Teaching Model Predictive Control

WHAT, WHEN, WHERE, WHY, WHO, AND HOW?

TIMM FAULWASSER, ERIC C. KERRIGAN, FILIP LOGIST, SERGIO LUCIA, MARTIN MÖNNIGMANN, ALESSANDRA PARISIO, and MORITZ SCHULZE DARUP

Over the course of four decades, model predictive control (MPC) has become one of the great success stories in systems and control. It has grown from its native habitat (chemical process control) into all domains of control applications—power and energy systems, mechatronics and robotics, as well as aerospace and aeronautics. Hence, in a modern systems and control curriculum, MPC triggers not so much the question of *if* it should be taught. In fact, industrial demand for and the continued research potential of MPC suggest that one should rather ask the Aristotelian 5W1H (what, when, where, why, who, and how?) about teaching MPC. This article presents insights into the 5Ws distilled from the results of a survey on teaching MPC conducted in the systems and control community. Moreover, the *how* is approached through blueprint suggestions for curricula for an undergraduate discrete-time linear-quadratic MPC course and for graduate courses covering the continuous-time nonlinear avenue and the learning-based route.

Digital Object Identifier 10.1109/MCS.2024.3402908
Date of current version: 19 July 2024

INTRODUCTION

The idea of MPC can at least be traced back to the 1960s, and both its theoretical basis [1], [2] and pioneering industrial applications [3], [4] were investigated from the start. Roughly 50–60 years later, its industrial success is evident, and already in his 2002 textbook, Jan Maciejowski [5] stated that "[MPC] is the only advanced control technique—that is, more advanced than standard PID control—to have had a significant and widespread impact on industrial process control."

Among other features, MPC stands out due to the effective combination of the following four key items:

1) the simplicity of the conceptual idea
2) its efficacy in handling complex systems subject to constraints
3) the availability of mature code for rapid prototyping and real-time feasible implementation
4) its flexibility to connect with emerging trends in systems and control, general engineering, and computer science [such as machine learning (ML) for disturbance forecasting and system modeling].

With respect to item 1), many control educators will agree that the conceptual idea of MPC to complement feedback by prediction is easy to understand for both undergraduate and graduate students. Similar experiences can be made in communicating with industrial practitioners not previously accustomed to MPC. Moreover, it stands to reason that the mathematical prerequisites required to understand a linear-quadratic MPC controller are not more complex than those needed for frequency-domain techniques in classical PID control. Yet, simplicity does not guarantee industrial success. Hence, the second important aspect of MPC is that it has proven itself to be a very useful method in manifold applications; cf. item 2). One core strength of MPC is the structured consideration of constraints on states and inputs, which is of interest in many application domains.

While the classic route to deriving optimal *feedback policies* via the Hamilton-Jacobi-Bellman equation or via the Pontryagin maximum principle (see, for example, [1], [6], [7], and [8]) is viable only in quite specific settings—that is, mostly linear dynamics or rather low-dimensional nonlinear systems—the concept of MPC yields receding-horizon feedback for a large class of systems.

Nowadays, real-time feasible MPC and nonlinear MPC (NMPC) applications go far beyond slow process systems; see [9] and references therein for overviews of early industrial applications. Achievable sampling rates for MPC range from GHz in the case of small-scale linear systems to the high-kHz range for nonlinear systems. Moreover, there exist several readily applicable software tools that simplify the implementation tremendously; cf. item 3). We refer to [10] for more details and references.

Turning to item 4), it is interesting to observe that while most research topics are subject to activity cycles—the infamous artificial intelligence (AI) winter [11] being a prominent example—MPC has not seen phases of reduced activity yet. As evident from Figure 1, closely related topics such as *optimal control* and *dynamic programming* have seen periods of varying scientific interest (as measured by the Google Ngram viewer, which counts the frequency of search terms in English texts). MPC, in contrast, has seen a steady growth since the early 1980s.

It is fair to ask why a research area in systems and control seemingly grows void of apparent activity cycles. The

Summary

We conducted a survey on teaching MPC and received more than 120 answers from lecturers and students from the systems and control community around the world. We first summarize the responses as neutral as possible in the "Insights From the Survey" section and then derive best practices for undergraduate MPC teaching, which are biased by the authors' unanimous opinion that MPC can and should be taught as early as possible. In fact, the survey outcomes reveal that MPC is even the first control course in some curricula today, corroborating that MPC may indeed be an attractive option for an introductory control course. The "Advanced Topics for Graduate MPC Courses" section proceeds with suggestions about topics and course layouts for students who already have basic, intermediate, or even expert knowledge about MPC. Finally, the "Prospects and Synergies for Future MPC Teaching" section briefly states the authors' view on some important future routes for MPC, among them new applications and the relation to machine learning (ML) and artificial intelligence (AI).

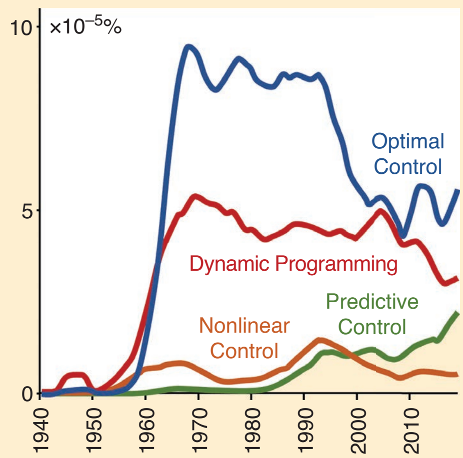

FIGURE 1 The relative frequency of occurrence of the terms *dynamic programming*, *optimal control*, *nonlinear control*, and *predictive control* in the period 1940–2019 according to the Google Ngram viewer.

answer can be found in the flexibility of MPC to connect with other control trends, which in turn led to the emergence of various branches, such as nonlinear, stochastic, robust, hybrid, distributed, economic, data-driven, and learning-based MPC. Likewise, MPC finds use in a variety of application domains: process engineering, mechatronics and robotics, aeronautics, logistics, power and energy systems, finance, cybersecurity, and supply chains. Arguably, this twofold diversity is a standout feature of MPC.

While this flexibility and compatibility have fueled research on MPC for more than three decades—one might go as far as saying that MPC is adaptive with respect to parallel research trends—it also renders teaching MPC increasingly complex, though necessary. This complexity does not imply that MPC courses have to necessarily be postponed to the graduate level. The need to teach MPC stems on the one hand from the requirement that higher education in engineering (undergraduate and graduate alike) should prepare for careers in industry. Systems and control would fail in this regard if MPC was not part of the core curriculum in control. On the other hand, it becomes increasingly difficult on the graduate level to adhere to *Humboldt's ideal of combining research and education* and to cover the entire spectrum of MPC branches. At the same time, hardly any research group will address the complete spectrum of MPC topics in its research. This raises the questions of what the research topics are that can be covered in a graduate course and what their relations with topics in systems and control and other neighboring disciplines are. Yet, prior to addressing these open points, it appears necessary to assess the current status and practice of teaching MPC.

It is *not* the purpose of this article to provide the receding-horizon optimal solution to teaching MPC. Neither does it elaborate on the question of whether MPC should be taught in dedicated courses. Rather, this article stands in line with the control systems tradition to discuss the education-related aspects of our domain; see, for example, [12], [13], and [14]. Here, however, we focus specifically on MPC.

In particular, the article reports the results of a survey on teaching NMPC conducted in the aftermath of a plenary discussion held at the 7th IFAC Conference on Nonlinear Model Predictive Control 2021, and hence, it can be seen as an extension and follow-up of our earlier conference paper [10] (see "Summary"). Specifically, we link the findings of the survey to proposals of different curricula for teaching MPC. That is, we essay to capture the 5W1H of teaching MPC. Moreover, we provide blueprint suggestions for addressing and covering MPC in undergraduate and graduate courses, thus addressing the "how." Given that MPC is of interest for all application domains of control and the fact that curricula may vary considerably even among departments of the same denomination, we do not tailor our blueprints to specific engineering domains. Instead, our suggestions focus on control-related perspectives, such as links to data-driven, learning, and dissipativity concepts, having readers in mind who do not yet consider themselves MPC aficionados. Hence, "MPC—Conceptual Idea" and "MPC Terminology—An Incomplete Glossary" provide a basic recap of the MPC concept and an at-a-glance overview of specific terminology.

The remainder of this article is structured as follows. We begin by reporting the setup and the findings of the conducted online survey on teaching MPC. We then turn toward three blueprints for MPC curricula in the following sidebars:

- "MPC for Undergraduates—From Discrete-Time LQR to Basic MPC With Some Advanced Flavors"
- "Continuous-Time NMPC—A Teaching Avenue With Stability and Dissipativity Stopovers"
- "Learning- and Data-Based MPC—Using Data to Alleviate the Challenges of MPC."

The first one picks up on the longstanding discussion of what should and what can be conveyed in undergraduate courses on control, see, for example, [15] and [16]. The second and third ones illustrate how recent research trends can be reflected in MPC courses.

INSIGHTS FROM THE SURVEY

We collected 128 answers to the survey. Invitations to participate were sent to the attendees of the last IFAC Nonlinear Model Predictive Control Conference, to the mailing list of the IFAC Technical Committee 2.4 Optimal Control, and to the IFAC Twitter channel. We summarize the results in pie and bar charts. Percentages in the pie charts add up to 100% up to rounding errors. Percentages in the bar charts do in general not add up to 100% because multiple answers were possible.

Most of the respondents are professors or lecturers (56%, Figure 2), followed by postdoctoral and doctoral students (together 27%). About 10% of the participants were students, and 7% held positions in industry.

The majority of our participants are located in Europe (57%, Figure 3), which may be attributed to all the authors being European and the survey being initiated through a panel discussion held at a conference in Europe. It may also reflect the research activity in the field in Europe. Most of our participants teach at electrical engineering departments (49%, Figure 4), followed by mechanical and chemical engineering departments (together 27%). Only 5% and 3% of our respondents work at mathematics or computer science departments, respectively. The headline "other" in Figure 4 refers to related engineering departments that we did not list as choices explicitly in the survey (aerospace, civil, and robotics, together 6%), to more specialized departments (systems, control, and automation, together 6%) or to general engineering departments (3%).

MPC COURSES AT THE PARTICIPANTS' INSTITUTIONS

The second part of the survey essentially assessed when in the curricula MPC appears and how intensively it is taught. Not surprisingly, dedicated MPC courses at the bachelor's

MPC—Conceptual Idea

MPC is based on the repeated solution of an optimal control problem (OCP). In the case of discrete-time systems, this OCP reads

$$V_N(x(t)) \doteq \min_{u(\cdot)} \sum_{\tau=0}^{N-1} \ell(x(\tau),u(\tau)) + V_f(x(N)) \tag{S1a}$$

$$\text{subject to} \quad \forall \tau \in \{0,\dots,N-1\}$$

$$x(\tau+1) = f(x(\tau),u(\tau)), \quad x(0) = x(t) \tag{S1b}$$

$$x(\tau) \in \mathbb{X}, \quad u(\tau) \in \mathbb{U} \tag{S1c}$$

$$x(N) \in \mathbb{X}_f. \tag{S1d}$$

The core idea of MPC is based on the following three-step procedure:

1) At time step t, obtain/estimate state information $x(t)$.
2) Find an optimal solution $u^\star : \{0,\dots,N-1\} \to \mathbb{U}$, $x^\star : \{0,\dots,N-1\} \to \mathbb{X}$ to OCP (S1).
3) Apply a part of the optimal input sequence to the plant; go to step 1).

This procedure leads to the closed-loop dynamics whose time evolution usually differs from the predicted optimal state trajectories. Despite the multitude of MPC schemes and formulations discussed in the literature, there are several constitutive commonalities.

- MPC is based on the repeated solution of an OCP (S1) that involves a performance criterion (S1a). This solution can be determined online or offline (and exploited online based on some appropriate data structure).
- MPC involves some kind of model of the underlying process, which can be linear or nonlinear, finite dimensional or infinite dimensional, or parametric or nonparametric/data driven.
- MPC typically involves constraints on inputs and states or functions thereof; cf. (S1c).

Moreover, one can distinguish three main variants of MPC, which differ in how and when the next optimization is performed.

- *Receding-horizon schemes*, i.e., the first part of the optimal input sequence is applied, and then the OCP is resolved on a horizon of equal length [Figure S1(a)]. In case more than one time-step of the optimal input sequence is applied, one refers to multistep schemes.
- *Rolling-horizon schemes*, i.e., the entire predicted input trajectory, $u^\star : \{0,\dots, N-1\} \to \mathbb{U}$, is applied before a new optimization is performed [Figure S1(b)]. This is the most extreme variant of so-called multistep MPC schemes, and such concepts are frequently considered for planning and scheduling, for example, in power systems or production planning.
- *Shrinking-horizon schemes*, i.e., underlying dynamics, are considered on some finite time interval $[0, T_{sys}]$, and the horizon shrinks in each MPC iteration [Figure S1(c)]. That is, at $t = 0$ on predicts over $[0, T_{sys}]$, while at a later time, the shortened horizon $[t, T_{sys}]$, is considered. This idea has originated in the control of chemical batch processes and it simplifies the closed-loop analysis.

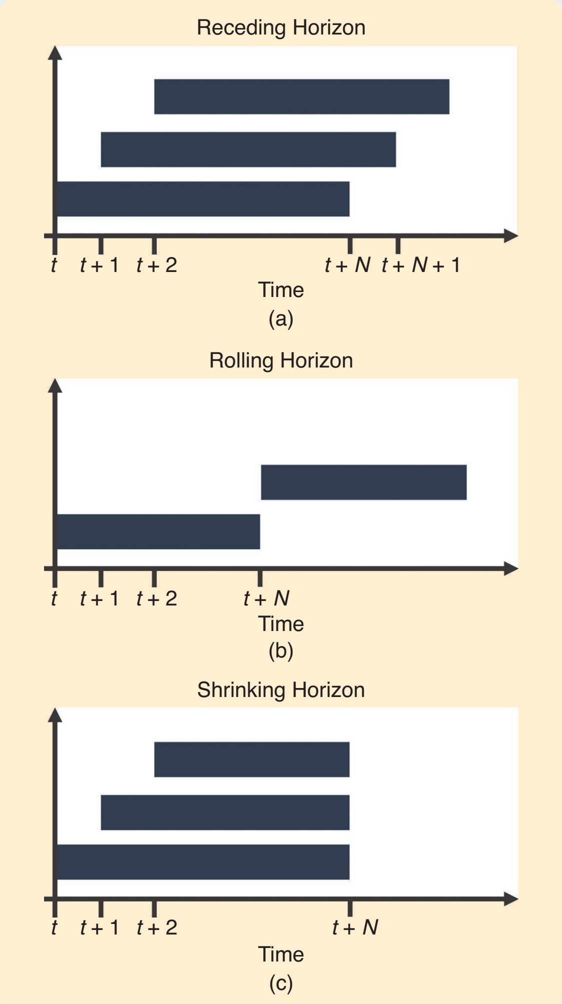

FIGURE S1 Horizon structures in MPC. (a) Receding horizon. (b) Rolling horizon. (c) Shrinking horizon.

Remarkably, the core idea of MPC has not changed much during the last decades. For instance, Stephen Boyd just recently summarized it in the podcast “inControl” as “You do this plan [that is, the prediction in MPC] so that you do something now that does not put you in a bad position in the future” [S1].

REFERENCE

[S1] A. Padoan and S. Boyd: Linear matrix inequalities, convex optimization, disciplined convex programming, rock & roll. inControl Podcast. Season 1, Ep. 10 [Online]. Available: www.incontrolpodcast.com

MPC Terminology—An Incomplete Glossary

Notion	Explanation
Cost function	Function $\ell : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ with $n = \dim x, m = \dim u$ penalizing control performance at each time step in OCP (S1); also known as *Lagrange term*, running cost, or stage cost in optimal control
Cost to go	Optimal performance on the infinite horizon $\Sigma_{\tau=0}^{\infty} \ell(x^\star(\tau), u^\star(\tau))$
Distributed MPC	A scheme wherein the OCP is solved in a distributed or decentralized fashion via tailored numerical algorithms
Economic MPC	A scheme with stage costs not related to the distance of the state to the target setpoint $\bar{x}$, that is, $\alpha(\| x - \bar{x} \|) \not\leq \ell(x,u),\ \alpha \in \mathcal{K}$; cf. stabilizing MPC
Explicit MPC	A scheme wherein the optimal feedback law is precomputed in analytical form offline, that is, prior to the application in a closed loop; also known as *offline MPC*
Feasibility	The existence of trajectories satisfying all constraints in OCP (S1)
Implicit MPC	The counterpart to explicit MPC, that is, optimal inputs are computed at the runtime of the controller; also denoted as *online MPC*
MPC	Model predictive control
Multistep MPC	A scheme wherein the optimal input signal $u^\star : \{0, \ldots, N-1\} \to \mathbb{U}$ is applied for more than one time step in a closed loop
Nonlinear MPC (NMPC)	MPC with nonlinear models or nonlinear constraints
OCP	Optimal control problem
Optimal value function	$V_N : x(k) \mapsto \Sigma_{\tau=0}^{N-1} \ell(x^\star(\tau), u^\star(\tau)) + V_f(x^\star(N))$; maps the initial condition $x(k)$ of OCP (S1) to the optimal performance
Prediction horizon	The time horizon $N \in \mathbb{N} \cup \{\infty\}$ considered in OCP (S1)
Recursive feasibility	A requirement that the feasibility of OCP (S1) at time step t implies its feasibility at time step $t+1$
Robust MPC	A scheme that considers set-bounded disturbances acting on the system (or model uncertainty) in the prediction
Stabilizing MPC	A basic scheme wherein the stage cost is lower-bounded by the distance to the target setpoint $\bar{x}$, that is, $\alpha(\| x - \bar{x} \|) \leq \ell(x,u)$ with α of class $\mathcal{K}$; also known as *tracking MPC*
Stochastic MPC	A scheme that considers stochastic disturbances acting on the system (or stochastic model uncertainty) in the prediction
Terminal control law	A control law valid on the terminal set $\mathbb{X}_f$, usually not implemented in a closed loop
Terminal constraints	Constraints (S1d) defined at the end of the prediction horizon; arise as inequalities and equalities
Terminal ingredients	An umbrella term for terminal constraints and terminal penalties
Terminal penalty	Function $V_f : \mathbb{R}^n \to \mathbb{R}$ penalizing $x(N)$ in (S1a), also known as *Mayer term* in optimal control
Terminal set	The set $\mathbb{X}_f \subseteq \mathbb{X} \subseteq \mathbb{R}^n$ in which the predicted trajectories have to be at $\tau = N$; cf. (S1d)
Turnpike property	A similarity property of parametric OCPs wherein optimal solutions for different initial conditions and horizons are structurally similar, frequently used in economic MPC

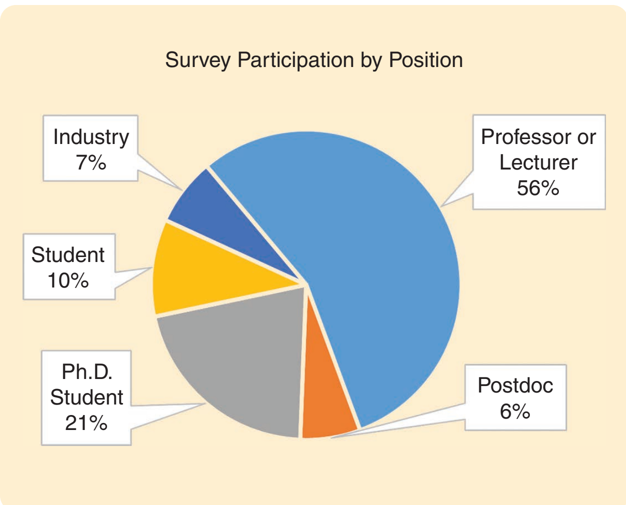

FIGURE 2 Survey participation by the position of the respondents.

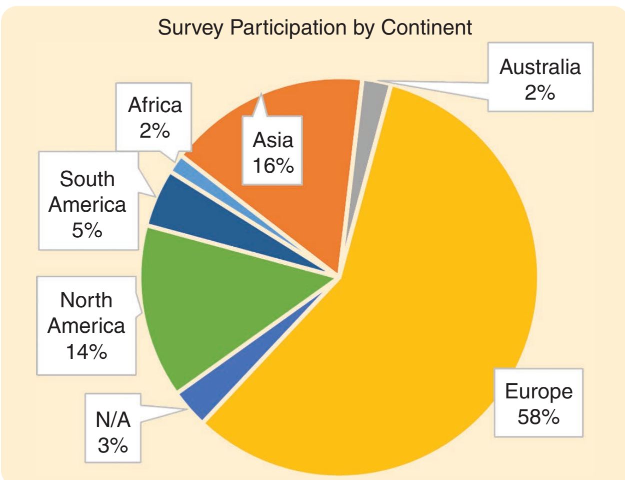

FIGURE 3 Survey participation by the continent the respondents are located on.

level are rare [5%; see Figure 5(a)], and MPC is treated only at the master's level most of the time (60%, same figure).

If MPC is treated at the bachelor's level, it appears in the third year in 93% of all cases and in the second year otherwise (data not shown in any figure). Remarkably, almost one-quarter of all participants (29 out of 128) responded that MPC is not taught at all in their departments [Figure 5(a)]. If, on the other hand, MPC is taught, it is treated in more than one course in 77% of all cases (82 of 99, not evident from any figure). Among the survey participants at departments offering MPC courses, 80% stated they are personally involved in teaching MPC.

Most first MPC courses are attended by 40 or fewer students (Figure 6). Somewhat surprisingly, large first MPC courses with more than 150 students exist (4% correspond to four of 99 answers that Figure 6 is based on).

More than 60% of the courses welcome students not only from the teaching department but also from other departments. If students from other departments are admitted, they are from the departments that already appeared under "other" in Figure 4 or from natural sciences such as physics. The answers to the question of which other departments offer MPC than the department of the respondent were consistent with both the results for the previous question and "others" in Figure 4.

CONTENTS OF EXISTING MPC COURSES

In the third part of the survey, we collected data on the layout and content of existing MPC courses. All results reported in this section are based on 99 answers, corresponding to the number of participants with MPC courses in their departments [all but "MPC not taught at all" in Figure 5(a)]. Because the questions in this part of the survey were not mandatory, the exact number of responses to the questions varies. All questions were answered by at least 93 participants if not noted otherwise.

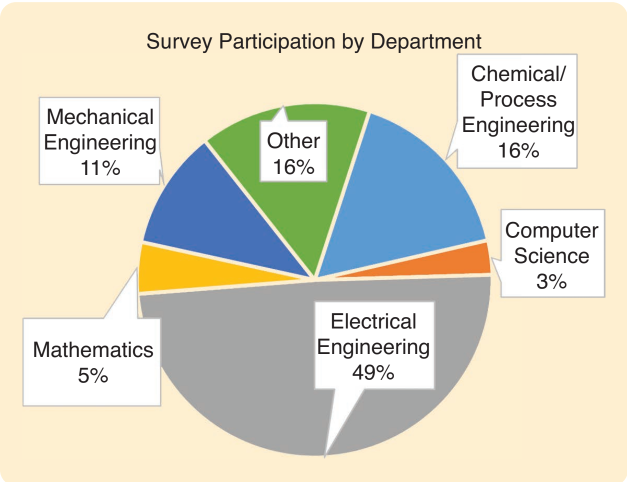

FIGURE 4 Survey participation by the department the respondents are affiliated with (see the "Insights From the Survey" section for a brief description of "other").

PREREQUISITES FOR ENROLLMENT AND MPC ENTRY POINTS

The majority of MPC courses assume knowledge in mathematics and linear control and systems theory (86%, 89%, and 77%, respectively; see Figure 7). Previous experience with modeling and identification is assumed less often (39% and 21%, respectively; see Figure 7). About one-half of our respondents require their students to know basic optimization concepts when entering the first MPC course (46%; see Figure 7).

MPC is introduced as an extension to the unconstrained linear-quadratic regulator (LQR) or as a linear-quadratic MPC with constraints (in 42% and 38% of all cases, respectively; see Figure 8). NMPC is introduced directly, that is, without treating linear MPC before, about half as frequently (17%; see Figure 8). Participants who selected "other" state that they use intuitive examples, dynamic matrix control, or generalized predictive control to introduce MPC. In the majority of cases, MPC is taught in discrete time (80%; see Figure 9). A few participants teach MPC in both discrete and continuous time (4%, category "other" in Figure 9).

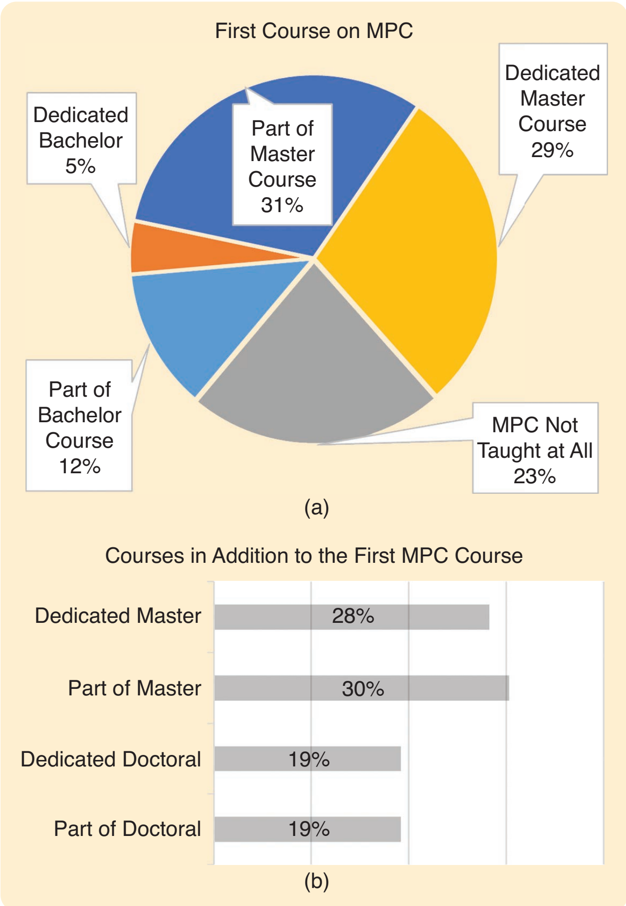

FIGURE 5 (a) The type of course that MPC first appears in. (b) Additional courses MPC appears in.

ADVANCED TOPICS, OPTIMIZATION, AND NUMERICS

Figure 10 lists the advanced topics covered in MPC courses at the respondents' institutions. We asked participants to list only topics that are covered for at least 60 min in this and the following questions. Participants who selected "other" mention set theoretic methods, reference governors, or economic real-time optimization, or state that no advanced topics are covered. It is evident from Figure 10 that NMPC is the most popular advanced topic (68%), followed by robust (49%) and explicit MPC (43%). Both learning-based and data-driven MPC have recently received increasing attention in MPC research, and we conjecture that these topics will also soon be more frequently treated in MPC courses than today (22%).

We also asked which topics related to optimization and optimal control are covered (Figure 11). Since 80% of all courses teach MPC in discrete time (see Figure 9), it is no surprise that Karush-Kuhn-Tucker (KKT) conditions appear much more frequently (in 77% of all courses; see Figure 11) than Hamilton-Jacobi-Bellman equations and Pontryagin's maximum principle. Dynamic programming is probably treated frequently (69%; see Figure 11) for the same reason.

About half of the courses do not cover the details of MPC implementations but use toolboxes and generic algorithms only (53%; see Figure 12). Advanced topics in numerics listed in Figure 12 are each treated in about 10% of the courses.

Our next seven questions inquired about which elements, such as pen-and-paper or programming exercises, MPC courses comprise today. Figure 13 reveals that almost 30% of these courses do not involve any pen-and-paper exercises. For about 60% of the courses, this format takes up to 25–75% of the allocated time (rows 2–5 in Figure 13), and less than 5% devote most of the time (that is, more than 75%) to pen-and-paper tasks.

We asked the corresponding question about programming exercises (see Figure 14). As expected, the results indicate that programming exercises are considered to be

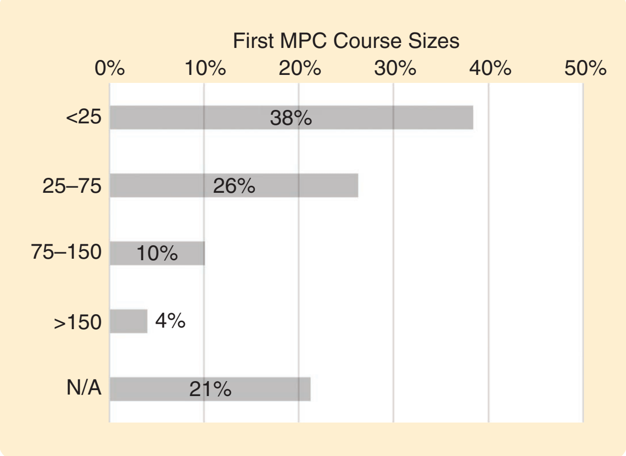

FIGURE 6 The number of students in the first MPC course.

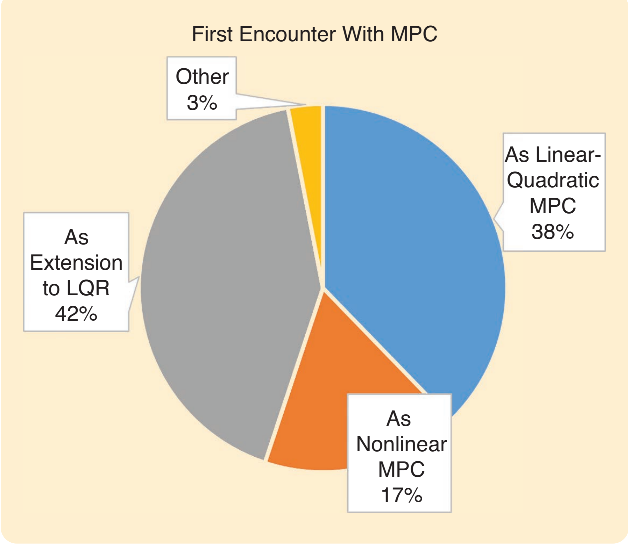

FIGURE 8 Approaches to introducing MPC as extensions to known control methods.

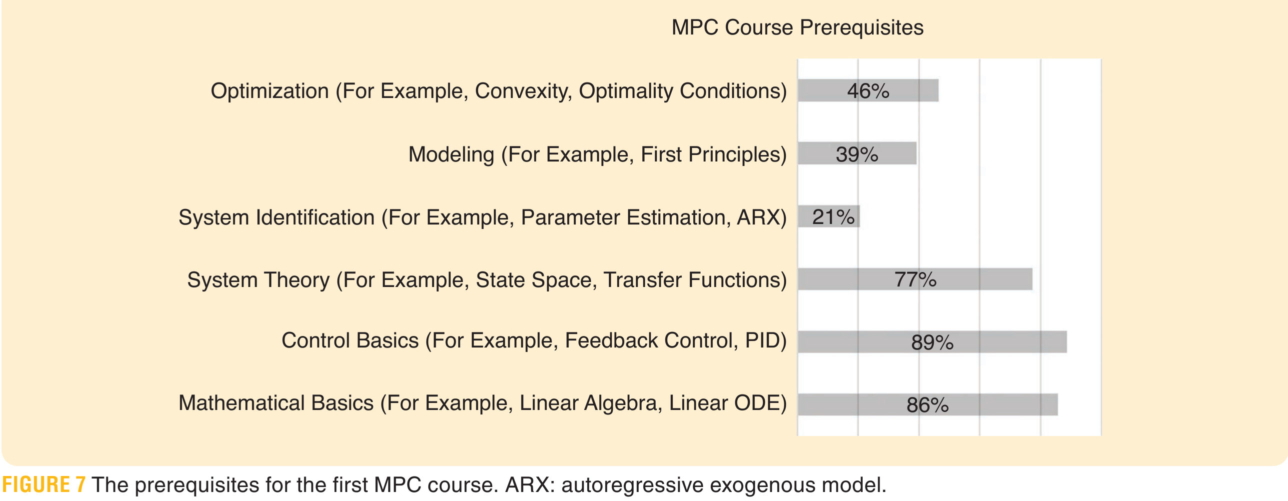

FIGURE 7 The prerequisites for the first MPC course. ARX: autoregressive exogenous model.

indispensable. Only about 6% of the courses covered here do not involve any programming exercises.

Programming exercises are most often carried out in MATLAB (see Figure 15). Python is the second most popular language. We note that the growing popularity of Python, which was recently boosted by the use of Python for data-driven methods, is likely to result in a larger share of Python-based MPC courses and control courses in general than evident from Figure 15.

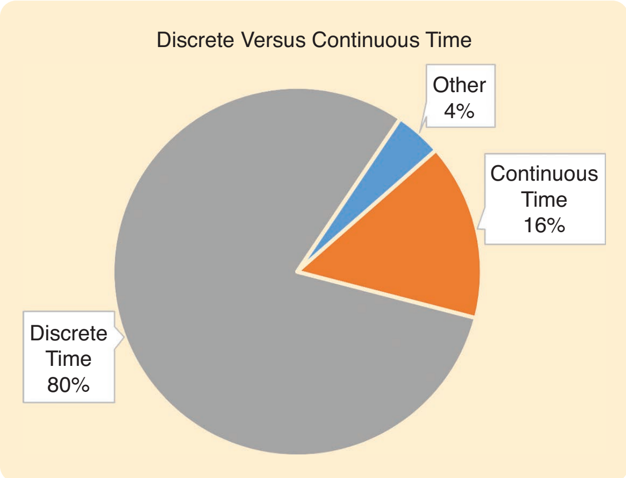

FIGURE 9 The fractions of the first MPC courses using discrete-time and continuous-time systems.

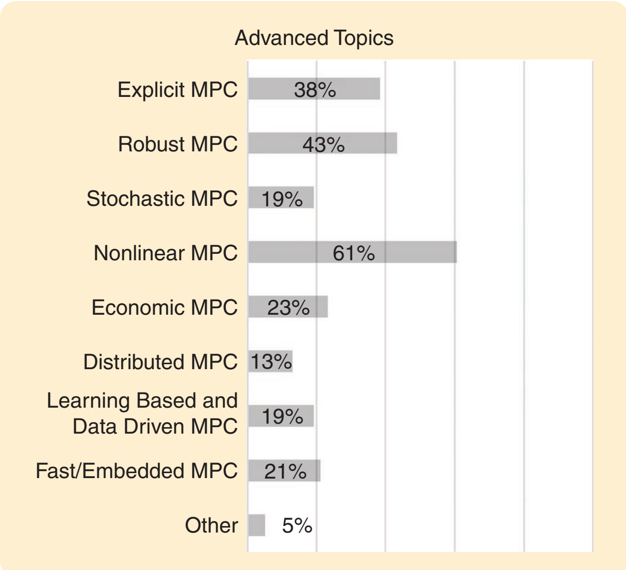

FIGURE 10 Advanced topics covered at the institutions of the participants.

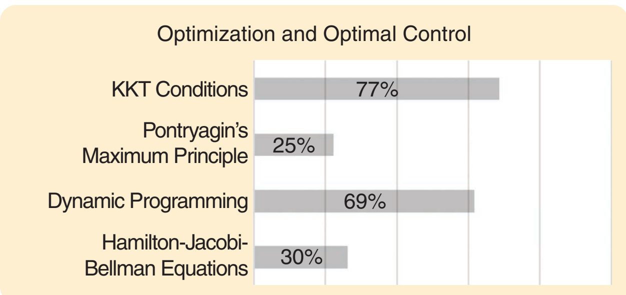

FIGURE 11 Topics related to optimization and optimal control covered in the existing MPC courses.

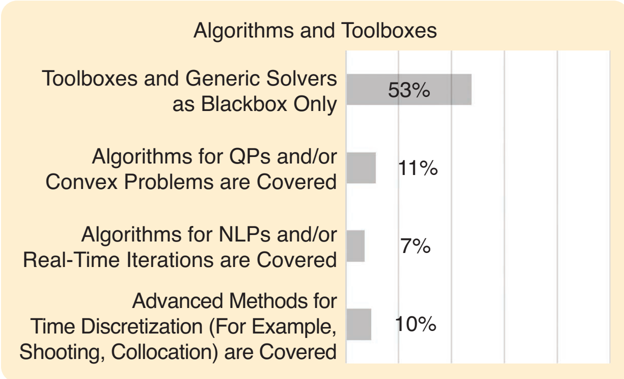

FIGURE 12 Advanced algorithms covered in the existing MPC courses assessed here.

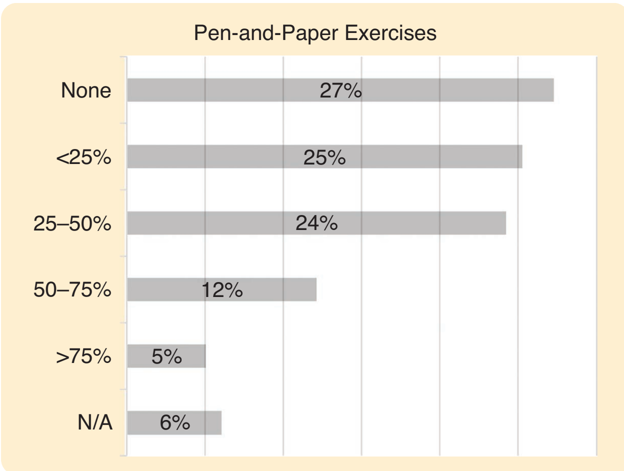

FIGURE 13 Time devoted to pen-and-paper exercises. The column labeled "50–75%," for example, means that about 12% of all MPC courses covered here spend 50–75% of their total exercise time on pen-and-paper exercises. N/A refers to empty answers.

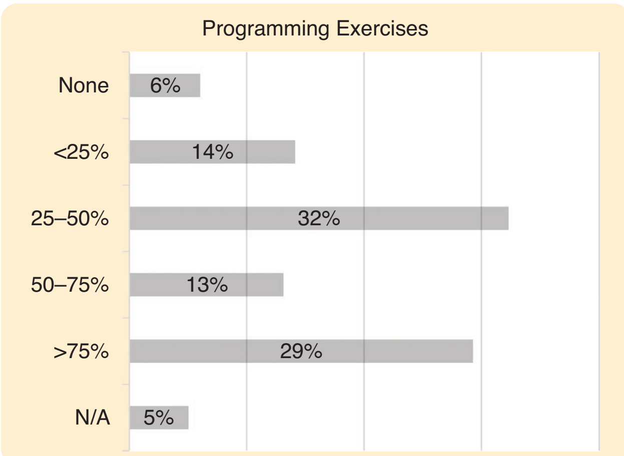

FIGURE 14 The time devoted to programming exercises (see Figure 13 for an explanation of axes labels).

The data in Figures 13 and 14 show that programming and numerical implementation are considered to be an intrinsically important aspect. Conversely, about 30% of the MPC courses do not encompass any pen-and-paper exercises at all.

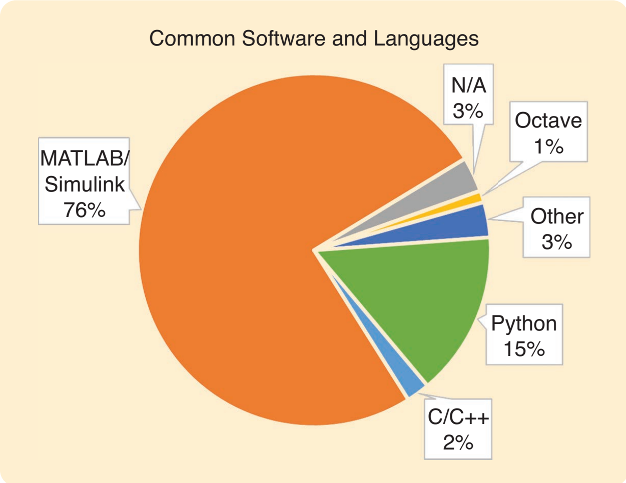

FIGURE 15 The software and languages used for programming exercises.

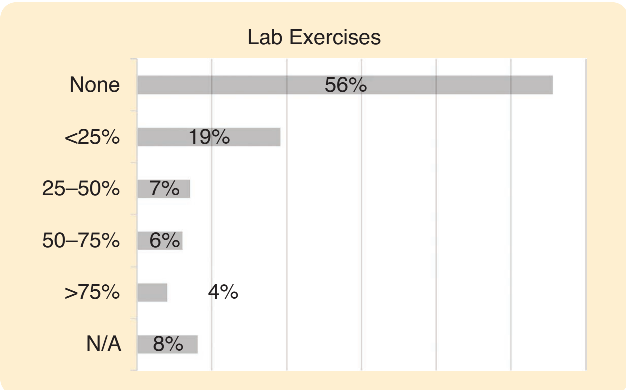

FIGURE 16 The time devoted to lab exercises (see Figure 14 for an explanation of axes labels).

TABLE 1 MPC Labs.

Lab	Times Mentioned
Level control for a multitank system	6
Tethered multirotor	3
Gantry crane	2
Inverted pendulum	2
Multicopter flight control	2
Power electronics and motor control	2
Hydrogen storage and PEM cell	1
Airflow and temperature control in a wind tunnel	1

About 37% of the MPC courses covered here involve labs (Figure 16), where we pointed out that an exercise has to involve a practical implementation (that is, go beyond mere programming) to qualify as a lab exercise. It is evident from Figure 16 that a few courses even strongly focus on lab exercises.

We asked for a brief description of the labs and summarized those that go beyond programming exercises (19 answers). The results are summarized in Table 1.

Industrial applications of MPC are treated in 69% of the courses covered by the survey, while 31% treat only academic examples (Figure 17). About 14% of the courses involve an industrial practitioner giving a firsthand account of an industrial application. In the remaining cases, these examples are based on research projects of the lecturer (32%) or industrial examples reported in the literature (38%).

TEXTBOOKS USED FOR MPC TEACHING

We collected 35 answers to our questions about textbooks that are used to accompany MPC courses. Six books [5], [17], [18], [19], [20], [21] were mentioned more than once in these 35 answers. Among the 35 answers, 11 listed more than one book or listed a single book and pointed out that it is used only as a reference, but the course does not follow this book. Overall, these answers indicate that most existing MPC courses are not based on textbooks but on tailored course material. We collected only five and seven answers

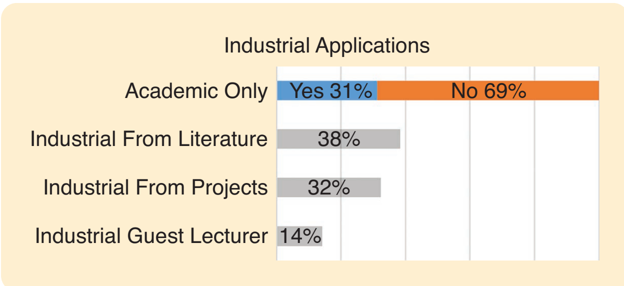

FIGURE 17 Industrial applications treated in the MPC courses.

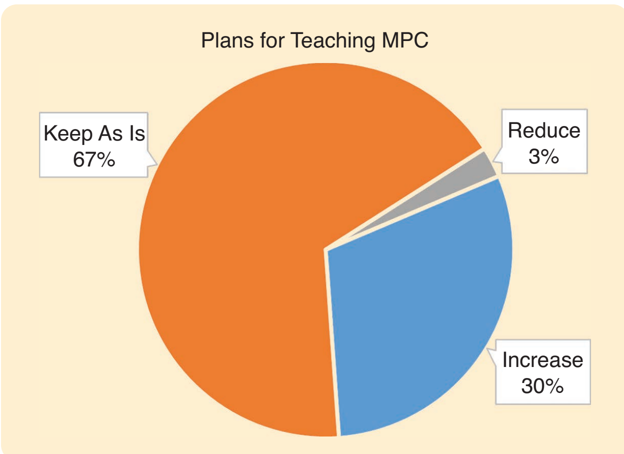

FIGURE 18 The anticipated future development of MPC courses.

to the questions about other publicly available written course material or videos, respectively. Because the answers to these two questions did not reveal any pattern, they are not treated here in more detail.

Finally, we asked our participants whether MPC is likely to become a more or less important subject in teaching. Evidently, MPC is believed to be covered sufficiently well today (67%; see Figure 18), or MPC teaching is to be expanded in the future (30%). Only 3% of our participants intend to reduce MPC teaching in the future.

LESSONS LEARNED FROM MPC COURSES: WHAT HAS WORKED WELL?

Our next question ("What has worked well in your MPC courses?") called for written answers. We collected 46 responses, ranging from short but concise comments to explanations spanning a paragraph. We summarize the results in the four following sections. Many of the 46 responses addressed more than one of these sections. One of the answers conjectured that combining MPC and reinforcement learning (RL) helps to increase enrollment because ML-related topics attract students.

MOTIVATING EXAMPLES AND CASE STUDIES

Many responses stress the importance of case studies or, more generally, good motivational examples for MPC (mentioned 14 times). Here, "example" and "case study" refer to simulations and results thereof used in lectures. Both simulations carried out by the students and MPC implementations for real (that is, not simulated) systems are treated further later. Case studies and examples from a wide variety of application domains are simulated, ranging from

MPC for Undergraduates—From Discrete-Time LQR to Basic MPC With Some Advanced Flavors

As already noted, classical linear-quadratic MPC requires only basic knowledge of systems theory, linear algebra, and calculus. More specifically, students should be familiar with linear state space models of the form

$$x(k+1) = Ax(k) + Bu(k), \qquad x(0) \doteq x_0 \tag{S2}$$

or their continuous-time counterparts. In this context, discrete-time models as in (S2) facilitate the MPC design, and they consequently seem most suitable for an undergraduate course. However, continuous-time models usually connect better to real-world problems, and it thus makes sense to either teach or recall zero-order-hold discretizations. Before addressing the actual MPC problem, a few basics on optimal control are helpful. A suitable starting point is the linear-quadratic regulator (LQR). In fact, the OCP

$$\min_{\substack{x(0),x(1),\ldots \\ u(0),u(1),\ldots}} \sum_{k=0}^{\infty} x^\top(k)Qx(k) + u^\top(k)Ru(k) \tag{S3}$$

which is subject to the dynamics (S2), already introduces some ingredients for MPC without including more challenging state or input constraints. For instance, (S3) already involves the quadratic stage cost $\ell(x,u) \doteq x^\top Qx + u^\top Ru$, and one could motivate this fact by either analyzing energy levels of illustrative systems or investigating weighted distances to the model's equilibrium point at the origin. While understanding the idea behind (S3) is simple, its solution via, for example, Bellman's principle of optimality, is not straightforward within an undergraduate course. However, especially when aiming for MPC, the exact solution of (S3) can be prepared with the finite-dimensional OCP

$$\min_{\substack{x(0),\ldots,x(N), \\ u(0),\ldots,u(N-1)}} x^\top(t_N)Sx(t_N) + \sum_{k=0}^{N-1} \ell(x(t_k),u(t_k)) \tag{S4}$$

where $N \in \mathbb{N}$ refers to the prediction horizon. At this point, various motivations for the quadratic approximation of the cost-to-go can be given depending on previous knowledge of the students. Then, (S4) can be easily condensed to the unconstrained quadratic program (QP)

$$U^* \doteq \operatorname*{argmin}_U \frac{1}{2} U^\top HU + x_0^\top F^\top U \tag{S5}$$

where U denotes the stacked inputs $u(0),\ldots,u(N-1)$ and where the (quadratic) term in x_0 has been omitted for brevity. Since constraints are not yet involved, the optimizer

$$U^* = -H^{-1}Fx_0 \tag{S6}$$

can easily be computed by setting the gradient of the cost function in (S5) equal to zero. This result reveals many interesting features of the approximated LQR problem (S4). First, it is easy to motivate that H needs to be positive definite because this implies the existence of H^{-1} and uniqueness of the minimum. Second, (S6) already indicates the state feedback nature of LQR. Third, reinserting (S6) in (S5) reveals that the optimal value function is quadratic in x_0. Postulating that the latter observation remains valid for $N \to \infty$, we can eventually solve (S3). In fact, setting the optimal value function of (S3) equal to $x_0^\top Px_0$ and investigating the corresponding Bellman equation finally leads to the discrete-time algebraic Riccati equation

$$P = A^\top(P - PB(R + B^\top PB)^{-1}B^\top P)A + Q$$

which allows one to compute P and the associated LQR with the controller gain $K^* \doteq -(R + B^\top PB)^{-1}B^\top PA$. In addition, P enables one to make the OCPs (S3) and (S4) equal by setting chemical reactors and power systems to mechatronic and robotic systems such as robotic arms and to drones and autonomous vehicles. Two answers point out that lectures on industrial applications by practitioners helped to attract and motivate students.

PROGRAMMING EXERCISES AND LABS

The importance and benefit of programming exercises and labs are mentioned in 16 responses. Some answers mention the use of collaborative programming tools for this purpose (both with MATLAB and Python). The implemented methods range from linear-quadratic MPC without terminal constraints and cost to economic, robust, and stochastic MPC. The simulation examples treated span the same range as the motivating examples and case studies listed previously.

MPC IMPLEMENTATIONS FOR REAL SYSTEMS

Finally, exercises that involve MPC implementation for real systems are reported by five participants. Systems treated in labs are discussed previously and listed in Table 1.

GOOD TEACHING PRACTICES IN GENERAL

Many answers to the question "What has worked well in teaching MPC?" relate to good teaching practices in general (mentioned 16 times), such as devoting sufficient time to exercises to allow for "learning by doing" or, more generally, a good balance of lectures on the one hand and exercises or labs on the other hand. Several respondents point out the importance of a well-organized course layout and high-quality course material. While this may apply to any course, some of the answers indicate a particular importance for MPC courses. According to our respondents, this is the case, for example, because knowledge

$S \doteq P$. This observation can be picked up later when stability guarantees for MPC are addressed.

Based on the aforementioned insights, the way to linear-quadratic MPC is not far. In fact, it basically builds on the OCP (S4) combined with set-based state and input constraints of the form

$$x(k) \in \mathbb{X} \quad \text{and} \quad u(k) \in \mathbb{U}.$$

In this context, it is important to stress that constraints, especially for inputs, are omnipresent in control tasks. Thinking about valves and temperature limits, box constraints can provide a first concretization of the sets $\mathbb{X} \subset \mathbb{R}^n$ and $\mathbb{U} \subset \mathbb{R}^m$. A simple task for the students could then consist of rewriting such sets in terms of affine inequalities, that is, finding a hyperplane description of $\mathbb{X}$ and $\mathbb{U}$. It is then easy to see that the condensing that allowed one to transform (S4) into (S5) can also be applied to condense all state and input constraints to

$$GU \leq Ex_0 + c. \tag{S7}$$

One can now point out that (S5) subject to (S7) is a standard QP that can be efficiently solved using off-the-shelf solvers such as, for example, `quadprog` in MATLAB (cf. Figure 15). Hence, at this point, the students are ready for their first numerical experiments with MPC. This could be based on the popular double-integrator toy example, but slightly more complex multiple input/multiple output systems might be more elucidating with respect to the capabilities of MPC. Yet, more important than the sample system is the clarification of how the solution of the OCP is turned into a model predictive *feedback* controller. Typically, this is realized by picking the first input $u^*(0)$ from the optimal input sequence U^*, applying it to the system, and repeating the procedure at the next sampling instant for the subsequent system state (which, of course, has to be measured or estimated/observed in real-world applications).

Once the students are familiar with the closed-loop realization of MPC and its behavior, various options for closer investigations exist. For instance, one can make an excursion to numerical solutions of QPs (or, more generally, convex optimization), although this does not seem to be very popular according to Figure 12. Still, a brief introduction to active-set solvers can open the door to advanced topics like explicit MPC [S2] or warm starting. From a control perspective, it may, however, be more interesting to further investigate which guarantees MPC can provide. In this context, an important point is to clarify the crucial difference between optimality and stability [S3]. Once this has been established, explaining methods to guarantee stability and recursive feasibility based on "safe" terminal sets, for example, sets on which LQR does not violate any constraints, is straightforward. The students should now also be ready to understand the basic concepts and challenges of advanced topics like robust, stochastic, nonlinear, or data-driven MPC, and a corresponding outlook can serve as a bridge to advanced/graduate MPC courses.

REFERENCES

[S2] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002, doi: 10.1016/S0005-1098(01)00174-1.
[S3] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000, doi: 10.1016/S0005-1098(99)00214-9.

from mathematics and systems and control courses has to be refreshed without losing students due to a too lengthy discussion of preliminaries. Other responses point out that it is not trivial to balance theory and the practical aspects of MPC.

Other good practices listed by our participants are short oral presentations by students (on an advanced MPC concept, for example), question and answer sessions, intermediate quizzes, inverted classroom teaching where videos are provided to study the material beforehand, and practical individual or team projects (programming MPC both for simulated systems and for real systems in labs).

LESSONS LEARNED FROM MPC COURSES: WHAT HAS NOT WORKED WELL?

Just as we inquired about best practices in MPC teaching, we asked about elements that did not work well. We collected 44 answers, which are summarized in the following four sections. Again, note that some of the 44 responses fit more than one of these sections.

THEORY IS OFTEN NOT APPRECIATED

Most of our participants (20 of 44) state that it is difficult to teach theory and to motivate students to work on theoretical problems themselves. The answers range from neutral comments that point out that it is challenging to teach theory simply because the material is advanced to those that explicitly state that students do not appreciate theory. Some of our respondents point out that this is in part due to a lack of sufficient training in linear algebra and calculus, where the use of even basic concepts, such as gradients and Hessians, is reported to be discouraging to students. These statements obviously reflect the importance of mathematical basics in the prerequisites for a first MPC course (see Figure 7). Unfortunately, none of our participants has a remedy for this problem. Some respondents confirm explicitly that, even if it is less popular with many students, theory must be addressed in an MPC course at least to discuss basic but crucial aspects such as stability.

COMPLEXITY OF THE TOPIC IN GENERAL

If problems arise in MPC courses, they often appear to be simply due to the complexity of the material in general or due to the difficulty of balancing prerequisites from several topics that are demanding by themselves. Out of the 44 free-text answers we collected, 14 point in this direction. Some responses indicate that students are overwhelmed by, or hesitant to, study nonlinear, robust, or stochastic MPC even if they very much appreciate linear MPC and are confident to use it. Our participants again credit this to the fact that advanced MPC topics come with prerequisites in additional mathematical topics, such as nonlinear systems theory, random processes, and probability theory.

DIFFICULTIES IN RECONCILING PREREQUISITES

Several answers state that problems arise because the MPC course has to spend too much time on prerequisites. Specifically, programming experience and background in optimization are listed (five and seven times, respectively). One participant states the same problem arises for integration schemes, which are required for NMPC. One response points out that it is appropriate but difficult to integrate more ML topics into existing MPC courses because it is impossible to cover ML basics in the MPC course in addition to the many other advanced concepts. All these answers again stress the importance of a thorough treatment of prerequisites before entering an MPC course. Interestingly, one respondent points out that a separate course on optimization has not served this purpose, but optimization must be integrated into the MPC course in their opinion.

OTHER ASPECTS

Three participants pointed out that students lose interest in MPC if only academic examples are treated. This is obviously in line with the importance of application examples reported in the previous section. One respondent pointed out that it is difficult to set up exams for an MPC course, and a classical written exam is not appropriate. Two answers independently report that students struggle with understanding the difference between open-loop and closed-loop control.

BEST PRACTICES FOR UNDERGRADUATE MPC

Remarkably, among the participants of our survey, 17% already teach MPC in bachelor courses. This observation is in line with the authors' view that MPC, while an advanced control scheme, is an exciting and inspiring topic for undergraduate teaching. In fact, basic knowledge about systems and control, linear algebra, and calculus are sufficient for introducing MPC as also confirmed by the most popular prerequisites for the first MPC course in Figure 7. Building only on these basic prerequisites also avoids the issues regarding missing programming experience or background in optimization (discussed in the "Difficulties in Reconciling Prerequisites" section previously).

We next discuss various approaches for including MPC in undergraduate courses—both in terms of a dedicated course and as part of more general control courses. Both variants also appear in the survey results, where the share of dedicated courses (with respect to all bachelor courses involving MPC) is between one-quarter and one-third (5% out of 17%). From Figures 8 and 9, it is further apparent that the first encounter with MPC is usually provided in the discrete-time domain (80%) and as an extension to (unconstrained) LQR (42%). Remarkably, a direct introduction of MPC schemes (without LQR as a precursor) also seems popular, where linear-quadratic MPC (38%) serves as the entry point roughly twice as often as NMPC (17%). Now, picking up the—according to our survey—most popular first MPC encounters, we briefly sketch a related course design in "MPC for Undergraduates—From Discrete-Time LQR to Basic MPC With Some Advanced Flavors" to provide an orientation for an MPC integration in the bachelor curriculum and, according to the survey results, for easy access to MPC.

Apart from this "mainstream" approach, various variants or completely different approaches can make sense depending on the associated department(s), course(s) of studies, or preliminary knowledge of students or lecturers. For instance, one can put more emphasis on the origin of prediction models and hence on, for example, first-principle modeling or system identification. Illustrating the corresponding methods with real-world problems will often result in nonlinear models. It may then be natural to directly introduce nonconvex OCPs, potentially followed by discretizations, leading to nonlinear programs (NLPs). QPs can still be of interest but now reflect only a special case. Such a course setup can provide an ideal basis for an advanced course on data-driven or learning-based predictive control.

Remarkably, while conceptually quite different from the previously discussed course in "MPC for Undergraduates—From Discrete-Time LQR to Basic MPC With Some Advanced Flavors," both courses share similar modules. In fact, differences mostly result from the different orders and levels of detail of the involved modules. This observation is illustrated in Figure 19, where the two discussed courses are sketched in the setups (a) and (b). Clearly, various additional course setups can be represented in a similar way. In

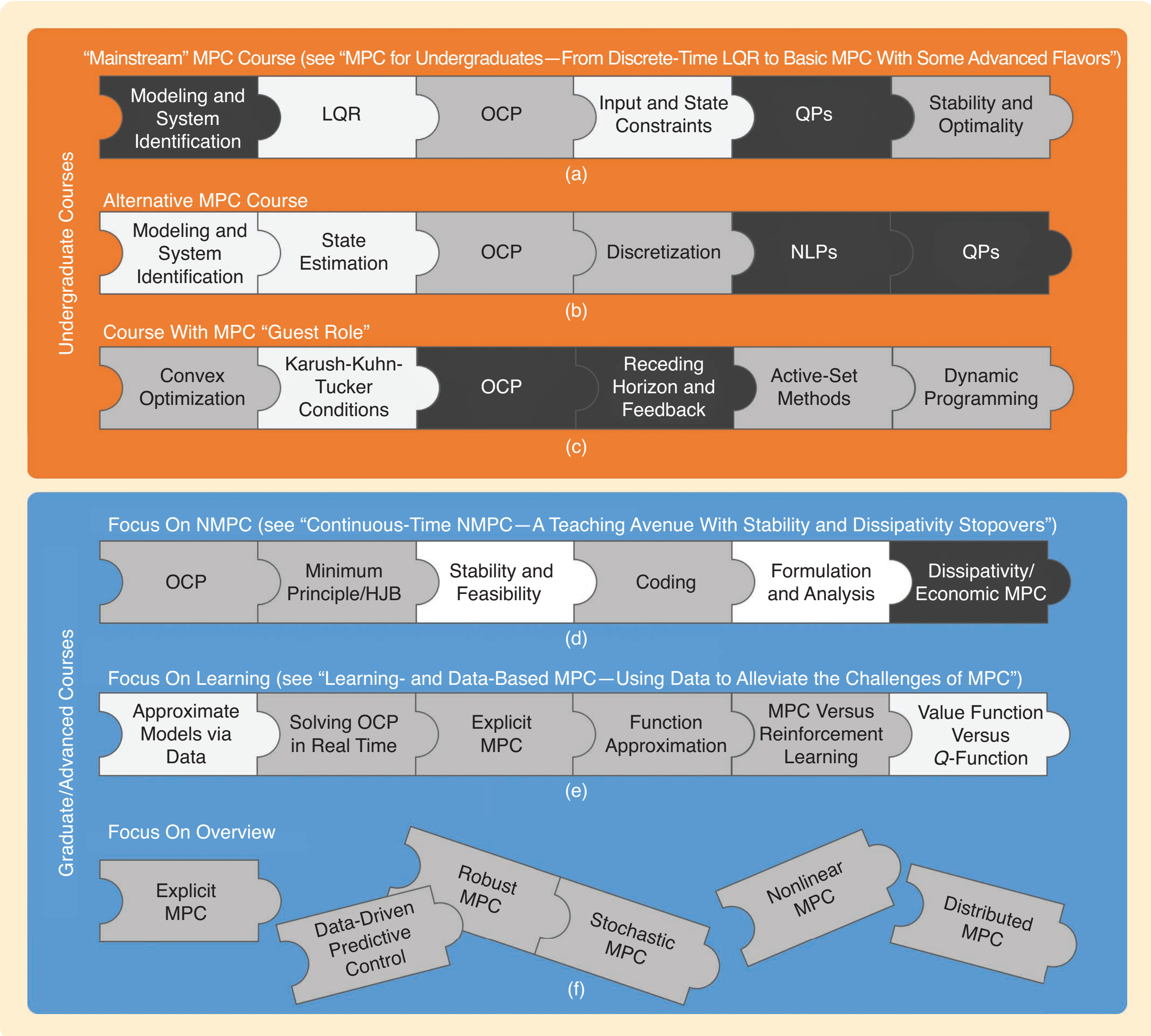

FIGURE 19 Possible setups for (a)–(c) undergraduate (d)–(f) graduate MPC courses. The puzzle pieces reflect a selection of possible modules. Gray shades illustrate the level of detail with which the individual modules may be taught, whereas light gray refers to in-depth coverage. Variants may result from reshuffling modules, adding new ones, or changing the level of detail as indicated in (f). Note that the illustration is simplified in many regards. For instance, in contrast to the identical shapes of the shown puzzle pieces, some modules (such as robust and stochastic MPC) might connect better than others.

particular, MPC could also be part of a control course with a broader scope, such as, for example, a robust control course. Moreover, MPC could also take the role of a powerful and illustrative application in a general (that is, not control-specific) course on (convex) optimization (see Figure 19(c)] or RL.

ADVANCED TOPICS FOR GRADUATE MPC COURSES

Undoubtedly, the contemporary success of MPC is deeply rooted in optimal control. A core element of advanced topics is the transition from the linear-quadratic setting to nonlinear dynamics. NMPC is indeed the most widely covered advanced topic according to the results of the survey (see Figure 10). There are different avenues to covering NMPC, and the choice of a discrete-time or a continuous-time setting is one of the main differences. In "Continuous-Time NMPC—A Teaching Avenue With Stability and Dissipativity Stopovers," we provide a detailed outline of how an advanced course on NMPC in

Continuous-Time NMPC—A Teaching Avenue With Stability and Dissipativity Stopovers

A compelling motivation for continuous-time NMPC approaches can be based on a prior introduction of classic optimal control. For starters, suppose that students have a basic understanding of OCPs

$$V_T(x_0) \doteq \min_{u(\cdot)} \int_{t_k}^{t_k+T} \ell(x(\tau), u(\tau))\,\mathrm{d}\tau + V_f(x(t_k+T)) \tag{S8a}$$

$$\text{subject to} \quad \forall \tau \in [t_k, t_k+T]$$

$$\dot{x}(\tau) = f(x(\tau), u(\tau)), \quad x(t_k) = x_k \tag{S8b}$$

$$u(\tau) \in \mathbb{U} \tag{S8c}$$

$$x(t_k+T) \in \mathbb{X}_f \tag{S8d}$$

and of the importance of the optimal value function $V_T : \mathbb{R}^n \to \mathbb{R}$. A crucial insight is the evident mismatch between the desire for (optimal) feedback laws and the immense difficulties of obtaining them in analytic form. Indeed, this intuitive idea can be traced back to the 1960s; see [1] and [2]. While from a mathematical point of view, sufficient background knowledge of the minimum principle and/or Hamilton-Jacobi-Bellman approaches is clearly beneficial for understanding NMPC, it may not be strictly necessary to dive deeply into it; cf. the empirical findings in Figure 11. For discrete-time and continuous-time approaches alike—and depending on the time allowance—one surely has to balance stability/feasibility analysis and numerical aspects with core design choices made when formulating the OCP.

STABILITY AND FEASIBILITY

For teaching purposes in general engineering programs, the specific details of different proof techniques for stability and feasibility analysis are second to an imperative observation; stability and optimality are, in general and in particular in MPC, disjoint categories. This is easily demonstrated by analytic and/or numerical examples. Moreover, the classic quote of Rudolf E. Kalman [S4] that "it is often assumed (tacitly and incorrectly) that a system with an optimal control law is necessarily stable" is still relevant. This observation can easily serve as a motivation for in-depth stability analysis and comparison of different proof techniques. This gist can also be derived by means of numerical case studies. As a starting point of formal stability analysis, the pivotal (control) Lyapunov inequality

$$\nabla V_f^\top f(x,u) + \ell(x,u) \le 0 \tag{S9}$$

which has to hold for all $x \in \mathbb{X}_f$ and some terminal feedback law $u = K_f(x)$, is of merit. The actual essence of (S9) is not so much its specific use in stability proofs but rather that it implies the inequality

$$V_f(x(t)) \ge V_\infty(x(t)) \doteq \int_t^\infty \ell(x^*(\tau), u^*(\tau))\,\mathrm{d}\tau \tag{S10}$$

which can be paraphrased as *terminal penalties bound the cost to go $V_\infty(x(t))$ from above.*

Similar to stability and feasibility analysis, the discussion surrounding numerical solution methods for continuous-time NMPC formulations leaves substantial freedom to address specific needs. In an applications-driven setting or if the time allowance is an active scheduling constraint, one can rely on a growing number of mature open source toolboxes that are available for typical coding environments, such as MATLAB, Python, or C/C++. We refer to [10] for entry points to the literature on open source and commercial software tools.

However, the prospect of learning outcomes attained by diving deeper into numerical methods is substantial. Primarily, these outcomes are coding and debugging competencies and the core engineering know-how of making advanced control schemes work. Beyond this, the deep interplay of the actual OCP formulation, the formal stability/feasibility properties, and the numerical solution are the most conceptually challenging aspects of teaching and of learning NMPC.

THE INTERPLAY OF OCP FORMULATION AND ANALYSIS

At large, OCP formulation refers to several design choices to be made.

1) How does one choose the cost function ℓ? What are the good reasons to go for convex-quadratic structures?
2) Shall one consider terminal ingredients (that is, terminal penalties and constraints) in the OCP formulation? If yes, in the form of strict constraints or via suitable relaxations?

continuous-time could look like, including other advanced concepts, such as dissipativity.

After NMPC, many advanced topics, such as explicit MPC, robust MPC, stochastic MPC, and economic MPC, are already being covered by a significant number of existing courses (see Figure 10). While the topic of learning-based and data-based MPC is currently covered in only 19% of the courses, we believe that this number will significantly increase in the coming years. We expect this to happen not just because ML is a popular topic but also because some fields, such as RL, are very closely related, and both fields can profit from each other. Because of that reason, we outline in "Learning- and Data-Based MPC—Using Data to Alleviate the Challenges of MPC" the structure and topics that a possible graduate MPC course on this topic could have. The central idea is to use the increasing availability of data to alleviate some central challenges of MPC in modeling, the online solution of OCPs, and the solution of infinite horizon problems in the presence of stochastic uncertainties.

3) How to address the evident mismatch between the often demanding and hard-to-check assumptions for stability guarantees and the need for cost-effective and time-constrained MPC design in applications?

1) Implicitly, our aforementioned considerations have assumed that setpoint stabilization is the considered control task at hand. This is usually reflected in the core assumption

$$\ell(x,u) \geq \alpha(\|x-\bar{x}\|), \quad \ell(\bar{x},\bar{u}) = 0, f(\bar{x},\bar{u}) = 0 \tag{S11}$$

whereby $(\bar{x},\bar{u}) \in \mathbb{R}^n \times \mathbb{R}^m$ is the target setpoint and corresponding input, and α is a suitable class $\mathcal{K}$ function. However, the observation that in many applications setpoint stabilization is more a means to an end, instead of the actual operational requirement, highlights the learning potential surrounding the choice of ℓ. Indeed, the mismatch between the optimal control perspective that stage costs ℓ strictly convex in u—for example, to preclude singular arcs (glossing over the active state constraints inducing these)—and frequently arising economic costs, which often turn out to be affine in u, are a good starting point for deeper exploration. In this context, one can also discuss the positive effects of adding small convex-quadratic regularizations to the objective.

2) Zooming in on the second item, that is, the role of terminal ingredients, a simulation-driven approach would compare solution times with and without terminal constraints. A theory-driven teaching concept can draw upon the classic notion of normality/abnormality of OCPs. That is, recalling that OCP analysis via the minimum principle relies heavily on the Hamiltonian

$$H(\lambda_0,\lambda,x,u) = \lambda_0\ell(x,u) + \lambda^\top f(x,u) \tag{S12}$$

and that it becomes very cumbersome if the constant adjoint variable λ_0 associated with the cost function is zero, one can easily motivate why one would like to work without strict terminal constraints. The core observation is that the absence of terminal constraints (and supposing that no state constraint is active at the final time) in OCP (S8) implies normality, that is, $\lambda_0 \neq 0$ holds for the optimal solution of (S8).

3) Turning toward the gap between stability analysis and engineering practice, arguably the application-relevant gist is that, supposing suitable controllability properties, sufficiently long horizons imply recursive feasibility and stability.

The aforementioned aspects 1–3 can be discussed in an analytical approach via the proofs of formal results, in terms of interactive and flipped-classroom discussion elements, and/or via numerical experiments.

Returning to the issue of OCP design for NMPC, control problems beyond pure setpoint stabilization—such as trajectory tracking, path following, or disturbance attenuation—are additional aspects that can easily be motivated via applications. For example, path-following formulations are very common in mechatronics and robotics. The main commonality of these problems is that—upon the introduction of suitable error coordinates—they are closely related to setpoint stabilization problems in terms of their structure.

DISSIPATIVITY AND ECONOMIC FORMULATIONS

The issues surrounding the choice of optimization objectives reflecting application requirements provide an elegant transition to economic NMPC formulations. A theory-driven approach can leverage dissipativity concepts for OCPs and related proof techniques. The dissipativity route to NMPC and optimal control relies on inequalities of the following type:

$$\nabla S^\top f(x,u) \leq \underbrace{-\alpha(\|x-\bar{x}\|) + \ell(x,u) - \ell(\bar{x},\bar{u})}_{\omega(x,u)} \tag{S13}$$

where $S:\mathbb{R}^n \to \mathbb{R}$ is a storage function bounded from below, and $\omega(x,u)$ is a supply rate, which bounds the change of storage from above. A comprehensive overview of discrete-time results on economic NMPC is given in [S5].

This inequality has far-reaching implications for optimal control. For example, it is easily seen that

$$S(x^\star(T)) - S(x_0) \leq V_T(x_0) \tag{S14}$$

which, upon assuming that $\lim_{T\to\infty} S(x^\star(T)) = 0$, gives $-S(x_0) - V_\infty(x_0) \leq 0$. This can be regarded as the counterpart to (S10). Moreover, the differential perspective yields

$$\nabla S^\top f(x,u) - \ell(x,u) \leq 0 \tag{S15}$$

(*Continued*)

Continuous-Time NMPC—A Teaching Avenue With Stability and Dissipativity Stopovers (*Continued*)

which is closely linked to the usual stability condition (S9) via

$$\nabla V_f^\top f(x,u) \leq -\ell(x,u) \leq -\nabla S^\top f(x,u).$$

An optimal control-driven approach toward economic NMPC formulations can intuitively address turnpike phenomena, that is, the observation that optimal solutions of dissipative OCPs tend to approach the optimal steady state; cf. Figure S2.

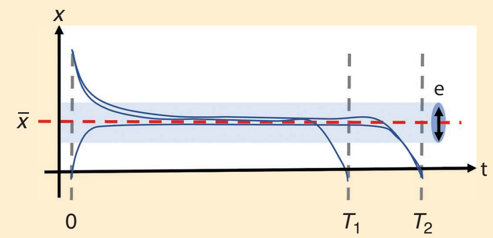

FIGURE S2 A sketch of the turnpike phenomenon.

While most of the aforementioned considerations can be transferred between continuous-time and discrete-time approaches, the former one shines when it comes to the consideration of models. Specifically, keeping ordinary differential equation (ODE) models derived from the first principles in the picture is beneficial when discussing the OCP design as such. Most engineering students in mechanical, aeronautical, and chemical engineering are accustomed to ODE models derived via the Euler-Lagrange formalism or via mass and energy balances. For

Learning- and Data-Based MPC—Using Data to Alleviate the Challenges of MPC

With the increasing availability of data in technical systems and the developments in related disciplines such as ML, the incorporation of learning- and data-based MPC concepts in teaching will increase rapidly in the near future. A data-based and learning-based MPC course is suited for an advanced audience that already has the basic knowledge of MPC and is familiar with a typical MPC problem formulation, for example, in discrete time as the one presented in (S1). Starting from this basic formulation, a powerful motivation for data- and learning-based approaches is that they can help to alleviate some of the main challenges of traditional MPC, which are often not covered in detail in a basic MPC course: obtaining the model in (S1b), being able to solve the potentially nonconvex problem (S1) in real time, or trying to achieve optimal solutions even for the problem of the infinite horizon and in the presence of stochastic uncertainties.

In the first part of the course, data are used to approximate or substitute the dynamic model (S1b). The term *data* in this case refers to a set of n_{data} input–output pairs, where a data output can refer to the consecutive states or to consecutive measured system outputs, depending on if full state measurement is assumed or not.

$$\mathbb{D}_{\text{dyn}} = \{((x_1(k), u_1(k)), x_1(k+1)); ((x_2(k), u_2(k)), x_2(k+1)); \\ \ldots; ((x_{n_{\text{data}}}(k), u_{n_{\text{data}}}(k)), x_{n_{\text{data}}}(k+1))\}.$$

The dataset can be used to perform explicit system identification that leads to an approximate system dynamics $f(\cdot)$, which is then used in the MPC problem (S1b). The focus on white-box, gray-box, or black-box system identification as well as the specific techniques [S9] can vary depending on the department where the course takes place. An alternative option to explicit system identification, which is in some cases equivalent, can then be presented. In this case, data can be included directly in the constraints instead of the dynamic system equations (S1b) to achieve an implicit representation of the system dynamics, following the fundamental lemma of behavioral systems theory [S10].

The second part of the course can show that data-based and learning techniques can help in mitigating another challenge of MPC: the solution of the optimization problem (S1) in real time. By analyzing the MPC problem (S1) in more detail, the concept of the parametric optimization problem can be introduced to students, where the trajectory of optimal control inputs is, for a given model, constraints and cost function, a function of the initial condition $u(\cdot)^* = \kappa(x_0)$. The concept of explicit MPC [S2] can then be introduced to motivate the idea of MPC without online optimization. Other methods that compute, instead of an exact solution, an approximate MPC feedback law $\kappa_{\text{approx}}(x_0)$ can then be presented as a possible alternative in the case of large-scale and nonlinear systems. To obtain such an approximate MPC feedback law, the dataset of interest includes data pairs of initial conditions and corresponding optimal inputs.

$$\mathbb{D}_{\text{opt}} = \{(x_1(0), u_1^*(0)); ((x_2(k), u_2(k)), x_2(k+1)); \\ \ldots; ((x_{n_{\text{data}}}(k), u_{n_{\text{data}}}(k)), x_{n_{\text{data}}}(k+1))\}.$$

At this stage, if not previously done when describing black-box methods for system identification, popular methods for function approximation can be introduced, such as neural networks $\kappa_{\text{approx}} = \mathcal{N}(x;\theta)$, which are described by a set of parameters θ.

application-oriented engineering audiences, the absence of state constraints in OCP (S8) enables one to work with conceptually easier variants of the minimum principle; cf. [S6], [S7], and [S8]. Moreover, a sequential structure of optimal control theory and numerical solution schemes followed by continuous-time NMPC material offers substantial teaching synergies:

- Optimal control and NMPC can be linked naturally.
- The discussion of the dissipativity concepts allows us to highlight the usefulness of one of the fundamental concepts in systems and control, while it does not require the introduction of additional mathematical concepts.
- Formal stability analysis of sampled-data continuous-time MPC provides a link to hybrid systems.
- For a Ph.D. course or for an applied mathematics/theory-oriented audience, optimality conditions with state constraints and sampled-data aspects as well as the existence of optimal solutions provide natural starting points to dive deeply into advanced aspects of functional analysis and to explore numerical concepts.

REFERENCES

[S4] R. E. Kalman, “Contributions to the theory of optimal control,” *Boletín de la Sociedad Matemática Mexicana*, vol. 5, no. 2, pp. 102–119, 1960.

[S5] T. Faulwasser, L. Grüne, and M. Müller, “Economic nonlinear model predictive control: Stability, optimality and performance,” *Found. Trends Syst. Control*, vol. 5, no. 1, pp. 1–98, 2018.

[S6] M. Gerdts, *Optimal Control of ODEs and DAEs*. Berlin, Germany: Walter de Gruyter, 2011.

[S7] R. F. Hartl, S. P. Sethi, and R. G. Vickson, “A survey of the maximum principles for optimal control problems with state constraints,” *SIAM Rev.*, vol. 37, no. 2, pp. 181–218, 1995, doi: 10.1137/1037043.

[S8] D. Liberzon, Calculus of Variations and Optimal Control Theory: A Concise Introduction. Princeton, NJ, USA: Princeton Univ. Press, 2012.

The optimal set of parameters that better describe the data for a given structure of the neural network can be obtained by *training* with the available data. That is, solving the optimization problem

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \frac{1}{n_{\text{data}}} \sum_{i=1}^{n_{\text{data}}} \| u_i^*(0) - \mathcal{N}(x_i(0);\theta) \|_2^2. \tag{S16}$$

In the third and last part of the course, the parallels between MPC and RL could be addressed. In particular, it might be very interesting to observe in detail the different usual notations as well as the system formulations. The typical RL setting considers that the system dynamics are described by Markov processes where state transitions are defined by a conditional probability density $\mathbb{P}(x(k+1) \mid x(k), u(k))$. Then, instead of solving the finite horizon problem (S1), the goal of RL is to find a policy π that maximizes the expected value of a *reward* $r(x(k), u(k))$ over an infinite horizon with a discount factor γ

$$\pi^* = \underset{\pi}{\operatorname{argmax}} \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r(x(k), u(k)) \mid u(k) = \pi(x(k))\right].$$

To achieve this goal, the dataset of interest includes the input–output trajectories as well as the rewards that are obtained when applying a certain input to the system

$$\begin{aligned}\mathbb{D}_{\text{rl}} = \{&(x_1(k), u_1(k), x_1(k+1), r(x_1(k), u_1(k)));\\ &\ldots; (x_{n_{\text{data}}}(k), u_{n_{\text{data}}}(k), x_{n_{\text{data}}}(k+1), r(x_{n_{\text{data}}}(k), u_{n_{\text{data}}}(k)))\}.\end{aligned}$$

Of special interest for an audience with a background in MPC can be the relationships between dynamic programming and the action-value function (or Q-function) that is defined as

$$Q_\pi(x,u) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k r(x(k), u(k)) \mid x, u\right] \tag{S17}$$

and denotes the expected reward obtained when the system is at state x and input u is used followed by the policy π. RL methods use different strategies to approximate the optimal action-value function $Q^*(x,u)$ using the available data $\mathbb{D}_{\text{rl}}$ so that the optimal policy can be computed as

$$\pi^*(x) = \underset{u}{\operatorname{argmax}}\, Q^*(x,u). \tag{S18}$$

The connections between RL and MPC can offer a natural opportunity to introduce the idea of stochastic MPC since probabilistic considerations are necessary for both methods.

The focus of the course on some of the described parts can vary largely depending on the department or particular interests of the students. At the same time, the motivation that data-based methods (including those not described here) can enable MPC when it is otherwise impossible can be a powerful argument to attract students interested in learning-based strategies to the field of MPC.

REFERENCES

[S9] L. Ljung, “Perspectives on system identification,” *Annu. Rev. Control*, vol. 34, no. 1, pp. 1–12, 2010, doi: 10.1016/j.arcontrol.2009.12.001.

[S10] J. C. Willems, P. Rapisarda, I. Markovsky, and B. L. M. De Moor, “A note on persistency of excitation,” *Syst. Control Lett.*, vol. 54, no. 4, pp. 325–329, 2005, doi: 10.1016/j.sysconle.2004.09.003.

PROSPECTS AND SYNERGIES FOR FUTURE MPC TEACHING

This blended survey and position article has attempted an assessment of the state of the art of teaching MPC in the systems and control community. Our survey indicates that MPC has secured a mature spot in the curricula of many—if not most—academic institutions with a strong engineering focus. At the same time, the evident continental bias of survey responses appears to indicate that substantial MPC research is happening in Europe. This, in turn, appears to translate into a number of dedicated and advanced MPC courses at European institutions.

Moreover, one may infer from our survey findings and from the provided curricula blueprints the hypothesis that the ever-increasing power and ubiquitous availability of open source optimization codes render teaching MPC at a late undergraduate level a feasible endeavor. Indeed, our survey results indicate that the first instances of such courses already exist.

Given that many engineering students take only one course in control, the question of whether or not an introductory MPC encounter can and should be the only control course is rather contentious and clearly depends on specific curricula and department culture. It is also reasonable to assume that undergraduate MPC courses are of particular interest to students who focus on systems and control already in the last part of their undergraduate studies. However, it goes without saying that an introductory MPC course is to be preferred over no undergraduate encounter with control.

Besides the continuous and continued evolution of MPC research—which is reflected in the substantial number of advanced MPC topics, such as stochastic, robust, economic, data-driven, etc.—the transfer of MPC knowledge to application domains and to application-specific study programs is of growing relevance for MPC teaching. Similar to mathematics education, where students majoring in math dive much deeper into several topics than typical engineering students, application-oriented engineering programs and application-driven control engineering benefit from different teaching scopes compared to the theory-oriented education of future Ph.D. students. In this context, promising application domains with a clear demand for (advanced) MPC knowledge include power and multienergy systems, automotive, mechatronics/robotics, and aeronautics, while in process control and process system engineering, MPC has already evolved into an indispensable role.

Another area in which MPC concepts flourish in research is ML for dynamic systems. Here, undergraduate education on MPC has the potential to demonstrate early that core engineering concepts are not replaceable by abstract AI and ML concepts. Rather, the dynamic system perspective and the feedback concepts, which are the major learning outcomes in systems and control, are valuable for AI/ML research and are indispensable for many applications. Hence, there is a fundamental teaching prospect beyond MPC as such—the systems and control community should push to be and to remain the key authority for conveying the dynamic systems perspective and the concept of feedback in engineering and computer science education alike.

ACKNOWLEDGMENT

We thank all participants of our survey for providing their insights. Furthermore, we thank Johannes van Randenborgh for preprocessing the survey data.

AUTHOR INFORMATION

Timm Faulwasser received the Diploma degree in engineering cybernetics at the University of Stuttgart, with majors in systems and control and philosophy, and received the Ph.D. degree from the Department of Electrical Engineering and Information Engineering, Otto-von-Guericke-University Magdeburg, Germany, in 2012. From 2008 until 2012, he was a member of the International Max Planck Research School for Analysis, Design and Optimization in Chemical and Biochemical Process Engineering Magdeburg. From 2013 to 2016 he was with the Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland, while from 2015 to 2019, he led the Optimization and Control Group at the Institute for Automation and Applied Informatics at the Karlsruhe Institute of Technology (KIT). From 2019 to 2024 he held a professorship for energy efficiency in the Department of Electrical Engineering and Information Technology, TU Dortmund University, Germany. Since April 2024 he leads the Institute of Control Systems at Hamburg University of Technology, 21073 Hamburg, Germany. His research interests include optimization-based and predictive control of nonlinear systems and networks. He serves as associate editor for *IEEE Transactions on Automatic Control* and *IEEE Control System Letters* as well as *Mathematics of Control Systems and Signals*. He received the 2021–2023 Automatica Paper Prize.

Eric Kerrigan is a professor of control and optimization at Imperial College London, SW7 2AZ, London, U.K., where he has a joint appointment in the Department of Electrical and Electronic Engineering and the Department of Aeronautics. His research interests include the design of efficient numerical methods for solving optimal control and estimation problems in real time, with applications in the design of aerospace, renewable energy, and information systems. He is an associate editor of *IEEE Transactions on Automatic Control*.

Filip Logist received the M.Sc. and Ph.D. degrees in chemical engineering from KU Leuven, Belgium, in 2003 and 2008. He worked full-time at KU Leuven from 2003 until 2016 with research stays at, for example, the RWTH Aachen and the University of Heidelberg, Germany. In 2016, he joined BASF Antwerpen, 2040 Antwerpen, Belgium, to become part of the Advanced Process Control Global Competence Team of BASF while still continuing as a (guest) professor in advanced process control at KU Leuven and a scientific collaborator at the University of Liège. He also provides guest lectures at the Universities of Ghent and Antwerp. His research interests include modeling and identification of dynamic (bio)chemical systems as well as their model-based control and optimization. For his realizations of model-based process optimization solutions in industry, he has been awarded a BASF Antwerpen Excellence Award in 2022 and 2023.

Sergio Lucia received the M.Sc. degree in electrical engineering from the University of Zaragoza, Spain, in 2010, and the Dr. Ing. degree in optimization and automatic control from TU Dortmund University, Germany, in 2014. He joined the Otto-von-Guericke Universität Magdeburg and visited the Massachusetts Institute of Technology as a postdoctoral fellow. He was an assistant professor at TU Berlin between 2017 and 2020. Since 2020, he has been a professor at TU Dortmund University, 44227, Dortmund, Germany, and head of the Chair of Process Automation Systems. His research interests include decision making under uncertainty and optimization as well as the interplay between machine learning techniques and control theory. He is currently an associate editor of the *Journal of Process Control* and the *Journal of Optimal Control Applications and Methods*.

Martin Mönnigmann received the Diploma and Ph.D. degrees from RWTH Aachen University, Germany, in 1998 and 2004, respectively. He was a postdoctoral research scholar at Princeton University from 2004 to 2005, deputy director of the Aachen Institute for Advanced Study in Computational Engineering from 2005 to 2007, and assistant professor at Technische Universität Braunschweig, Germany, from 2007 to 2009. In 2009, he was appointed full professor and head of the chair for Automatic Control and Systems Theory, Ruhr-Universität Bochum, 44801, Bochum, Germany. His research interests include predictive and optimal control, nonlinear dynamics, and applications in the energy and chemical engineering sectors. He is an associate editor of the *Journal of Process Control* and *Automatica*.

Alessandra Parisio is currently a reader (associate professor) with the Department of Electrical and Electronic Engineering, University of Manchester, M13 9PL, Manchester, U.K. Her research interests include energy management systems under uncertainty, model predictive control, stochastic constrained control, and distributed optimization for power systems. She was a recipient of the IEEE PES Outstanding Engineer Award in January 2021 and the Energy and Buildings Best Paper Award (for 2008–2017) in January 2019. She is the editor of *IEEE Transactions on Control of Network Systems* and *IEEE Transactions on Automation and Science Engineering*.

Moritz Schulze Darup (moritz.schulzedarup@tu-dortmund.de) received the Diploma degree in mechanical engineering, the B.Sc. degree in physics, and the Ph.D. in automatic control from Ruhr-Universität Bochum, Germany, in 2008, 2010, and 2014, respectively. From 2014 to 2016, he visited Oxford University, U.K., as a postdoctoral fellow. In 2017, he joined Paderborn University, Germany, as a lecturer. By 2019, he advanced to the position of assistant professor and leader of an Emmy Noether Group for Encrypted Control. Since 2020, he has been a full professor of control and cyberphysical systems at TU Dortmund University, 44227, Dortmund, Germany. His research interests include secure, predictive, and data-driven control. He is a Senior Member of IEEE.

REFERENCES

[1] E. B. Lee and L. Markus, *Foundations of Optimal Control Theory* (The SIAM Series in Applied Mathematics). New York, NY, USA: Wiley, 1967.

[2] A. I. Propoi, "Application of linear programming methods for the synthesis of automatic sampled-data systems," *Avtomat. i Telemekh*, vol. 24, no. 7, pp. 912–920, 1963.

[3] C. R. Cutler, B. L. Ramaker, "Dynamic matrix control - A computer control algorithm," in *Proc. Joint Autom. Control Conf.*, San Francisco, Aug. 1980.

[4] J. Richalet, A. Rault, J. L. Testud, and J. Papon, "Model predictive heuristic control: Applications to industrial processes," *Automatica*, vol. 14, no. 5, pp. 413–428, 1978, doi: 10.1016/0005-1098(78)90001-8.

[5] J. M. Maciejowski, *Predictive Control With Constraints*. London, U.K.: Prentice Hall, 2002.

[6] M. Athans and P. L. Falb, *Optimal Control - An Introduction to Theory and Its Applications*. New York, NY, USA: McGraw-Hill Book Company, 1966.

[7] A. E. Bryson and Y.-C. Ho, *Applied Optimal Control*. Waltham, MA, USA: Ginn and Company, 1969.

[8] D. E. Kirk, *Optimal Control Theory: An Introduction*. New York, NY, USA: Dover, 2004.

[9] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Eng. Pract.*, vol. 11, no. 7, pp. 733–764, 2003, doi: 10.1016/S0967-0661(02)00186-7.

[10] T. Faulwasser, S. Lucia, M. Schulze Darup, and M. Mönnigmann, "Teaching MPC: Which way to the promised land?" *IFAC-PapersOnLine*, vol. 54, no. 6, pp. 238–243, 2021, doi: 10.1016/j.ifacol.2021.08.551.

[11] ainewsletter.com, Jan. 2005, "Ai Newsletter," Archived newsletter, https://www.ainewsletter.com/

[12] K. J. Astrom, R. E. Klein, and A. Lennartsson, "Bicycle dynamics and control: Adapted bicycles for education and research," *IEEE Control Syst. Mag.*, vol. 25, no. 4, pp. 26–47, Aug. 2005, doi: 10.1109/MCS.2005.1499389.

[13] M. K. Jouaneh and W. J. Palm, III. "Control systems take-home experiments [Focus on Education]," *IEEE Control Syst. Mag.*, vol. 33, no. 4, pp. 44–53, Aug. 2013, doi: 10.1109/MCS.2013.2258764.

[14] J. A. Rossiter et al., "Control education for societal-scale challenges: A community roadmap," *Annu. Rev. Control*, vol. 55, May 2023, pp. 1–7, doi: 10.1016/j.arcontrol.2023.03.007.

[15] K. J. Åström and R. M. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton, NJ, USA: Princeton Univ. Press, 2021.

[16] A. Rossiter, A. Serbezov, A. Visioli, K. Žáková, and M. Huba, "A survey of international views on a first course in systems and control for engineering undergraduates," *IFAC J. Syst. Control*, vol. 13, Sep. 2020, Art. no. 100092, doi: 10.1016/j.ifacsc.2020.100092.

[17] F. Borelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. New York, NY, USA: Springer-Verlag, 2017.

[18] E. F. Camacho and C. B. Alba, *Model Predictive Control*, 2nd ed. New York, NY, USA: Springer-Verlag, 2007, 1st edition published in 1999.

[19] L. Grüne and J. Pannek, *Nonlinear Model Predictive Control*. New York, NY, USA: Springer-Verlag, 2017.

[20] B. Kouvaritakis and M. Cannon, *Model Predictive Control*. New York, NY, USA: Springer-Verlag, 2016.

[21] J. B. Rawlings, D. Q. Mayne, and M. M. Diehl, *Model Predictive Control: Theory, Computation, and Design*. Madison, WI, USA: Nob Hill Publishing, 2017.