *This paper is intended to stimulate a discussion between academia and industry on automotive control. It lists requirements that usually are not covered in control classes but are important for the way control is practiced in the automotive industry, among them are calibratability, extendibility, or testability of controllers. Then, three examples are discussed in more detail, one being energy management for hybrid electric vehicles. In the discussion, we suggest some research topics.*

Résumé — L'art du génie automatique : science en rencontre avec la réalité industrielle — Cet article a l'intention de stimuler une discussion entre le monde universitaire et l'industrie sur le contrôle appliqué à l'automobile. Il énumère des exigences de bases, traditionnellement absentes des programmes de génie automatique et pourtant essentielles à la pratique de cette discipline dans l'industrie automobile. Parmi celles-ci on trouve le calibrage, l'extensibilité, ainsi que la testabilité des systèmes asservis. Trois exemples sont discutés plus en détail, dont entre autres la gestion d'énergie pour les véhicules hybrides. Dans les conclusions, nous proposons quelques sujets de recherche.

INTRODUCTION

When people who have just finished their academic degree start their technical career in industry, they are eager to apply the latest controller design methods they have learned. Fifteen years ago, when the first author joined *Ford* Research in Germany, the latest hype was H_∞ control. As part of the job interview, he had given a presentation about H_∞ controller synthesis, and he expected to apply it in his daily work.

Sure enough, he never did. Except for one single occasion when he wanted to explore the limits that could be achieved for a control loop, he never even came close to applying any H_∞ control in a project. Why is this?

It is a manifestation of the gap between academia and industry that has been around for a long time (Zhu, 1996). This gap in the application and the notion of what constitutes the state of the art in control is slightly different for each industry – see for instance Klatta and Marquardt (2009) for the process industry. In this paper, we provide a personal view with a focus on the automotive industry. An earlier survey paper treating powertrain control is Cook *et al.* (2006); vehicle dynamics applications are treated in Hrovat *et al.* (2011), for instance.

Control in the Automotive Industry

During the past decades, the automotive industry has developed a particular way of working with control. This way of working does not necessarily follow any guidelines established by academia, and it does not change with every hype in the control community.

In Europe, suppliers traditionally sell Electronic Control Units (ECU) together with the software that implements low level drivers and high level control algorithms to the car manufacturers. The OEM (Original Equipment Manufacturers) then calibrate the software, *i.e.*, they determine the values for the parameters in the software and thus define the final behavior. This allows them to choose some trade offs differently from others and hence to distinguish themselves from competitors.

More recently, OEM increasingly add some of their own software implementing high level control algorithms to the base software bought from suppliers. This is done because the way how a feature is controlled more and more serves as a distinguishing factor for OEM. Nevertheless, the basic approach of first developing the control structure and later calibrating it is not changed; different people with different expertise usually are involved in these activities. The control structure is set up by people who are experts in control and software development; calibrators are experts for finding trade offs between different attributes and features.

Organization of the Paper

The rest of the paper is organized as follows. In the next section, we provide some more background information such that we can formulate some requirements on control systems and their development, as derived from the way our industry works, in Section 2. Section 3 illustrates them with some concrete examples. The paper concludes with the identification of some gaps in research from an industrial point of view.

1 BACKGROUND

Our industry is traditionally dominated by mechanical engineers. The first controllers in the vehicle were also mechanical – just think of the bimetal-operated valves for controlling the engine coolant flow. When electronic control was introduced, control structures with many parameters were developed such that the mechanical engineers still had their knobs with which they could tune the behavior in the vehicle. This is not really a systematic synthesis method for controllers, but it was suitable for the purpose and it allowed for easily reusing the controllers in other vehicle models.

However, the approach reaches limits by now. Calibration guides for the control software contained in an engine control unit amount to several thousand pages, and the number of tuning knobs is overwhelming. No one can have a complete overview of all the control features anymore. Nevertheless, the way of working in our industry is optimized around this approach.

In parallel, academia developed true controller synthesis methods. These are methods that are based on a model of the plant and produce controller structure and controller parameter values in one go and optimized for the plant. An early method was the LQ regulator, later came LQG/LTR, H_2 and H_∞ controller synthesis, model predictive control, and many others. In all these methods, weights are used to tune the resulting controller. However, these weights or parameters are not directly related to the properties automotive calibration engineers need to influence. Moreover, they usually cannot be tuned online for an existing controller, but only at the time of controller synthesis.

These controller design methods have not been adopted much in the automotive industry. The next section thus discusses some controller properties that are quite essential for the way of working in the automotive industry.

2 REQUIREMENTS

The way of working implies certain requirements on controllers and their development. Controller synthesis methods that do not respect these requirements are less likely to be used.

2.1 Calibratability

In order to be able to split the control system development work between control engineers who design the control structure and calibrators who fine-tune the behavior, the controller has to be calibratable. At the end of the work by the control engineers, the control structure is compiled, linked, and then flashed into the ROM (Read Only Memory) of the ECU. The calibrators do not change this code structure anymore; they change the values of the parameters in the control structure, and they can flash these calibration values to a different region in the ECU's ROM.

This need for calibratability is one of the main reasons why H_∞ control or other model-based controller synthesis methods hardly get applied in automotive control even if the plant was sufficiently linear for such a linear time-invariant control method to work. In order to tune the controller, some design weights would have to be changed (Christen, 1996) – either only their values or even their structure as well –, then some Riccati equations have to be solved to get the controller parameter values. Calibration tools available today are not set up to do this, and if the structure of the weights was changed, the controller structure would be changed such that a re-compilation of the control structure most likely would be needed.

Moreover, these weights usually do not influence the properties to be calibrated directly, but only in an indirect way such that it would be difficult for calibrators to achieve their goals without diving deeply into the theory of weighting selection. The influence of the calibratables should be clear and ideally without side effects.

In the further future, the lines between calibrators and control strategists may get blurred such that the distinction between control structure design and calibration will be less relevant. But until then, calibratability is an important requirement.

2.2 Transferability

Another reason for the need for calibratability is the practice to re-use the same control structure for different variants of the same vehicle line or even for different vehicle lines; suppliers want to use the same control structure for several of their customers. The adaptation to the different vehicle models is done by developing specific calibrations for each.

Another implication of this practice is that the control structure must be the superset of what is used in all the vehicle models.

2.3 Maintainability and Extendibility

As a third consequence of the practice to re-use the control structure with different vehicle models, it must be possible to extend the control structure with new functionality needed for new vehicle models, *i.e.*, the superset of all functionalities must be extendible. The effort and cost for such extensions should be low.

As a consequence of the need for extendibility, engineers newly assigned to do such an extension should be able to quickly understand the existing control structure. Hence, its complexity must be limited, and it should be easy to document design decisions that lead to its structure.

2.4 Sensor Set

A large part of the control algorithms in a vehicle is for diagnostic functionality, some of which is required by law. The sensor set is often chosen with diagnostics in mind, and the functional control then has to live with this same sensor set. Besides driving costs, any additional sensor would be a potential source of malfunction, thus leading to additional diagnostics and limp-home strategies.

2.5 Resources

Computing power and storage memory are scarce resources in automotive control units. Most of the ECU are not floating point units even though the larger ones, such as the engine control unit, now use floating point arithmetic. These ECU are usually heavily loaded as they are used to control many features rather than only a single one. New control features should thus use the resources sparingly.

2.6 Development Process

Some parts in automotive control are safety critical. New control functionality is assessed in a hazard analysis. If the hazard classification exceeds a certain level, safety goals need to be formulated that may restrict the functionality originally planned. Moreover, the development process itself should follow state-of-the-art practices in such cases (*Fig. 1*). This means that adequate tools need to be used for requirements capturing, software versioning, and testing; these tools have to support traceability between the products of all these development steps: It must be possible to find the tests which verify that a requirement is met or to find the requirements and tests that go with a specific function in the software. – Müller-Lerwe *et al*. (2008) for a detailed discussion of these safety aspects on the example of a micro-hybrid vehicle.

2.7 Testability

As just mentioned, safety critical software needs to be verified in extensive testing. It is much more efficient to test the control software with a dedicated test harness in simulation than to perform tests in the vehicle. It is also much easier and cheaper to reach high test coverage in simulations than in a vehicle. And some types of tests, such as formal verification, can only be done on a computer.

However, in order to be able to test and to do it efficiently and with high automation, the software needs to be architectured and designed with testing in mind. By modularization, small entities can be thoroughly tested before they are assembled and further tested in the integrated software. More about testing can again be found in Müller-Lerwe *et al*. (2008).

2.8 Level 2 Monitor

Safety critical software needs to be supervised by a plausibility checker that runs in parallel and can suspend the main functionality if it reaches implausible decisions. Such a concept has been elaborated in the context of electronic accelerator pedals and is known as the e-gas concept. It is covered by AUTOSAR (2010), for instance.

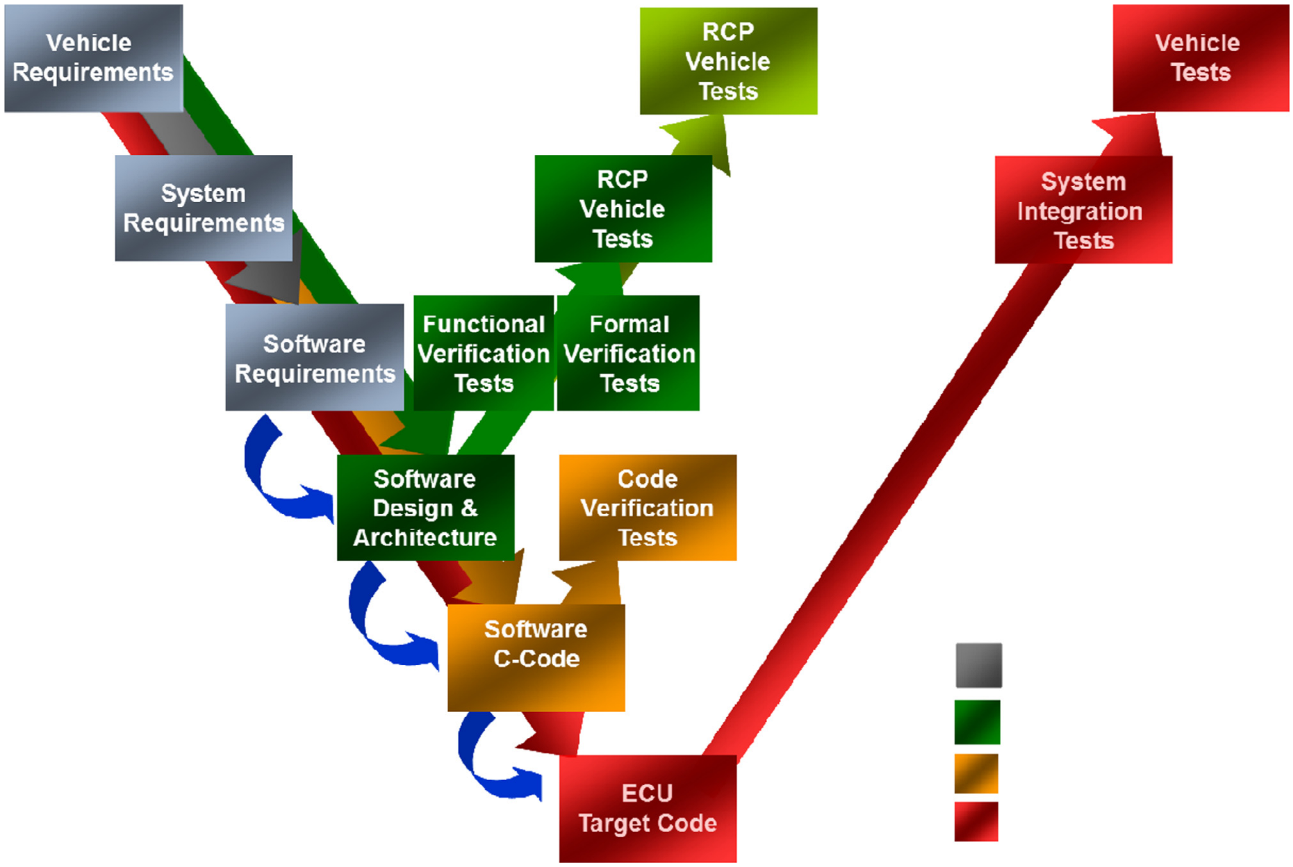

Figure 1

Cascade of functional requirements and corresponding tests.

The plausibility watchdog, also known as level 2 monitor, defines limits for what the main functional software (level 1) can do, and it intervenes if these limits are violated. In addition, there may be a level 3 monitor that supervises the correct working of the processor on which the level 1 and possibly the level 2 software is executed.

3 EXAMPLES

We want to look more closely into three examples that help to illustrate the requirements introduced above and their consequences. The first example deals with H_∞ control that was used in the introduction. This was the hype of the nineties; it is currently replaced by a similar hype for Model Predictive Control (MPC). We will look into this in the second example. The third example deals with energy management strategies for hybrid electric vehicles.

3.1 Model-Based Controller Synthesis: H_∞ Control

Even though direct application of H_∞ control failed because it cannot cope with calibratability, Roduner *et al.* (1997) have reported on the indirect synthesis of an H_∞ controller. They used a low-order model to describe an engine with the three-way catalyst, synthesized H_∞ controllers for many operating points and mapped them to a controller of a fixed structure based on proportional (P) and integrating (I) elements. Because of this fixed controller structure, they were also able to introduce a single parameter that allowed them to choose the bandwidth of the control system. Inspired by H_∞ controllers, they thus arrived at a fixed-structure controller that:

- retained some robustness properties of the H_∞ controllers;
- was scheduled with the engine operating point;
- could be calibrated by a calibrator.

This last point actually encompasses two aspects. Firstly, the fact that a parameter exists that allows for adapting the bandwidth of the control system. Secondly, that this parameter is used in a control structure that is understood by calibrators such that they feel comfortable to use it.

Transferability of the controller from one engine family to another is achieved because a very simple plant model is used such that only the time constant and a time delay need to be chosen for another engine. These two parameters also enter the controller as parameters such that the transfer to other engines does not involve the redesign of H_∞ controllers by solving Riccati equations.

Obviously, this way of using H_∞ control has not much to do with the way it is taught in control engineering classes. Augmenting the plant with appropriate weights and pressing the button to get the Riccati equations solved are just the easy part of obtaining this controller. It then had to be analyzed, mapped to the fixed structure, and a parameterization with the plant parameters and for the bandwidth had to be found.

An alternative approach to calibratable H_∞ controllers is the use of Linear Parameter-Varying (LPV) models and methods for the synthesis (Christen, 2002). The plant itself may be linear time-invariant or parameter-varying; the weights that are used to specify the performance of the control system contain one or several parameters that introduce the calibratability. Thus, the augmented plant shown in Figure 2 is LPV and corresponding algorithms based on linear matrix inequalities need to be used to calculate the controller.

The weights in this augmentation scheme assign minimum ($\omega_{f,l}$) and maximum ($\omega_{f,u}$) bandwidths, respectively. They are parametrized with $f_\omega \in [0.5, 2]$ such that these bandwidths can vary:

$$W_{\bar{r}}(s) = \frac{s/\left(f_\omega \omega_{f,u}\right)}{s/\left(100 f_\omega \omega_{f,u}\right) + 1}, \quad W_r(s) = \frac{100}{s/\left(f_\omega \omega_{f,l}/100\right) + 1}$$

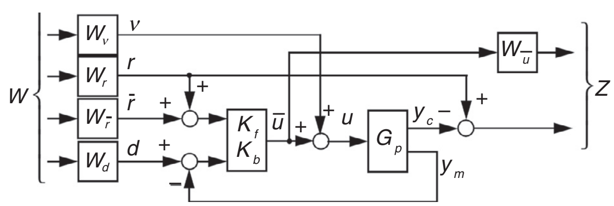

Figure 2

Augmentation scheme for the design of calibratable LPV H_∞ controllers (G_P: plant, K_f, K_b: feedforward and feedback parts of the controller, W_i: LPV weights).

where $W_{\bar{r}}(s)$ prescribes the maximum bandwidth and $W_r(s)$ the minimum bandwidth for feedforward control. The other weights are parameterized in a similar way.

In the example chosen in Christen (2002), the method works. In simulation, the bandwidth of the control system could indeed be adjusted by a factor of ½ to 2. However, this was all in continuous time. Discretization for the implementation on an ECU is less clean; it involves cumbersome matrix manipulations including matrix inversions that need to be executed at runtime. Moreover, the synthesis of LPV controllers with linear matrix inequalities is rather conservative such that the achievable performance is lower than for time-invariant H_∞ controllers calculated by solving Riccati equations.

But even if this worked, the true problem often is not so much the controller itself that controls one or several signals to track the reference signals. Generating online adequate reference signals capturing the true control objective is often only achieved with a rather high calibration effort. The true control objective may be to maximize drivability while satisfying legislated emission limits on a legal drive cycle. In order to achieve this, lookup tables scheduled on demanded torque, engine speed, engine temperature, ambient pressure, etc. need to be calibrated that provide setpoints for fuel injection and spark settings, boost pressure, exhaust gas recirculation, valve timing, and many other signals. In comparison to the calibration effort for this, the tuning of the actual controllers is trivial.

3.2 Model-Based Controller Synthesis: MPC

In recent years quite a few publications have appeared that describe MPC applications in an automotive context (del Re *et al.*, 2010). Model predictive control has replaced the Linear Time-Invariant (LTI) controller syntheses such as H_∞ or H_2 (LQG/LTR) control as the method of choice for research applications. MPC can deal with LTI plant models (or nonlinear models in the case of nonlinear MPC) that are subject to input saturation or other constraints; it boils down to a constrained optimization. There are two approaches to MPC, with implicit and explicit calculation of the actuator signals, respectively.

The original MPC with implicit calculation of the control move needs online optimization. Obviously, the computing power required for this was prohibitive for automotive applications for a long time. As an alternative, explicit MPC was developed (Alessio and Bemporad, 2009) where the optimization is done offline. The controller then is stored as maps such that the actuator signals can be looked up during operation and do not need to be computed in an optimization.

Model predictive controllers can be made calibratable (Naus *et al.*, 2010). The parameters used for calibration could be incorporated in the cost criterion for the online or offline optimization and adjust the relative weight of the controlled signals *versus* the manipulated signals. Naus *et al.* (2010) use this approach, but they complement it by incorporating the parameters also in the model itself and in some of the constraints. This allows them to have simple tuning knobs for calibration but still full flexibility to formulate the MPC problem – cost criterion and constraints.

While calibratability is possible with both, implicit and explicit MPC, the requirement of testability (*Sect. 2.7*) speaks in favor of explicit MPC. After the offline optimization, the final controller is available for extensive testing in simulation, and it will always provide the same response for given controller input signals, and the response time is well defined. This may be different for the online optimization. Depending on the algorithm used, both the response and the response time may vary even for the same controller inputs. (Nevertheless, there is a potential advantage of implicit MPC: it is easier to incorporate reconfigurability in case of sensor or actuator failure.)

However, explicit MPC is relatively limited in its applicability. The offline optimization, which amounts to finding the parameters in lookup tables scheduled on reference and state variables and constraints, suffers from the curse of dimensionality. Their number quickly blows up if many constraints are used. The huge lookup tables generated tend to occupy more memory than ordinary ECU have available for a single feature, and finding the parameter value corresponding to the operating point takes quite some computation time as well. Hence, explicit MPC is limited to rather small problems and thus is not the all-encompassing solution. As an alternative, fast optimization algorithms are being developed (Mattingley *et al.*, 2011) such that MPC problems with slower dynamics might be solved implicitly in the future.

A further complication with MPC is the fact that it constructs state feedback laws. Because in most cases, not all state variables can be measured (see also *Sect. 2.4* above), observers need to be developed which provide estimates for those state variables that are not measured directly.[1]

[1] This is different for H_2 and H_∞ controllers since they include a compatible observer that is automatically generated during the controller synthesis.

3.3 Energy Management for Hybrid Vehicle Applications

Energy management for hybrid electric vehicles is usually formulated as an optimization problem (Sciarretta and Guzzella, 2007). In the most simple formulation, fuel consumption over a given drive cycle is minimized while meeting the speed profile of the cycle (and hence the requested propulsion torque) and bringing the battery state of charge at the end of the cycle back to the initial level. The cost function for the optimization could thus be:

$$J = \sum_{0}^{T} \dot{m}_f(x,u) + \phi_{SOC}(x_T)$$

where T is the final time of the cycle, $\dot{m}_f(x, u)$ is the fuel flow as a function of the state x (usually the State Of Charge (SOC) of the battery) and the actuator settings u. The function $\phi_{SOC}(x)$ indicates the SOC related cost; it is only applied at the end of the cycle.

The result of such an optimization is actuator settings scheduled *versus* time rather than based on system states and input signals. With MPC, this is changed: since the optimization is repeated at each sampling instant and then the actuator setting for only the first step is implemented, it is indeed a feedback law (Borhan *et al.*, 2012).

This most simple formulation of the optimization problem neglects many aspects that are relevant for implementations that would be acceptable to OEM. These aspects are usually subsumed by the notions of drivability and NVH (noise, vibration, harshness). For instance, the optimal strategy may lead to switching on and off the combustion engine at a rather high frequency, or gears may be shifted rather frequently. Most drivers would get annoyed with such a behavior. They would find it counter-intuitive and question its optimality with respect to fuel economy.

Engine on/off and gear shifting frequency is taken into account for instance in Opila *et al.* (2011) where they include additional terms in the cost function:

$$J = \sum_{0}^{T} \left(\dot{m}_f(x,u) + \alpha I_{GE}(x,u) + \beta I_{EE}(x,u)\right) + \phi_{SOC}(x_T)$$

where I_{GE} and I_{EE} are indicator functions for gear events and engine events, respectively. The drivability terms are weighted with the factors α and β, which need to be chosen by the control engineer.

Opila *et al.* (2011) are not using MPC, but what they term shortest path stochastic dynamic programming, with which they synthesize causal controllers. Their final controller shows a drivability very similar to an industrial benchmark control policy, but with improved fuel economy.

Another approach is used in Dextreit *et al.* (2008) and Dextreit and Kolmanovsky (2013). The authors use game theory in an MPC like setup – the control actions $u \in U$ try to minimize the cost criterion over a short horizon while the antagonist (driver or environment) tries to maximize it by choosing its values $w \in W$. Mode chattering is prevented by dynamically changing the constraints, *i.e.*, the set U from which u can be chosen.

Their game theory controller attains the highest fuel saving of all the causal control strategies the authors compare; only a noncausal deterministic dynamic programming controller achieves slightly lower fuel consumption.

Calibratability is not discussed in the references mentioned in this section. However, it should be possible to incorporate it in a similar way as described in Section 3.2.

4 DISCUSSION

The examples illustrate the change in controller synthesis methods that got applied in the past and get applied currently in research. At the same time, and more importantly in the context of this paper, they illustrate the lack of a systematic treatment of the aspects mentioned in Section 2. Of all those aspects, the development process and the level 2 monitoring (*Sect. 2.6, 2.8*) are quite independent of the controller design method and thus will not be discussed any further. Calibratability (*Sect. 2.1*) for H_∞ control and MPC has been examined to some extent but we feel that the proposed solutions are rather *ad hoc*. For transferability and maintainability (or extensibility) from Sections 2.2 and 2.3, there are not even proposed solutions. If these requirements are met with a controller, then it is by chance.

Testing of control strategies and software implementing them is a research topic in itself (Clarke *et al.*, 2000); tools like Matlab provide extensions that support the systematic testing of the decision parts and the continuous control parts of the control strategies. However, to the authors' knowledge, controller synthesis methods do not systematically take testability (*Sect. 2.7*) into account.

The final requirement to be discussed is the restricted sensor set (*Sect. 2.4*) that often precludes the application of state feedback. Observers are needed to reconstruct the state. Observers that match the state feedback part of the controller are automatically synthesized for H_∞ or H_2 control. For other methods, including MPC, special care must be taken to obtain robust control systems and avoid pitfalls as the one described by Doyle (1978).

4.1 Suggested Research Topics

So far, we have been trying to incorporate the required abilities (calibratability, etc.) into existing controller design methods. However, one should start with asking more fundamental questions:

- is it possible to construct a controller design method that incorporates these abilities by construction and still is a true controller synthesis method?
- if the abilities are added to an existing controller synthesis method, which method is best suited as a starting point?
- again, if the abilities are added to existing methods, is it possible to develop a bolt-on approach that allows for introducing the abilities independent of the controller design method? This might be similar to some approaches to anti-windup (Kothare *et al.*, 1994).
- in order to answer these questions in a systematic way, metrics may have to be developed that capture the degree and ease of meeting the requirements. These metrics would then allow the comparison of different approaches.

Apart from these fundamental questions, approaches to meet the individual requirements need to be developed. Just two topics out of a long list are:

- how to introduce single parameters such that calibrators can easily influence control system properties in a targeted manner?
- how to design observers for MPC? The separation principle that is essential for the observer design for linear systems does not hold for nonlinear systems (Magni and Scattolini, 2010), *i.e.*, for MPC with linear or nonlinear plant models subject to constraints. Is it nevertheless possible to find an approach that allows for the simultaneous design of observer and state feedback as for H_∞ control?

4.2 Research Split between Academia and Industry

These questions are not something industry can answer; they are too fundamental and theoretical in nature. Most engineers in industry have been out of university for too long to be able to efficiently tackle such questions. Universities with their many fresh minds are better suited to investigate relatively narrow topics with rigor and theoretically well founded.

Industry, on the other hand, has its strength in integration, an area where universities cannot be very strong because it needs a workforce with wide experiences and very familiar with established processes; if this is lacking, the coordination of many people from different areas cannot be efficient.[2]

We thus see academia as the predominant knowledge generator for control engineering. A further important role of academia is in the pruning and final presentation of knowledge: the education of students and engineers in industry. The role of the industry is then to integrate the knowledge for and in the products they sell. This is quite obvious for today's cars with their ever increasing content of control and software.

Another approach to tackling some of these questions may be open innovation (Ili *et al.*, 2010) – inviting the public, including universities, to participate in the innovation process. This approach is not pursued often yet because the industry is uneasy about the intellectual property situation associated with it.

CONCLUSIONS

This last section shall not be a summary of the paper. Instead we would like to take up the title of the paper: the art of control engineering. What is "art" in control engineering?

When control engineers are looking for the best solution to an application, they need a large toolbox with tools provided by both, academia and industry: analytical tools, synthesis tools, and procedural tools that guide in the way of working. The art lies in the way of choosing and combining these tools, but also in recognizing when deviations from existing tools are adequate or new tools need to be developed.

In order to best practice this art, the personal toolbox of every control engineer should be filled with powerful tools. This can only happen in a dialog between academia and industry: for industry to learn about the latest theoretical developments; for academia to hear about the way of working and tools that are missing for industrial applications. Both parties should work together to keep the toolbox filled with relevant tools.

Science meets industrial reality – this is thus not meant as a disillusionment and dismissal of advances in academia, but rather as an appeal to fruitful exchange.

[2] In view of this, we regret that university projects tend to get larger in recent years. It probably has to do with funding agencies that find it easier to judge (and sell to their governing bodies) single large projects rather than many small ones behind which they would have to find the overarching vision themselves.

ACKNOWLEDGMENTS

We would like to thank our colleagues, especially Karl Siebertz, Craig Stephens, and Davor Hrovat, for helpful discussions.

REFERENCES

Alessio A.,Bemporad A. (2009) A survey on explicit model predictive control, Raimondo L.M., Magni L., Allgöwer F. (eds), *Nonlinear Model Predictive Control: Towards New Challenging Applications, in Lecture Notes in Control and Information Sciences* **384**, 345-369, Springer-Verlag, Berlin.

AUTOSAR (2010) *Technical Safety Concept Status Report*, AUTOSAR Development Partnership Document No. 233, Version 1.1.0, Release 4.0, Rev. 2.

Borhan H., Vahidi A., Phillips A.M., Kuang M.L., Kolmanovsky I.V., Di Cairano S. (2012) MPC-based energy management of a power-split hybrid electric vehicle, *IEEE Transactions on Control Systems Technology* **20**, 3, 593-603.

Christen U. (1996) *Engineering Aspects of H_∞ Control*. Diss. ETH No. 11433, Zurich, Switzerland.

Christen U. (2002) Calibratable model-based controllers, *Proceedings of the 2002 IEEE International Conference on Control Applications*, Glasgow, pp. 1056-1057.

Clarke E.M., Grumberg O., Peled D.A. (2000) *Model Checking*, MIT Press, Cambridge, MA.

Cook J.A., Sun J., Buckland J.H., Kolmanovsky I.V., Peng H., Grizzle J.W. (2006) Automotive powertrain control – A survey, *Asian J. Control* **8**, 3, 237-260.

del Re L., Allgöwer F., Glielmo L., Guardiola C., Kolmanovsky I. (eds) (2010) *Automotive Model Predictive Control: Models, Methods and Applications,* in *Lecture Notes in Control and Information Sciences* **402**, Springer-Verlag, Berlin.

Dextreit C., Assadian F., Kolmanovsky I.V., Mahtani J., Burnham K. (2008) Hybrid electric vehicle energy management using game theory, *2008 SAE World Congress*, Detroit, Michigan, USA, *SAE Paper* 2008-01-1317.

Dextreit C., Kolmanovsky I.V. (2013) Game theory controller for hybrid electric vehicles, *IEEE Transactions on Control Systems Technology*, prepeint.

Doyle J. (1978) Guaranteed margins for LQG regulators, *IEEE Transactions on Automatic Control* **23**, 4, 756-757.

Hrovat D., Tseng H.E., Lu J., Deur J., Assadian F., Borrelli F., Falcone P. (2011) Vehicle controls, Levine W.S. (ed.) *Control System Applications*, CRC Press, Boca Raton, Chap. 3.

Ili S., Albers A., Miller S. (2010) Open innovation in the automotive industry, *R&D Management* **40**, 3, 246-255.

Klatta K.-U., Marquardt W. (2009) Perspectives for process systems engineering—Personal views from academia and industry, *Computers Chemical Engineering* **33**, 3, 536-550.

Kothare M.V., Campo P.J., Morari M., Nett C.N. (1994) A unified framework for the study of anti-windup designs, *Automatica* **30**, 12, 1869-1883.

Magni L., Scattolini R. (2010) An overview of nonlinear model predictive control, in del Re L., Allgöwer F., Glielmo L., Guardiola C., Kolmanovsky I. (eds), *Automotive Model Predictive Control: Models, Methods and Applications,* in *Lecture Notes in Control and Information Sciences* **402**, Springer-Verlag, Berlin.

Mattingley J., Wang Y., Boyd S. (2011) Receding horizon control: automatic generation of high-speed solvers, *IEEE Control Systems Magazine* **31**, 3, 52-65.

Müller-Lerwe A., Busch R., Rambow T., Christen U., Gussen U., Kees M. (2008) Safety aspects on a micro-hybrid vehicle with manual gearbox, *2008 SAE World Congress*, Detroit, Michigan, USA, *SAE Paper* 2008-01-0118.

Naus G.J.L., Ploeg J., Van de Molengraft M.J.G., Heemels W. P.M.H., Steinbuch M. (2010) Design and implementation of parameterized adaptive cruise control: An explicit model predictive control approach, *Control Engineering Practice* **18**, 8, 882-892.

Opila D.F., Wang X., McGee R., Gillespie R.B., Cook J.A., Grizzle J.W. (2011) An energy management controller to optimally trade off fuel economy and drivability for hybrid vehicles, *IEEE Transactions on Control Systems Technology* **20**, 6, 1490-1505.

Roduner C.A., Onder C.H., Geering H.P. (1997) Automated design of an air/fuel controller for an SI engine considering the three-way catalytic converter in the H_∞ approach, *Proceedings of the 5th Mediterranean Conference on Control and Systems, Paphos*, Cyprus.

Sciarretta A., Guzzella L. (2007) Control of hybrid electric vehicles: optimal energy-management strategies, *IEEE Control Systems Magazine* **27**, 2, 60-70.

Zhu J.J. (ed.) (1996) Special issue on control education, *IEEE Control Systems Magazine* **16**, 2, 8-56.

Manuscript accepted in October 2013
Published online in April 2014

Cite this article as: U. Christen and R. Busch (2015). The Art of Control Engineering: Science Meets Industrial Reality, *Oil Gas Sci. Technol* **70**, 1, 31-39.