Cross-Application Perspectives: Application and Market Requirements

Greg Stewart and Tariq Samad

Introduction

Is deep controls expertise sufficient to make an impact on industry and society? Control technology has, after all, had a transformational effect on several application domains and is seen as a crucial enabler for dramatic advances in several others. A common core—the fundamentals of control science and engineering—underlies these past successes and future prospects.

Yet the answer to the question is an emphatic no. Success in practice requires considerably more than generic controls expertise. Exploiting the intellectual richness of the field is contingent on gaining a deep understanding of the intricacies and idiosyncrasies of specific domains.

> *The success of control in practice is contingent on gaining an understanding of the intricacies and idiosyncrasies of specific domains.*

Different application areas differ in ways that are often underappreciated. These points of difference are a mélange of technical and nontechnical factors. A short list includes industry supply chains, hardware and software constraints, engineering organizational structures, sensor and actuator quality and availability, the prevalence of legacy versus new systems, first-principles understanding, educational level of staff, the availability of operational data, and regulatory requirements.

In this section we address the issues and requirements involved in realizing practical, successful industry deployments of new control technology. We contrast selected domains with regard to application and market requirements and discuss aspects of industry applications that are critical to understand in attempting to achieve impact with advanced control.

The Role of Context in Control Engineering Innovation

> *As control engineering is fundamentally about the integration of many elements—plant, sensors, actuators, computing platform, algorithms—it is essential that control engineering researchers fully appreciate all aspects of the environment they wish to improve.*

By definition, *innovation* involves changing an existing situation. As control engineering is fundamentally about the integration of many elements—plant, sensors, actuators, computing, algorithms—it is essential that control engineering researchers fully appreciate all aspects of the environment they wish to improve. This includes understanding the current control design process and performance criteria, then evaluating the changes that are incurred with

From: *The Impact of Control Technology*, T. Samad and A.M. Annaswamy (eds.), 2011. Available at www.ieeecss.org.

the proposed innovation. Csikszentmihalyi uses an old Italian expression: “Impara l'arte, e mettila da parte” (learn the craft, and then set it aside). What tasks or expenses will the innovation simplify or eliminate? What new tasks will be introduced as a result of the innovation? Will the innovation bring a net benefit (usually measured in money) to the industrial application?

To answer these questions, one must not consider an innovation in isolation, but instead must evaluate the overall benefit of the new system created by integrating the innovation into the previous system.[1]

When developing an advanced control innovation, one should bear in mind which portions of the current control design process will need to be changed or replaced. To cite a few examples:

1. Inventing a new PID controller tuning technique may affect only the person responsible for tuning that loop.
2. Introducing a new H_∞ or nonlinear controller would have the impact of 1 above and further require that the real-time control software be changed, a technique for obtaining plant models be in place, and often an industrial-quality (intuitive and error-free) tuning tool that enables nonexperts to tune the advanced controller be available.
3. The introduction of a computationally intensive technique such as model predictive control (standard, not explicit MPC) would have the impact of 1 and 2 and may require an upgrade of the hardware platform to host the algorithm.

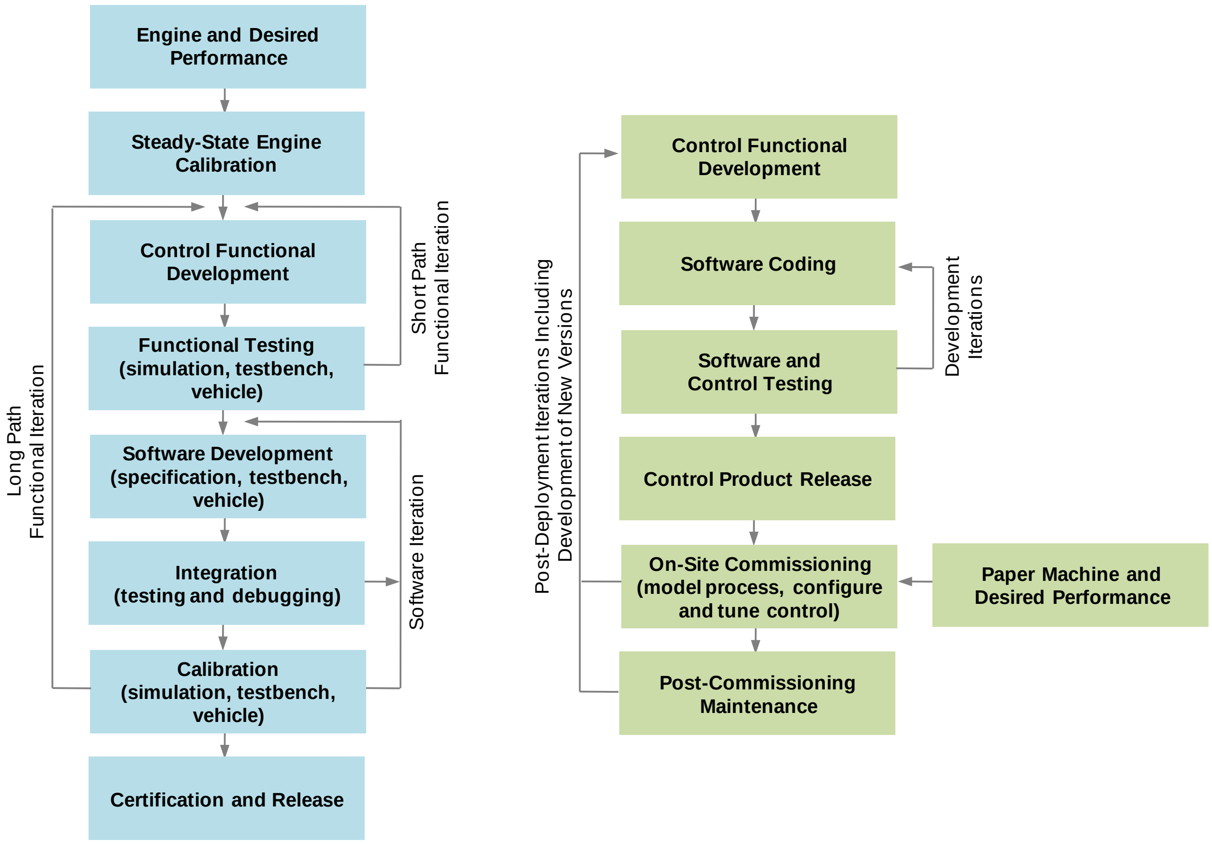
To surface some of the key ideas, we can contrast various control design processes. Fig. 1 illustrates the high-level workflow for the development of (1) heavy-duty engine control, and (2) papermaking control. In both cases, many of the familiar design steps are present, but a stark contrast exists in the position of the plant itself. In engine control, a production or prototype engine is available at the start of the control development process, and the control development proceeds with the use of engine measurements and experiments until the tailored and tuned control strategy is released along with the engine production fleet. Conversely, in process control, each plant is often custom designed, and thus each plant is usually very different from every other plant. Advanced control tools are typically developed with the facility to accommodate these plant-to-plant differences by virtue of including model identification software and the ability to straightforwardly configure the controller structure (number of setpoints, actuators, constraints) at the time of commissioning the control. Furthermore, once an engine control is released to market, relatively few opportunities exist for modifying the controller. In process control, the expectation is that during the “post-commissioning maintenance” phase, the models will be re-identified and the control retuned on a frequent, sometimes even weekly, basis.

Generally, an innovation must consider which portion(s) of the current system it will change or replace. The changed system must be “complete” in the sense that the user must be able to perform his/her tasks from beginning to end. Two common examples of incomplete innovations can be cited: (1) a control innovation whose tuning requires Ph.D.-level control expertise where such does not exist at the application, and (2) a complex advanced control algorithm whose memory and processor requirements are too large for the target hardware platform. These classes of innovations cannot be adopted on their own but instead require additional work to become industrially viable.

---

[1] Here the term *system* is understood in its broad sense to mean an overall situation that may include a design process or accepted method for performing a task or set of tasks.

Furthermore, before performing the work and incurring the expense required to adopt an innovation, an organization will weigh the potential value the innovation is expected to bring. A successful innovation will bring more value than it costs, where these criteria are considered along the usual dimensions that include equipment costs, development time, performance, training, and personnel costs.



(a) Engine Control Development Process

(b) Papermaking Control Development Process

Figure 1. Two example industrial control development processes. Many of the activities are included in both situations, but the ordering of the steps is quite different. In particular, the point at which plant-specific information is available is quite different (early for engine control, late for process control) and thus influences the structure of the respective control development processes.

The Elements of Control in Practice

Generally, the impact of an innovation must be evaluated on a case-by-case basis for each potential industry application. However, although each application has its unique characteristics, we find that the control-relevant facets of engineering problems follow some general categories:

- Plant,
- Sensors and actuators,
- Hardware platform,
- Software structure and process,

- Controller tuning (including model identification), and
- Certification.

A common theme in control engineering is evident when a change in one aspect of the application environment enables changes (whether intended or not) in other aspects. For example, both papermaking control and thickness control in steel cold tandem mills initially relied on a proprietary software platform that made it challenging to introduce advanced control. Once open software platforms were introduced, it became much easier to introduce advanced control at the software application level, and both industries now employ robust control and multivariable control in many applications.

Plant

The challenges presented to the control engineer by the plant are generally well known. Some of the leading considerations include the degree of nonlinearity; the complexity of the dynamics; the magnitude of model uncertainty; the constraints on input, output, and states; and the condition number of multivariable plants.

Despite its maturity as a discipline, control engineering is often a technology that is considered only after the plant has been designed. The design of a plant such that it can be effectively controlled is still rare in many applications.

Emerging needs:

- Co-design of plant, sensors, actuators, and control for desired closed-loop performance.
- Control-oriented modeling in terms of physical-based parameters. This would enable a common language between plant designers and control engineers.

Sensors and Actuators

Sensors and actuators are the “handles” by which a control algorithm accesses a plant. Both classes of instrumentation will have requirements in terms of cost, range, bandwidth, and reliability. When considering actuators, it is especially important to understand the role of typical nonlinearities in the control loop—backlash can often be accommodated by detuning the control algorithm, whereas stiction may not.

The performance of sensors is particularly important as feedback control is designed to translate the sensor information into the operation of the plant itself. Sensor accuracy, bias, and cross-sensitivities to their anticipated environment must be considered by the control engineer during the design.

Emerging needs:

- Smart sensors with onboard observers.
- Integration of hardware sensors with inferential sensing for redundancy.
- Networks of wireless sensors.

Computational Platform

The parameters of the intended computational platform are a key consideration when developing a control algorithm. Processor speed, memory, sampling time, architecture, and redundancy all play a role in determining the feasibility of implementation of the algorithm. In the automotive industry, the processor speeds may be in the range of 40 to 56 MHz and 2 to 4 MB of flash memory may be available for control to be executed within milliseconds. On the other hand, modern equipment in the process industries may have a 2.83-GHz processor and 3 GB of memory to execute control actions in seconds or minutes. Very different control approaches may be considered in each case.

The design of control for embedded processors may require the consideration of additional computational aspects such as numerical accuracy in fixed- or floating-point applications.

Emerging needs:

- Hardware-specific algorithm design (for example, designing control algorithms that are robust to fixed-point implementation).
- Control-specific hardware design.

Software Development Process

Since modern control is typically implemented as algorithms in a software application, the importance of the software development process is central. Typically this process follows the phases of proof of concept, application prototyping, testing, software specification, software coding for target, software testing, and finally performance testing. In some industries, the software development process is identified as a key bottleneck in reducing time to market. This is one of the areas where modern control engineering could be expected to make a contribution.

In many applications, a control engineering innovation must accommodate the requirement to develop code to an industry standard. In particular, applications in industries such as aerospace and automotive are finding validation and verification tasks accounting for around half the cost of overall product development.

The issue of interfacing a new control strategy to legacy software is very important. In industrial situations, advanced control is often introduced into an existing software environment, and thus it becomes crucial to define the scope of the control innovation early and evaluate its impacts on the overall software environment. For example, replacing several SISO control loops with a MIMO controller is not always straightforward. In legacy systems, the SISO loops may exist in disparate portions of the overall software environment and can be expected to be connected to legacy diagnostics functionality, which must be maintained in the new control development.

Emerging needs:

- Verifiable control design methods.

Controller Tuning (Including Model Identification)

Given that a control strategy's success or failure can be determined by how its tuning parameters are set, it is surprising how much more attention the research community typically devotes to the development of the core control algorithm, often leaving the setting of the tuning parameters to the end users' discretion. In many industrial situations, the personnel responsible for controller tuning may have little or no advanced control training yet are responsible for delivering acceptable closed-loop performance. This simple fact goes a long way toward explaining the persistence of PID control and simple tuning rules in industrial practice.

Emerging needs:

- Techniques that guarantee closed-loop performance while requiring “industry-realistic” control knowledge.
- Computationally efficient tuning algorithms.
- Systematic and reliable modeling and tuning.

Acknowledgments

This section is the result of a breakout session held at the International Workshop on the Impact of Control: Past, Present, and Future, October 18–20, 2009, in Berchtesgaden, Germany. The authors gratefully acknowledge the contributions of Christian Philippe, Francesco Cuzzola, Paul Houpt, Clas Jacobson, Keith Glover, L.K. Mestha, Boris Lohmann, Manfred Morari, Paolo Coeli, Bob Yeh, Thomas Mannchen, and Ilya Kolmanovsky. Thanks also to Johan Backstrom of Honeywell for providing additional insight into the history of industrial control in the process industries.