

GENERAL INSTRUCTION

- **Authors:** We cannot accept new source files as corrections for your paper. If possible, please annotate the PDF proof we have sent you with your corrections and upload it via the Author Gateway. Alternatively, you may send us your corrections in list format. You may also upload revised graphics via the Author Gateway.

Queries

- Q1. Author: We have made a number of changes to the wording of the paper, English grammar, and readability. Please check the whole text for correctness and please suggest any improvements.
- Q2. Author: Please confirm or add details for any funding or financial support for the research of this article.
- Q3. Author: Please provide the expansion of LQR, LMI, and MPC at first mention.
- Q4. Author: Please check Ref. [15] for correctness.
- Q5. Author: Please provide the subject in which Sébastien Gros received the Ph.D. degree.
- Q6. Author: Please check the sentence “He was part...” for correctness.
- Q7. Author: Please provide the subject in which Mario Zanon received the Diplôme d’Ingénieur degree.

Data-Driven Economic NMPC Using Reinforcement Learning

Sébastien Gros^{ID} and Mario Zanon^{ID}

Abstract—Reinforcement learning (RL) is a powerful tool to perform data-driven optimal control without relying on a model of the system. However, RL struggles to provide hard guarantees on the behavior of the resulting control scheme. In contrast, nonlinear model predictive control (NMPC) and economic NMPC (ENMPC) are standard tools for the closed-loop optimal control of complex systems with constraints and limitations, and benefit from a rich theory to assess their closed-loop behavior. Unfortunately, the performance of (E)NMPC hinges on the quality of the model underlying the control scheme. In this paper, we show that an (E)NMPC scheme can be tuned to deliver the optimal policy of the real system even when using a wrong model. This result also holds for real systems having stochastic dynamics. This entails that ENMPC can be used as a new type of function approximator within RL. Furthermore, we investigate our results in the context of ENMPC and formally connect them to the concept of dissipativity, which is central for the ENMPC stability. Finally, we detail how these results can be used to deploy classic RL tools for tuning (E)NMPC schemes. We apply these tools on both, a classical linear MPC setting and a standard nonlinear example, from the ENMPC literature.

Index Terms—Adaptive nonlinear model predictive control (NMPC), economic NMPC (ENMPC), reinforcement learning, strict dissipativity.

I. INTRODUCTION

R EINFORCEMENT learning (RL) is a powerful tool for tackling Markov decision processes (MDP) without depending on a model of the probability distributions underlying the state transitions. Indeed, most RL methods rely purely on observed state transitions, and realizations of the stage cost in order to increase the performance of the control policy. RL has drawn an increasing attention thanks to its striking accomplishments ranging from computers beating Chess and Go masters [31], to robot learning to walk or fly without supervision [1], [40].

Manuscript received October 16, 2018; revised April 8, 2019; accepted April 16, 2019. Date of publication; date of current version. Recommended by Associate Editor M. Alamir. (Corresponding author: Mario Zanon.)

S. Gros is with the Department of Engineering Cybernetics, Norwegian University of Science and Technology, 7491 Trondheim, Norway (e-mail: sebastien.gros@ntnu.no).

M. Zanon is with the IMT School for Advanced Studies 55100 Lucca, Italy (e-mail: mario.zanon@imtlucca.it).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TAC.2019.2913768

Most RL methods are based on learning the optimal control policy for the real system either directly, or indirectly. Indirect methods rely on learning an approximation of the optimal action-value function underlying the MDP, typically using variants of temporal-difference learning [19]. Since the action-value function is, in general, unknown *a priori*, a generic function approximator is typically used to approximate it. A common choice in the RL community is to use a deep neural network (DNN).

Direct RL methods seek to learn the optimal policy directly. Most direct RL methods are based on the stochastic or deterministic policy gradient methods, see [32], [37]. Both rely on carrying an approximation of the action-value function, or at least of the value function underlying the policy. Similar to indirect methods, direct RL methods also typically use DNNs to approximate the optimal policy and the associated (action) value function.

Unfortunately, the closed-loop behavior of a system subject to an approximate optimal policy supported by a DNN or a generic function approximation can be difficult to formally analyze. It can therefore be difficult to generate certificates of the behavior of a system controlled by a generic RL algorithm. This issue is especially salient when dealing with safety-critical systems. The development of safe RL methods, which aims at tackling this issue, is an open field of research [18].

Nonlinear model predictive control (NMPC) is a formal control method based on solving at every time instant an optimal control problem to generate the optimal control policy. The optimal control problem seeks to minimize a sum of stage costs over a prediction horizon, subject to state trajectories provided by a model of the real system, the current observed state of the real system, and possibly state and input constraints to be respected. The optimal control problem then delivers an entire input and state sequence spanning the prediction horizon. However, only the first input is applied to the real system. At the next time instant, the entire optimal control problem is solved again using a new estimation of the state of the system.

If the system model underlying the NMPC scheme is perfect, and with the addition of an adequate terminal cost, the NMPC scheme delivers the optimal control policy. Classic NMPC is based on a stage cost lower bounded by \mathcal{K}_∞ functions, while economic NMPC (ENMPC) accepts a generic stage cost [4], [12], [26]. A rich and mature theory exists in the literature to analyze the properties of classic NMPC when operating in closed loop with a real system, establishing desirable key properties, such as recursive feasibility and stability [16], [23], [27].

ENMPC has recently attracted the attention of the research community, and a stability theory has been developed fairly recently [4], [6], [12], [24], [26], but is arguably still under development. Since it can tackle the system constraints directly and since it benefits from a rich theory analyzing its behavior, (E)NMPC is arguably an ideal candidate for safety-critical applications.

Unfortunately, the performance of (E)NMPC schemes relies on having a good model of the system to be controlled. A data-driven adaption of the NMPC model to better fit the real system is a fairly obvious approach to tackle this issue. However, since the model does not necessarily match the real system, fitting the model to the data does not necessarily result in the (E)NMPC scheme delivering the optimal policy, and can even be counterproductive. Some attempts have been recently proposed to tackle this problem, such as in [17].

The problem of optimizing a system based on a model having the wrong structure is well known in the field of real-time optimization [2], [3], [13], [14], [22], and has been addressed via the modifier approach [15], [22], [28], [29], [38], whereby the cost function of the optimization problem is adapted rather than the model.

In this paper, we propose to use (E)NMPC schemes instead of DNNs to support the parametrization to approximate the (action) value functions and the policy. Similarly to the idea originated in RTO, we show that the NMPC scheme can deliver the optimal control policy even if the underlying model is incorrect, by adapting the stage cost, terminal cost, and constraints only. This observation is applicable to any NMPC scheme, such as classic NMPC, ENMPC, robust and stochastic ENMPC. Furthermore, we establish strong connections between this cost adaptation and the concept of strict dissipativity, which is fundamental to the stability theory of ENMPC.

One practical outcome of the theory proposed in this paper is that all RL techniques can be directly used to tune the NMPC scheme to increase its performance on the real system. Because the theory proposed is very generic, this observation holds for a stochastic system being controlled by an (E)NMPC scheme based on a deterministic model or a robust NMPC scheme using a scenario tree. Another practical outcome of the theory proposed in this paper is that if a stage cost attached to a given system yields a stabilizing optimal control policy, then an ENMPC scheme with a positive stage cost can, in principle, be tuned to deliver the optimal policy. A last practical outcome of the proposed theory is that using (E)NMPC, as a parametrization for RL instead of DNN, allows one to use the rich theory underlying (E)NMPC schemes in the context of RL, and deliver certificates on the behavior of the policy resulting from the learning process.

The combination of learning and control techniques has been proposed in [7], [8], [20], [25]. To the best of our knowledge, however, our paper is the first work proposing to use NMPC as a function approximator in RL and investigating the connection between RL and economic MPC.

The paper is structured as follows. Section II establishes the fundamental result of the paper, showing that an (E)NMPC scheme based on the wrong model can, under some conditions, nonetheless deliver the optimal control policy. Section III details

how these results can be used in practice. Section IV details how some classic RL techniques can be deployed to adjust the (E)NMPC parameters. Section V further develops the theory and details its connection to the fundamental concept of strict dissipativity underlying the stability theory of ENMPC, and details its consequences for using ENMPC as a parametrization for RL. Section VI deploys the theory in the case of LQR with Gaussian noise, for which all the objects discussed in the theory can be built explicitly, and their practical meaning assessed. Section VII proposes some illustrative examples.

II. OPTIMAL POLICY BASED ON AN INEXACT MODEL

We will consider that the real system we want to control is described by a discrete MDP having the (possibly) stochastic state transition dynamics

$$\mathbb{P}[s_+ | s, a] \quad (1)$$

where s, a is the current state-input pair and s_+ is the subsequent one. We will label $L(s, a)$ the stage cost associated to the MDP, possibly infinite for some state-input pairs s, a , which, we will assume, can take the form

$$L(s, a) = l(s, a) + \mathcal{I}_\infty(h(s, a)) + \mathcal{I}_\infty(g(a)) \quad (2)$$

where we use the indicator function

$$\mathcal{I}_\infty(x) = \begin{cases} \infty & \text{if } x_i > 0 \text{ for some } i \\ 0 & \text{otherwise} \end{cases}. \quad (3)$$

In (2), function l captures the cost given to different state-input pairs, while the constraints

$$g(a) \leq 0, \quad h(s, a) \leq 0 \quad (4)$$

capture undesirable state and inputs, and infinite values are given to L when (4) is violated. Note that we have separated pure input constraints and mixed constraints for reasons that will be made clear later on.

With the addition of a discount factor $0 < \gamma \leq 1$, (1) and (2) yield the optimal policy $\pi_\star(s)$. Note that the notation (1) is standard in the literature on MDPs, while the control literature typically uses the notation $s_+ = f(s, a, \zeta)$, where ζ is a stochastic variable and f a possibly nonlinear function. The action-value function Q_\star and value function V_\star associated with the MDP are defined by the Bellman equations [9]

$$Q_\star(s, a) = L(s, a) + \gamma \mathbb{E}[V_\star(s_+) | s, a] \quad (5a)$$

$$V_\star(s) = Q_\star(s, \pi_\star(s)) = \min_a Q_\star(s, a). \quad (5b)$$

Throughout the paper we will assume that the MDP underlying the real system, the associated stage cost L , and the discount factor γ yield a well-posed problem, i.e., the value functions defined by (5) are well posed, and finite over some sets.

We then consider a model of the real system having the state transition dynamics

$$\mathbb{P}[\hat{s}_+ | s, a] \quad (6)$$

which typically do not match (1) perfectly. Note that (6) trivially includes deterministic models as a special case. Consider a stage

183 cost defined as

$$\hat{L}(s, a) = \begin{cases} Q_*(s, a) - \gamma V^+(s, a) & \text{if } |V^+(s, a)| < \infty \\ \infty & \text{otherwise} \end{cases} \quad (7)$$

184 where $V^+(s, a) = \mathbb{E}[V_*(\hat{s}_+) | s, a]$ and where the expectation
185 is taken over the distribution (6). It is useful here to specify that
186 the conditional formulation of the stage cost \hat{L} proposed in (7)
187 is a technicality dedicated to having a well-defined \hat{L} even when
188 both the action value function and value function take infinite
189 values.

190 We establish next the central theorem of this paper, stating
191 that under some conditions, the optimal policy π_* that minimizes
192 the stage cost L for the true dynamics (1) is also generated by
193 using model (6) combined with the stage cost \hat{L} . Hence it is
194 possible to generate the optimal policy based on a wrong model
195 by modifying the stage cost. It may be useful to specify here that
196 our approach further in the paper will be to bypass the possibly
197 difficult evaluation of (7), and replace it by learning \hat{L} directly
198 from the data, see Section IV.

199 *Theorem 1:* Consider the optimal value function

$$\hat{V}_N(s) = \min_{\pi} \mathbb{E} \left[\gamma^N V_*(\hat{s}_N^\pi) + \sum_{k=0}^{N-1} \gamma^k \hat{L}(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right] \quad (8)$$

200 associated to the stage cost (7), the state transition model (6), and
201 the terminal cost $V_*(s)$ over an optimization horizon N . Here
202 we define $\hat{s}_{0,\dots,N}^\pi$ as the (possibly stochastic) trajectories of the
203 state transition model $\mathbb{P}[\hat{s}_+ | s, a]$ under a policy π , starting
204 from $\hat{s}_0^\pi = s$. We will label $\hat{\pi}$ as the optimal policy associated
205 to $\hat{V}_N(s)$ and $\hat{Q}_N(s, a)$ the associated action-value functions.

206 Consider the set \mathcal{S} such that

$$|\mathbb{E}[V_*(\hat{s}_k^\pi)]| < \infty, \quad \forall s \in \mathcal{S}, \quad \forall k. \quad (9)$$

207 Then the following identities hold on \mathcal{S} .

- 208 i) $\hat{V}_N(s) = V_*(s)$.
- 209 ii) $\hat{\pi}(s) = \pi_*(s)$.
- 210 iii) $\hat{Q}_N(s, a) = Q_*(s, a)$ for the inputs a such that
211 $|\mathbb{E}[V_*(\hat{s}_+) | s, a]| < \infty$.

212 *Proof:* Let us consider the N -step value function \hat{V}_N^π asso-
213 ciated to the stage cost \hat{L} , the state transition model $\mathbb{P}[\hat{s}_+ | s, a]$
214 and a policy π , defined as

$$\hat{V}_N^\pi(s) = \mathbb{E} \left[\gamma^N V_*(\hat{s}_N^\pi) + \sum_{k=0}^{N-1} \gamma^k \hat{L}(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right]. \quad (10)$$

215 Assumption (9) ensures that, at least for $\pi = \pi_*$, all the terms
216 in the sum in (10) have a finite expected value for $s \in \mathcal{S}$, such
217 that $\hat{V}_N^\pi(s)$ is well defined and finite over \mathcal{S} for some policies
218 π . Using a telescopic sum, we can rewrite (10) as

$$\hat{V}_N^\pi(s) = Q_*(s, \pi(s)) + \mathbb{E} \left[\sum_{k=1}^{N-1} \gamma^k A_*(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right] \quad (11)$$

219 where the advantage function A_* is defined as

$$A_*(s, a) = \begin{cases} Q_*(s, a) - V_*(s) & \text{if } |Q_*(s, a)| < \infty \\ \infty & \text{otherwise} \end{cases}. \quad (12)$$

Using the Bellman equalities

$$\min_{\pi} Q_*(s, \pi(s)) = Q_*(s, \pi_*(s)) = V_*(s) \quad (13a)$$

$$\min_{\pi} A_*(s, \pi(s)) = A_*(s, \pi_*(s)) = 0 \quad (13b)$$

we observe that all terms in (11) are minimized by the policy
221 π_* , such that the following equalities hold on \mathcal{S} :

$$\hat{V}_N(s) = \min_{\pi} \hat{V}_N^\pi(s) = \hat{V}_N^{\pi_*}(s) = V_*(s) \quad (14)$$

where the first equality holds by definition, the second holds
223 because π_* is the minimizer of \hat{V}_N^π , and the last equality holds
224 from (13). It follows that $\hat{V}_N(s) = V_*(s)$ holds on \mathcal{S} for any
225 N , hence the identities (i) and (ii) hold. We furthermore observe
226 that on \mathcal{S} and for any input a such that $|\mathbb{E}[V_*(\hat{s}_+) | s, a]| < \infty$,
227 the following equalities hold:

$$\begin{aligned} \hat{Q}_N(s, a) &= \hat{L}(s, a) + \gamma \mathbb{E} \left[\hat{V}_{N-1}(\hat{s}_+) | s, a \right] \\ &= Q_*(s, a) + \gamma \mathbb{E} \left[\hat{V}_{N-1}(\hat{s}_+) - V_*(\hat{s}_+) | s, a \right] \\ &= Q_*(s, a) \end{aligned} \quad (15)$$

which yields statement (iii). ■ 229

Note that if the transition model is exact, i.e., $\mathbb{P}[\hat{s}_+ | s, a] = \mathbb{P}[s_+ | s, a]$, then the stage cost defined in (7) satisfies $\hat{L}(s, a) = L(s, a)$, $\forall s \in \mathcal{S}$, with \mathcal{S} defined in (9). It is interesting to discuss
232 here the case $N \rightarrow \infty$ for which, under some conditions, the
233 terminal cost can be dismissed. We detail this in the following
234 Corollary. 235

Corollary 1: Under the assumptions of Theorem 1 and under
236 the additional assumption

$$\lim_{N \rightarrow \infty} \mathbb{E} [\gamma^N V_*(\hat{s}_N^\pi)] = 0 \quad (16)$$

then

$$\hat{V}_\infty(s) = \lim_{N \rightarrow \infty} \min_{\pi} \mathbb{E} \left[\sum_{k=0}^{N-1} \gamma^k \hat{L}(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right] = V_*(s) \quad (17)$$

and the equalities $\hat{\pi}(s) = \pi_*(s)$ and $\hat{Q}_\infty(s, a) = Q_*(s, a)$ hold
239 for $|\mathbb{E}[V_*(\hat{s}_+) | s, a]| < \infty$. 240

Proof: Let us consider the N -step value function \hat{V}_N^π asso-
241 ciated to the stage cost \hat{L} , the state transition model $\mathbb{P}[\hat{s}_+ | s, a]$
242 and a policy π defined as

$$\hat{V}_\infty^\pi(s) = \lim_{N \rightarrow \infty} \mathbb{E} \left[\sum_{k=0}^{N-1} \gamma^k \hat{L}(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right] \quad (18a)$$

$$\begin{aligned} &= Q_*(s, \pi(s)) + \lim_{N \rightarrow \infty} \mathbb{E} \left[-\gamma^N V_*(\hat{s}_N^\pi) \right. \\ &\quad \left. + \sum_{k=1}^{N-1} \gamma^k A_*(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right] \end{aligned} \quad (18b)$$

where (18b) holds as, by assumption, all terms in the summa-
244 tions in (18) have finite expected values. This entails that

$$\hat{V}_\infty^{\pi_*}(s) = \lim_{N \rightarrow \infty} V_*(s) + \mathbb{E} \left[-\gamma^N V_*(\hat{s}_N^{\pi_*}) \right]. \quad (19)$$

246 Using (16), we finally observe that the following inequalities
247 hold on S :

$$\hat{V}_\infty(s) = \min_{\pi} \hat{V}_\infty^\pi(s) = \hat{V}_\infty^{\pi^*}(s) = V_*(s) \quad (20)$$

248 and are justified similarly to (14). \blacksquare

- 249 Let us next make a few observations regarding Theorem 1.
- 250 1) Assumption (9) requires that the model trajectories under
251 policy π_* are contained within the set where the value
252 function V_* is finite with a unitary probability.
- 253 2) Assumption (16) can be construed as some form of
254 stability condition on the model dynamics under pol-
255 icy π_* . This observation is especially clear in the case
256 $\gamma = 1$, which then imposes the condition $\lim_{N \rightarrow \infty} \mathbb{E}[V_*(\hat{s}_N^\pi)] = 0$.
- 257 3) Condition (16) is not required in Theorem 1, however,
258 it is highly desirable to fulfil it in practice when a finite
259 horizon is required, so that the terminal cost in (8) has a
260 limited impact on the optimal policy $\hat{\pi}$.
- 261 4) Though very similar, Theorem 1 and Corollary 1 cover
262 different cases, since taking the limit for $N \rightarrow \infty$ in
263 Theorem 1 does not require Assumption (16) to hold.
- 264 5) Theorem 1 proposes a modified stage cost and a termi-
265 nal cost such that the finite-horizon problem (8) delivers
266 the optimal policy π_* . It ought to be noted that prob-
267 lem (8) would actually use $\pi_*(\hat{s}_k)$ to select the inputs at
268 every stage k of the prediction $\hat{s}_{0,\dots,N-1}$. However, iden-
269 tities (i)–(iii) of Theorem 1 do not necessarily require this
270 restriction, i.e., it is sufficient to find a stage cost and a
271 terminal cost such that the resulting optimal control prob-
272 lem yields π_* as its *initial* policy (at stage $k = 0$) only to
273 get identities (i)–(iii).

III. ENMPC AS A FUNCTION APPROXIMATOR

275 We will now detail how the theory presented above applies
276 to using a parametrized ENMPC scheme to approximate the
277 optimal policy and value functions π_* and V_* , Q_* , even if the
278 model underlying the ENMPC scheme is not highly accurate.
279 We will detail in Section IV how the ENMPC parameters can
280 be adjusted to achieve this approximation.

281 The constraints (4) can be used in the ENMPC scheme to
282 explicitly exclude undesirable states and inputs. Since the EN-
283 MPC scheme will be based on an imperfect model f_θ , it will
284 seek the minimization of the modified stage cost $\hat{L}(s, a)$ rather
285 than the original one $L(s, a)$. As a result, while the pure input
286 constraints $g(a) \leq 0$ are arguably fixed, the mixed constraints
287 used in the ENMPC scheme ought to be modified as well, in
288 order to capture the domain where $\hat{L}(s, a)$ is finite. Since \hat{L} is
289 not known *a priori*, the domain where it is finite depends, among
290 other things, on the discrepancy between f_θ and (1), and will
291 have to be learned. We will therefore consider introducing the
292 parametrized mixed constraints $h_\theta(s, a) \leq 0$ in the ENMPC
293 scheme, where θ will be parameters that can be adjusted via
294 RL tools. Equation (26) and Corollary 2 below provide a more
295 formal explanation of these observations.

296 Since RL tools cannot handle infinite penalties, we will need
297 to consider a relaxed version of L and of the mixed constraints

h_θ . We can now formulate the parametrized ENMPC scheme
298 that will serve as a function approximation in the RL tools. \blacksquare

We will consider a parametrization of the value function V_*
301 using the following ENMPC scheme parametrized by θ :
302

$$V_\theta(s) = \min_{u, x, \sigma} \lambda_\theta(x_0) + \gamma^N (V_\theta^f(x_N) + w_f^\top \sigma_N) \\ + \sum_{k=0}^{N-1} \gamma^k (l_\theta(x_k, u_k) + w^\top \sigma_k) \quad (21a)$$

$$\text{s.t. } x_{k+1} = f_\theta(x_k, u_k), \quad x_0 = s \quad (21b)$$

$$g(u_k) \leq 0 \quad (21c)$$

$$h_\theta(x_k, u_k) \leq \sigma_k, \quad h_\theta^f(x_N) \leq \sigma_N. \quad (21d)$$

Problem (21) is a classic ENMPC formulation when $\gamma = 1$
303 and $\lambda_\theta = 0$ [16], [27]. Note that we have used x, u in order
304 to clearly distinguish the ENMPC prediction from the actual
305 closed-loop state and control trajectory. \blacksquare

We observe that the ENMPC scheme (21) holds a model
307 parametrization f_θ , a constraint parametrization h_θ as discussed
308 above, and a parametrization of the stage cost l_θ and terminal
309 cost V_θ^f . The extra cost λ_θ is discussed in detail in Section V-
310 A. Reasonable guesses for these functions are to use $l_\theta = l$,
311 $h_\theta = h$, and any classic heuristic to build the terminal cost
312 approximation V_θ^f , such as a quadratic cost stemming from the
313 LQR approximation of the ENMPC scheme. RL tools are then
314 used to modify these initial guesses toward higher closed-loop
315 performances. \blacksquare

The ℓ_1 relaxation of the mixed constraints (21d) relying on
317 slack variables σ_k is fairly standard in practical implementa-
318 tion of (EN)MPC schemes. For w, w_f large enough, the solution
319 to (21) is identical to the unrelaxed one whenever a feasible
320 trajectory exists for the initial state s [30]. In this case, we refer
321 to the relaxation as *exact*. The specific role of the relaxation will
322 be detailed in Section IV-C. Function λ_θ in (21a) is not required
323 anywhere in the following developments, but will play a central
324 role in forming nominal stability guarantees of the ENMPC
325 scheme (21), see Section V. We define the policy
326

$$\pi_\theta(s) = u_0^* \quad (22)$$

where u_0^* is the first element of the input sequence u_0^*, \dots, u_{N-1}^*
327 solution of (21) for a given s . We additionally define the action-
328 value function Q_θ
329

$$Q_\theta(s, a) = \min_{u, x} \quad (21a) \quad (23a)$$

$$\text{s.t. } (21b) - (21d) \quad (23b)$$

$$u_0 = a. \quad (23c)$$

Note that the proposed parametrization trivially satisfies the
330 fundamental equalities underlying the Bellman equations, i.e.,
331

$$\pi_\theta(s) = \arg \min_a Q_\theta(s, a), \quad V_\theta(s) = \min_a Q_\theta(s, a). \quad (24)$$

Let us make some key observations on (21) and (22). The
332 deterministic model (21b) can be construed as a special case of
333

334 the stochastic state transition (6), using

$$\mathbb{P}[\hat{s}_+ | s, a] = \delta(\hat{s}_+ - f(s, a)) \quad (25)$$

335 where δ is the Dirac distribution. Moreover, suppose that we
336 can select θ such that $\lambda_\theta = 0$ and such that the stage cost and
337 constraints in (21) satisfy

$$\hat{L}(s, a) = l_\theta(s, a) + \mathcal{I}_\infty(h_\theta(s, a)) + \mathcal{I}_\infty(g_\theta(a)) \quad (26a)$$

$$V_*(s) = V_\theta^f(s) + \mathcal{I}_\infty(h_\theta^f(s)) \quad (26b)$$

338 for \hat{L} given by (7), where $l_\theta, V_\theta^f < \infty$.

339 *Corollary 2 (of Theorem 1):* Assume that the NMPC
340 scheme (21) is parametrized using a rich enough parametrization
341 with an exact relaxation (i.e., w, w_f large enough). Then,
342 the NMPC scheme (21) delivers the optimal policy π_* and
343 value functions V_*, Q_* for any state s for which assumption
344 (9) is satisfied.

345 *Proof:* By assumption, there exists θ such that (26) holds,
346 then Theorem 1 directly yields the desired result. ■

347 Note that assumption (9) entails that \mathcal{S} is the forward-invariant
348 set of the dynamics (21b) under the policy π_* , associated to the
349 condition $|V_*(s)| < \infty$.

350 Unfortunately, a parameter θ satisfying (26) is clearly not
351 guaranteed to exist, and does not exist in most nontrivial practical
352 cases. Indeed, the modified stage cost \hat{L} defined by (7)
353 can be highly intricate, and satisfying (26) can require a very
354 elaborate parametrization of the ENMPC cost and constraints.
355 Even if a θ satisfying (26) does exist, finding it is arguably
356 highly difficult as evaluating (7) can be extremely demanding
357 and requires the knowledge of the real stochastic state transition
358 (6). In this paper, we propose to circumvent this difficulty by
359 1) relying on a limited parametrization of the ENMPC scheme,
360 at the price of not achieving $\pi_\theta = \pi_*$ exactly, and 2) deploying
361 RL techniques in order to adjust the ENMPC parameters θ , so
362 as to avoid computing \hat{L} altogether (see Section IV). In the RL
363 context, we ought to consider the ENMPC scheme (21) as a
364 function approximator for V_*, Q_* , and π_* .

365 The choice of cost parametrization is clearly important, but
366 beyond the scope of this paper. In the example below (see
367 Section VII), we have made fairly trivial choices and obtained
368 interesting results nonetheless. It is clear, however, that a rich
369 parametrization in theory is desirable. It is also desirable that the
370 stage cost and terminal cost approximations l_θ and V_θ^f always
371 remain (quasi)convex in order to facilitate the computation of
372 the NMPC solution. In contrast, the parametrization of the initial
373 cost λ_θ can be unrestricted. In the future, we will investigate
374 rich function approximations, including positive sums of convex
375 functions and sum-of-square approaches.

376 A. Robust NMPC Using Scenario Trees

377 Robust NMPC implemented via scenario tree approaches also
378 readily fits in the framework presented here. Indeed, the scenario
379 tree can be construed as a stochastic process with a discrete
380 probability distribution. More specifically, the stochastic model

state transition for a scenario tree dynamics reads as

$$\mathbb{P}[\hat{s}_+ | s, a] = \sum_{i=1}^{N_s} W_i \delta(\hat{s}_+ - f_i(s, a)) \quad (27)$$

382 where f_{1,\dots,N_s} are the N_s different dynamics underlying the
383 scenario tree, and W_{1,\dots,N_s} are the associated probabilities,
384 with $W_i \geq 0$ and $\sum_{i=1}^{N_s} W_i = 1$. All the observations made in
385 Section III are then also valid in this context. One ought then to
386 see the discrete probability distribution w_i as part of the parameters
387 to be adjusted in the (E)NMPC scheme.

B. Model Parametrization

389 As detailed in Section III, the ENMPC scheme (21) can, in principle,
390 capture the optimal policy π_* without having to adjust the model (21b). This observation, however, does not preclude an adaptation of the model (21b) in order to drive the NMPC policy π_θ toward the optimal one π_* , and one can easily argue that allowing such an adaptation introduces additional freedom for the NMPC scheme (21) to better approximate the optimal policy π_* . The interplay between the cost and constraints adaptation and the model adaptation is the object of current research.

IV. RL FOR ENMPC

399 Theorem 1 guarantees that it is, in theory, possible to generate
400 the optimal policy and value functions using an ENMPC scheme
401 based on a possibly inaccurate model. In practice, one has to
402 rely on ad hoc parametrization θ of the NMPC scheme, yielding
403 the value function $Q_\theta(s, a)$, $V_\theta(s)$ and the policy $\pi_\theta(s)$. The
404 goal is then to adjust the parameters θ such that the ENMPC
405 scheme policy fits the optimal policy as closely as possible.

406 Because computing \hat{L} given by (7) is difficult and requires the
407 knowledge of the true dynamics, we will rely on RL techniques
408 to adjust the parameters θ . We will focus here on classical RL
409 approaches. These techniques typically require the sensitivities
410 of the value functions. We briefly detail next how to compute
411 these sensitivities for (21)–(23).

A. Sensitivities of the ENMPC Scheme

412 We detail next how to evaluate the gradients of functions
413 $Q_\theta, V_\theta, \pi_\theta$. To that end, let us define the Lagrange function
414 associated to the ENMPC Problem (23) as
415

$$\begin{aligned} \mathcal{L}_\theta(s, y) = & \lambda_\theta(x_0) + \gamma^N V_\theta^f(x_N) + \chi_0^\top (x_0 - s) + \mu_N^\top h_\theta^f(x_N) \\ & + \sum_{k=0}^{N-1} \chi_{k+1}^\top (f_\theta(x_k, u_k) - x_{k+1}) + \nu_k^\top g_\theta(u_k) \\ & + \gamma^k L_\theta(x_k, u_k) + \mu_k^\top h_\theta(x_k, u_k) + \zeta^\top (u_0 - a) \end{aligned}$$

416 where χ, μ, ν, ζ are the multipliers associated to constraints
417 (21b)–(21d), and (23b) and (23c), respectively, and we will
418 note $y = (x, u, \chi, \mu, \nu, \zeta)$ the primal-dual variables associated
419 to (23).

420 Note that, for $\zeta = 0$, $\mathcal{L}_\theta(s, y)$ is the Lagrange function associated
421 to the NMPC problem (21). We observe that [10]

$$\nabla_\theta Q_\theta(s, a) = \nabla_\theta \mathcal{L}_\theta(s, y^*) \quad (29)$$

422 holds for y^* given by the primal-dual solution of (23). The gradient (29) is, therefore, straightforward to build as a byproduct
423 of solving the NMPC problem (23). We additionally observe
424 that

$$\nabla_\theta V_\theta(s) = \nabla_\theta \mathcal{L}(s, y^*) \quad (30)$$

426 for y^* given by the primal-dual solution to (21) completed with
427 $\zeta^* = 0$. Finally, the gradient of the NMPC policy with respect
428 to the parameters θ is given by [10]

$$\nabla_\theta \pi_\theta(s) = -\nabla_\theta \xi_\theta(s, y^*) \nabla_y \xi_\theta(s, y^*)^{-1} \frac{\partial y}{\partial u_0} \quad (31)$$

429 for y^* given by the primal-dual solution to (21) with $\zeta^* = 0$, and where $\xi_\theta(s, y)$ gathers the primal-dual KKT conditions
430 underlying the NMPC scheme (21).

432 We remark that (29) and (30) are well defined for any s, θ
433 such that no inequality constraint in the ENMPC schemes (21)
434 and (23), respectively, is weakly active. When an inequality
435 constraint is weakly active, (29) and (30) may be defined only
436 up to the subgradients generated by the possible active sets. This
437 technical issue is fairly straightforward to circumvent in practice
438 by use of interior-point techniques when solving (21) and
439 (23). Additionally, the policy gradient (31) is only valid if the
440 ENMPC (21) satisfies the linear independence constraint qualifi-
441 cation, and the second-order sufficient conditions [10], which
442 are typically satisfied by properly formulated ENMPC schemes.

443 We detail next how TD learning is deployed on the NMPC
444 scheme (23), both in an on and off policy fashion.

445 B. Q-Learning for (E)NMPC

446 A classical approach to Q -learning [36] is based on parameter
447 updates driven by the temporal difference with instantaneous
448 policy updates

$$\tau_k = L_\theta(s_k, a_k) + \gamma V_\theta(s_{k+1}) - Q_\theta(s_k, a_k) \quad (32a)$$

$$\theta \leftarrow \theta + \alpha \tau_k \nabla_\theta Q_\theta(s_k, a_k) \quad (32b)$$

449 where

$$L_\theta(s_k, a_k) = l_\theta(x_k, a_k) + w^\top \max(0, h_\theta(x_k, a_k)) \quad (33)$$

450 and where a_k is selected according to the NMPC policy $\pi_\theta(s)$,
451 with the possible addition of occasional random exploratory
452 moves [36]. The scalar $\alpha > 0$ is a step size commonly used in
453 stochastic gradient-based approaches. A version of Q -learning
454 with batch policy updates reads as

$$\tau_k = L_{\tilde{\theta}}(s_k, a_k) + \gamma V_{\tilde{\theta}}(s_{k+1}) - Q_{\tilde{\theta}}(s_k, a_k) \quad (34a)$$

$$\tilde{\theta} \leftarrow \tilde{\theta} + \alpha \tau_k \nabla_{\tilde{\theta}} Q_{\tilde{\theta}}(s_k, a_k) \quad (34b)$$

455 where a_k is selected according to the fixed NMPC policy $\pi_\theta(s)$,
456 while the learning is performed within an alternative NMPC
457 scheme based on the parameters $\tilde{\theta}$, and not applied to the
458 real system. Note that the on-policy approach (32) entails that
459 changes in the NMPC parameters θ are readily applied in closed

loop after each update (32b), while the off-policy approach (34)
allows one to learn a new set of policy parameters $\tilde{\theta}$ while de-
ploying the original NMPC scheme, based on the parameters
 θ , on the real system. The learned parameters $\tilde{\theta}$ can then be
introduced in closed-loop at convenience, by performing the
replacement $\theta \leftarrow \tilde{\theta}$, after they have converged and after a formal
verification of the corresponding NMPC scheme has been
carried out, see [21], [33].

It ought to be made clear here that RL methods of the type (32) or (34) yield no guarantee to find the global optimum of the parameters. This limitation pertains to most applications of RL relying on nonlinear function approximators, such as the commonly used DNN. In practice, however, RL improves the closed-loop performance over the one of the initial parameters.

C. Role of the Constraints Relaxation

We can now further discuss the ℓ_1 constraints relaxation (21d) in the light of the TD approaches (32) and (34). In the absence of constraints relaxation, the value functions take infinite values when a constraint violation occurs, and they are, therefore, meaningless in the context of RL, as only finite value functions can be used in (32)–(34). The proposed constraint relaxation ensures that the value functions retain finite value even in the presence of constraints violation, such that the RL updates (32)–(34) remain well defined and meaningful. This technical observation has a fairly simple and generic interpretation: Any form of learning is meaningless if infinite penalties are assigned to violating limitations.

In practice, violating some safety-critical constraints may be unacceptable. In that context, avoiding the violation of crucial constraints ought to be prevented from the formulation of the (E)NMPC scheme. Here, robust NMPC techniques are arguably an important tool to avoid such difficulties. The applicability of the proposed theory to the robust (E)NMPC formulations of Section III-A is, therefore, of crucial importance for safety-critical applications. The interplay of robust (E)NMPC with RL and the handling of safety-critical constraints will be the object of future publications.

D. Deterministic Policy Gradient Methods for ENMPC

It is useful to underline here that Q -learning techniques seek the fitting of Q_θ to Q_* under some norm, with the hope that $Q_\theta \approx Q_*$ will result in $\pi_\theta \approx \pi_*$. There is, however, no *a priori* guarantee that the latter approximation holds when the former does. In order to formally maximize the performance of policy π_θ , it is useful to turn to policy gradient methods. For the sake of brevity, we propose to focus on deterministic policy gradient methods here [32], based on the policy gradient equation

$$\nabla_\theta J(\pi_\theta) = \mathbb{E} [\nabla_\theta \pi_\theta(s) \nabla_a Q_{\pi_\theta}(s, a)] \quad (35)$$

where J is the expected closed-loop cost associated to running policy π_θ on the real system, and Q_{π_θ} the corresponding action-value function. Note that the expectation \mathbb{E} is taken over trajectories of the real system subject to policy π_θ . A necessary condition of optimality for policy π_θ is then

$$\nabla_\theta J(\pi_\theta) = 0. \quad (36)$$

511 Deterministic policy gradient methods are often built around the
 512 TD actor-critic approach, based on [32]

$$\begin{aligned} \tau_k &= L(s_k, a_k) + \gamma Q_w(s_{k+1}, \pi_\theta(s_{k+1})) - Q_w(s_k, a_k) \\ w &\leftarrow w + \alpha_w \tau_k \nabla_w Q_w(s_k, a_k) \\ \theta &\leftarrow \theta + \alpha_\theta \nabla_\theta \pi_\theta(s_k) \nabla_a Q_w(s_k, \pi_\theta(s_k)) \end{aligned} \quad (37)$$

513 for some $\alpha_w, \alpha_\theta > 0$ small enough, where $Q_w \approx Q_{\pi_\theta}$ is
 514 an approximation of the corresponding action-value function.
 515 Computationally efficient choices of action-value function
 516 parametrization Q_w in the ENMPC context will be the object of
 517 future publications.

518 V. RL AND STABLE ENMPC

519 The main idea in ENMPC is to optimize performance (defined
 520 by a suitably chosen cost) rather than penalizing deviations from
 521 a given reference. Since the cost is generic, the value function
 522 is not guaranteed to be positive definite and proving stability
 523 becomes challenging. The main idea for proving stability is to
 524 introduce a cost modification, called *rotation*, which does not
 525 modify the optimal solution, but yields a positive-definite value
 526 function, and hence is a Lyapunov function, such that nominal
 527 stability follows.

528 Section II focused on learning the optimal policy even in case
 529 a wrong model $\mathbb{P}[\hat{s}_+ | s, u]$ is used. In this section, we aim at
 530 enforcing nominal stability even if the corresponding stage cost
 531 \hat{L} is indefinite.¹ To this end, in the NMPC cost, we replace \hat{L}
 532 by a newly-defined stage cost \bar{L} , which we force to be positive
 533 definite. We then use a cost modification in order to recover the
 534 correct, possibly indefinite, value, and action-value functions
 535 corresponding to \bar{L} . Throughout the section, we assume that the
 536 value and action-value functions are bounded on some set.

537 In order to show how the cost modification can be imple-
 538 mented by the term λ_θ in (21), we first introduce the proposed
 539 cost modification as a generalization of the standard one, prove
 540 that it does not modify the optimal policy, and discuss how it
 541 is used to learn the optimal value and action-value functions.
 542 Then, we discuss how the cost modification relates to the cost
 543 rotation typically used in ENMPC. Finally, we illustrate how
 544 the cost modification can be used in a parametrized ENMPC
 545 scheme to be used as function approximator within RL.

546 A. Generalized Cost Rotation

547 Consider modifying the cost of the problem according to

$$\bar{L}(s, a) = \hat{L}(s, a) + \Lambda(s, a) - \gamma \mathbb{E} [\Lambda(\hat{s}_+, \hat{\pi}(\hat{s}_+)) | s, a] \quad (38a)$$

$$\bar{V}_f(s) = V_*(s) + \Lambda(s, \hat{\pi}(s)) \quad (38b)$$

548 where $\bar{V}_f(s)$ denotes the rotated terminal cost and where

$$\Lambda(s, a) \geq \Lambda(s, \hat{\pi}(s)), \quad \forall s, a \quad (38c)$$

549 holds. Moreover, we require $\Lambda(s, a)$ to be such that $\bar{L}(s, a)$ is
 550 finite whenever $\hat{L}(s, a)$ is finite.

¹We remark that, in this context, stability is obtained for the model used by NMPC for predictions. Future work will investigate obtaining stability guarantees for the real process $\mathbb{P}[s_+ | s, u]$.

It is important to underline that defining a function $\Lambda(s, a)$ strictly satisfying condition (38c) is in general hard, as it requires knowledge of $\hat{\pi}$. A simpler choice satisfying (38c) with equality is $\Lambda(s, a) = \lambda(s)$, where λ is any function satisfying the boundedness assumption on $\bar{L}(s, a)$.

Theorem 2: The modification (38) preserves the optimal control policy $\hat{\pi}$ corresponding to \hat{L} and the model $\mathbb{P}[\hat{s}_+ | s, u]$. Moreover, the optimal value and action-value functions satisfy

$$\bar{V}_N(s) = \hat{V}_N(s) + \Lambda(s, \hat{\pi}(s)) \quad (39a)$$

$$\bar{Q}_N(s, a) = \hat{Q}_N(s, a) + \Lambda(s, a). \quad (39b)$$

Proof: We will first prove the theorem for $N = 1$ and then proceed by induction. The definition of action value function reads

$$\hat{Q}_1(s, a) = \hat{L}(s, a) + \gamma \mathbb{E} [V_*(\hat{s}_+) | s, a]. \quad (40)$$

Then, we can write

$$\begin{aligned} \hat{Q}_1(s, a) &+ \Lambda(s, a) \\ &= \hat{L}(s, a) + \Lambda(s, a) + \gamma \mathbb{E} [V_*(\hat{s}_+) | s, a] \\ &= \bar{L}(s, a) + \gamma \mathbb{E} [\hat{V}_f(\hat{s}_+) | s, a] = \bar{Q}_1(s, a). \end{aligned}$$

Since by definition $\hat{Q}_1(s, a) > \hat{Q}_1(s, \hat{\pi}(s_+)), \forall a \notin \hat{\pi}(s_+)$ and by construction (38c) holds, we obtain that

$$\hat{Q}_1(s, a) > \bar{Q}_1(s, \hat{\pi}(s_+)), \quad \forall a \notin \hat{\pi}(s_+).$$

Therefore, the optimal policy is preserved and

$$\bar{V}_1(s) = \hat{V}_1(s) + \Lambda(s, \hat{\pi}(s)) = \hat{V}_*(s) + \Lambda(s, \hat{\pi}(s)).$$

We now use bootstrapping to obtain

$$\begin{aligned} \hat{Q}_{k+1}(s, a) &+ \Lambda(s, a) \\ &= \hat{L}(s, a) + \Lambda(s, a) + \gamma \mathbb{E} [V_k(\hat{s}_+) | s, a] \\ &= \bar{L}(s, a) + \gamma \mathbb{E} [\hat{V}_k(\hat{s}_+) | s, a] = \bar{Q}_{k+1}(s, a). \end{aligned}$$

Therefore, cost modification (38) preserves the policy over any horizon N , i.e., $\bar{\pi} = \hat{\pi}$, and (39) hold. ■

We will show next that, for any given policy, the modified NMPC scheme can yield any bounded value function, while this does not hold for the action-value function. Consequently any attempt at learning the optimal policy by solely relying on learning the value function and the proposed modification cannot succeed.

Consider a problem formulated using a stage cost $\check{L}(s, a)$, with the corresponding value functions associated to the optimal policy $\check{\pi}$

$$\check{V}_N(s) = \check{L}(s, \check{\pi}(s)) + \gamma \mathbb{E} [\check{V}_{N-1}(\check{s}_+) | s, \check{\pi}(s)] \quad (41a)$$

$$\check{Q}_N(s, a) = \check{L}(s, a) + \gamma \mathbb{E} [\check{V}_{N-1}(\check{s}_+) | s, a]. \quad (41b)$$

Lemma 1: Consider the set $\check{\mathcal{S}}$ such that for all $s \in \check{\mathcal{S}}$ both $\check{V}_N(s)$ and $\hat{V}_N(s)$ are bounded. Then, there exists a cost modification $\Lambda(s, a) = \lambda(s), \forall a$ such that

$$\check{V}_N(s) + \lambda(s) = \hat{V}_N(s). \quad (42)$$

581 *Proof:* The result is obtained by choosing $\lambda(s) = \hat{V}_N(s) -$
 582 $\check{V}_N(s)$. ■

583 B. Strict Dissipativity

584 Since most of the literature on dissipativity-based ENMPC
 585 focuses on deterministic formulations, in this subsection we
 586 also restrict to the deterministic case and we adopt $\gamma = 1$, as is
 587 usual in the literature on ENMPC. Strict dissipativity is typically
 588 used in ENMPC in order to obtain a positive-definite cost to
 589 construct a Lyapunov function and prove closed-loop stability.
 590 Strict dissipativity holds if there exists a function λ such that

$$\lambda(s_+) - \lambda(s) \leq -\rho(\|s - s_e\|) + L(s, a) - L(s_e, a_e) \quad (43)$$

591 with ρ a positive-definite function and

$$(s_e, a_e) = \underset{s, a}{\operatorname{argmin}} \quad L(s, a) \quad \text{s.t. } s = f(s, a).$$

592 In [4], it has been proven that strict dissipativity is sufficient
 593 for closed-loop stability, provided that λ is bounded and the
 594 constraint set is compact. In [24], it has been proven that, under
 595 a controllability assumption and if (s_e, a_e) lies in the interior of
 596 the constraint set, strict dissipativity with a bounded function λ is
 597 also necessary for closed-loop stability on a compact constraint
 598 set.

599 For simplicity and without loss of generality, we assume that
 600 $L(s_e, a_e) = 0$.

601 *Lemma 2:* If there exists a function $\Lambda(s, a)$ such that $\min_a \Lambda$
 602 $(s, a) = \Lambda(s, \hat{\pi}(s)) = \lambda(s)$, with $\lambda(s)$ satisfying the strict dissi-
 603 pativity inequality (43), then $\bar{L}(s, a) \geq \rho(\|s - s_e\|) \geq 0$ holds.

604 *Proof:* The proof is obtained by noting that

$$\begin{aligned} \bar{L}(s, a) &= L(s, a) + \Lambda(s, a) - \Lambda(\hat{s}_+, \hat{\pi}(s_+)) \\ &\geq L(s, a) + \lambda(s) - \lambda(\hat{s}_+) \geq \rho(\|s - s_e\|) \end{aligned}$$

605 where the first inequality follows from (38c), and the second
 606 inequality is a direct consequence of (43) and $L(s_e, a_e) = 0$. ■

607 Lemma 2 is of paramount importance in the context of EN-
 608 MPC because it establishes that any stabilizing optimal policy
 609 π_\star originating from any given stage cost $L(s, a)$ can be learned
 610 by using a parametrization that yields a positive-definite stage
 611 cost $\bar{L}(s, a)$. Note that the proposed cost modification is a gen-
 612 eralization of the cost rotation used for ENMPC, as we replace
 613 $\lambda(s)$ with $\Lambda(s, a)$.

614 We ought to stress here that though the results presented in
 615 this section pertain to the deterministic case, we expect them
 616 to extend to the stochastic case as well. Unfortunately a mature
 617 dissipativity theory for stochastic ENMPC has not been devel-
 618 oped yet. It would arguably not be surprising if a stochastic
 619 dissipativity criterion can be put in the form

$$\bar{L}(s, a) \geq \rho(\|s - s_e\|)$$

620 with \bar{L} defined in (38) in combination with appropriate termi-
 621 nal conditions, hence offering a generalization of [35]. These
 622 questions will be the object of future research.

623 C. Economic Reward and NMPC Parametrization

624 In the following, we will first show that the proposed cost
 625 modification can be reduced to a cost on the initial state, which

is introduced in the ENMPC scheme (21) as the term λ_θ . Second,
 626 we will explain how we can use a stability-enforcing positive-
 627 definite stage cost to approximate the value and action-value
 628 function of the economic cost.
 629

For simplicity, we use the cost modification $\Lambda(s, a) = \lambda(s)$.
 630 Note that we drop the dependence on the control in Λ for prac-
 631 tical reasons: 1) keeping it would require knowledge of the
 632 optimal policy and 2) $\lambda(s)$ is a valid choice in the sense that it
 633 satisfies (38c).
 634

We observe that the cost modification can be summarized by
 635 the initial cost $\lambda(x_0)$ by relying on the fact that for all predicted
 636 trajectories the modified cost reads as [4], [12]
 637

$$\begin{aligned} &= \gamma^N V^f(x_N) + \sum_{k=0}^{N-1} \gamma^k l(x_k, u_k) + \gamma^k \lambda(x_k) - \gamma^{k+1} \lambda(x_{k+1}) \\ &= \lambda(x_0) + \gamma^N V^f(x_N) + \sum_{k=0}^{N-1} \gamma^k l(x_k, u_k). \end{aligned}$$

In (21), we parametrize the cost modification as $\lambda_\theta(x_0)$

We now turn to the question of enforcing stability in the
 639 parametrized ENMPC scheme. In the literature on EMPC the
 640 cost modification is used to prove stability in case the stage
 641 cost is not positive definite. In our case, we are interested in
 642 the opposite: We would like to use a positive-definite stage cost
 643 to enforce stability while the cost modification is used in order
 644 to obtain value and action-value functions corresponding to the
 645 economic (nonpositive definite) cost.
 646

Thanks to Theorem 1 and Lemma 2, introducing the term λ_θ
 647 in (21) guarantees that if the optimal policy π_\star is stabilizing for
 648 the model dynamics (21b), then the ENMPC scheme (21) can
 649 deliver the optimal policy π_\star and value functions V_\star, Q_\star with
 650 a stage cost l_θ lower bounded by \mathcal{K}_∞ .
 651

VI. ANALYTICAL CASE STUDY: THE LQR CASE

One of the few cases where (26) can be constructed and sat-
 653 isfied exactly is the LQR case. We use it as a simple illustration
 654 of the meaning of \hat{L} , Assumption (16) and the cost modification
 655 $\Lambda(s, a)$. Consider a centered linear quadratic Gaussian control
 656 problem with the true dynamics and stage cost
 657

$$s_+ = As + Ba + e, \quad e \sim \mathcal{N}(0, \Sigma) \quad (44)$$

$$L(s, a) = \begin{bmatrix} s \\ a \end{bmatrix}^\top \begin{bmatrix} T & N \\ * & R \end{bmatrix} \begin{bmatrix} s \\ a \end{bmatrix}. \quad (45)$$

The associated value functions, if they exist, read as

$$V_\star = s^\top S s + V_0$$

$$Q_\star = \begin{bmatrix} s \\ a \end{bmatrix}^\top \begin{bmatrix} T + \gamma A^\top S A & N + \gamma A^\top S B \\ * & R + \gamma B^\top S B \end{bmatrix} \begin{bmatrix} s \\ a \end{bmatrix} + V_0 \quad (46)$$

where $V_0 = \gamma(1 - \gamma)^{-1} \operatorname{Tr}(S\Sigma)$. Matrix S and the associated
 659 optimal policy K_\star are delivered by the Schur complement of
 660 the quadratic form in Q_\star , i.e., the discounted LQR equations
 661

$$T + \gamma A^\top S A = S + (N + \gamma A^\top S B) K_\star \quad (47a)$$

$$(R + \gamma B^\top S B) K_\star = N^\top + \gamma B^\top S A. \quad (47b)$$

662 *A. LQR With Imperfect Model and \hat{L}*

663 We now consider the deterministic model

$$\hat{s}_+ = \hat{A}s + \hat{B}a. \quad (48)$$

664 In order to verify Theorem 1, we consider the stage cost

$$\begin{aligned} \hat{L}(s, a) &= Q_*(s, a) - \gamma V_*(\hat{s}_+) \\ &= \begin{bmatrix} s \\ a \end{bmatrix}^\top \begin{bmatrix} \hat{T} & \hat{N} \\ * & \hat{R} \end{bmatrix} \begin{bmatrix} s \\ a \end{bmatrix} + (1 - \gamma) V_0. \end{aligned}$$

665 This implies that \hat{T} , \hat{N} , \hat{R} must satisfy

$$\hat{T} + \gamma \hat{A}^\top S \hat{A} = T + \gamma A^\top S A \quad (49a)$$

$$\hat{N} + \gamma \hat{A}^\top S \hat{B} = N + \gamma A^\top S B \quad (49b)$$

$$\hat{R} + \gamma \hat{B}^\top S \hat{B} = R + \gamma B^\top S B. \quad (49c)$$

666 The resulting value function reads as $\hat{V}(s) = s^\top \hat{S}s + \hat{V}_0$,
667 where \hat{S} and the associated policy \hat{K} satisfy

$$\hat{T} + \gamma \hat{A}^\top \hat{S} \hat{A} = \hat{S} + (\hat{N} + \gamma \hat{A}^\top \hat{S} \hat{B}) \hat{K} \quad (50a)$$

$$(\hat{R} + \gamma \hat{B}^\top \hat{S} \hat{B}) \hat{K} = \hat{N}^\top + \gamma \hat{B}^\top \hat{S} \hat{A}. \quad (50b)$$

668 Using (49), we obtain that matrices $\hat{S} = S$ and $\hat{K} = K_*$ sat-
669 isfy (50).670 *B. Assumption (16) in the LQR Case*671 In order for the discrete algebraic Riccati equation
672 (DARE) (50) to deliver a valid LQR solution \hat{S} , \hat{K} (in the
673 sense of a Bellman optimality backup), Assumption (16) needs
674 to be satisfied. In the case $\gamma = 1$, this requires that $\hat{A} - \hat{B}\hat{K}_*$
675 has all its eigenvalues inside the unit circle. We illustrate this
676 fact by the following simple example:

$$A = 1, \quad B = 1, \quad T = 1, \quad R = 2, \quad N = 0$$

677 which yields $S = 2$ and $K_* = 0.5$. By using the model $\hat{A} = 2$,
678 $\hat{B} = 1$, we obtain

$$\hat{T} = -5, \quad \hat{R} = 2, \quad \hat{N} = -2$$

679 such that $\hat{A} - \hat{B}\hat{K}_* = 1.5$ and K_* corresponds to the nonsta-
680 bilizing solution $\hat{S} = 2$ of the DARE. Note, however, that the
681 DARE does have a stabilizing solution, which reads

$$\hat{S} = 7, \quad \hat{K} = 4/3 \quad \text{such that} \quad \hat{A} - \hat{B}\hat{K} = 2/3.$$

682 *C. Economic LQR and Cost Modification $\Lambda(s, a)$* 683 We now consider an economic LQR, for which $\hat{L}(s, a)$ is
684 indefinite. We introduce matrices δT , δN , δR to define

$$\Lambda(s, a) := \begin{bmatrix} s \\ a \end{bmatrix}^\top \begin{bmatrix} \delta T & \delta N \\ \delta N^\top & \delta R \end{bmatrix} \begin{bmatrix} s \\ a \end{bmatrix}$$

685 and observe that $\delta N = K_*^\top \delta R$, $\delta R \succeq 0$ must hold in order for
686 (38c) to be fulfilled, where K_* is the LQR controller gain asso-
687 ciated to the stage cost L for the true system (47b). By defining $\delta S := \delta T - K_*^\top \delta R K_*$, we obtain

$$\mathbb{E} [\Lambda(s_+, \hat{\pi}(\hat{s}_+)) | s, a] = \begin{bmatrix} s \\ a \end{bmatrix}^\top \begin{bmatrix} \hat{A}^\top \delta S \hat{A} & \hat{A}^\top \delta S \hat{B} \\ \hat{B}^\top \delta S \hat{A} & \hat{B}^\top \delta S \hat{B} \end{bmatrix} \begin{bmatrix} s \\ a \end{bmatrix}$$

such that admissible stage cost modifiers are based on the
689 quadratic forms
690

$$\begin{aligned} \delta W_L &= \begin{bmatrix} K_*^\top \delta R K_* & K_*^\top \delta R \\ \delta R K_* & \delta R \end{bmatrix} - \begin{bmatrix} \gamma \hat{A}^\top \delta S \hat{A} - \delta S & \gamma \hat{A}^\top \delta S \hat{B} \\ \gamma \hat{B}^\top \delta S \hat{A} & \gamma \hat{B}^\top \delta S \hat{B} \end{bmatrix} \\ &= \delta W_L^0 + \delta W_L^1. \end{aligned}$$

It follows that the modified action-value function reads

$$\hat{Q}(s, a) = \begin{bmatrix} s \\ a \end{bmatrix}^\top (W + \delta W_L) \begin{bmatrix} s \\ a \end{bmatrix}. \quad (51)$$

Since we are interested in stability-enforcing schemes, one
692 needs to choose δR and δS such that $W + \delta W_L \succ 0$ holds.
693 Since this expression is linear in δR and δS , the problem
694 amounts to solving an LMI.
695In the following, we state explicitly how the two contributions
696 in δW_L relate to conditions (38c) and (43) respectively. The term
697 δW_L^1 can be framed as a quadratic stage cost rotation and the
698 condition
699

$$\delta W_L^1 = - \begin{bmatrix} \gamma \hat{A}^\top \delta S \hat{A} - \delta S & \gamma \hat{A}^\top \delta S \hat{B} \\ \gamma \hat{B}^\top \delta S \hat{A} & \gamma \hat{B}^\top \delta S \hat{B} \end{bmatrix} \succ 0$$

is the strict dissipativity condition (43) for the linear-quadratic
700 case [42]. For $\delta R = 0$, we obtain $\Lambda(s, a) = \lambda(s) = s^\top \delta S s$.
701The term δW_L^0 resembles the stage cost used in [11], [41] and
702 satisfies
703

$$\Lambda^0(s, a) = \begin{bmatrix} s \\ a \end{bmatrix}^\top \delta W_L^0 \begin{bmatrix} s \\ a \end{bmatrix} = (a + K_* s)^\top \delta R (a + K_* s)$$

such that $\Lambda^0(s, a) > \Lambda^0(s, -K_* s)$, $\forall a \neq -K_* s$, $\forall s$. As dis-
704 cussed in [11], [41], the use of $\Lambda^0(s, a)$ as stage cost with a
705 zero terminal cost yields a scheme, which delivers the optimal
706 feedback for the nominal model. The related value function and
707 action-value function, however, are zero.
708

VII. NUMERICAL EXAMPLES

In this section, we propose two examples in order to illustrate
710 the theoretical developments.
711

A. Linear MPC

We first consider a simple linear MPC example to illustrate
713 the methods mentioned above. We consider the MPC scheme
714

$$\min_{x, u} V_0 + \frac{\gamma^N}{2} x_N^\top S_N x_N + \sum_{k=0}^{N-1} f^\top \begin{bmatrix} x_k \\ u_k \end{bmatrix} \quad (52a)$$

$$\sum_{k=0}^{N-1} \frac{1}{2} \gamma^k \left(\|x_k\|^2 + \frac{1}{2} \|u_k\|^2 + w^\top s_k \right) \quad (52b)$$

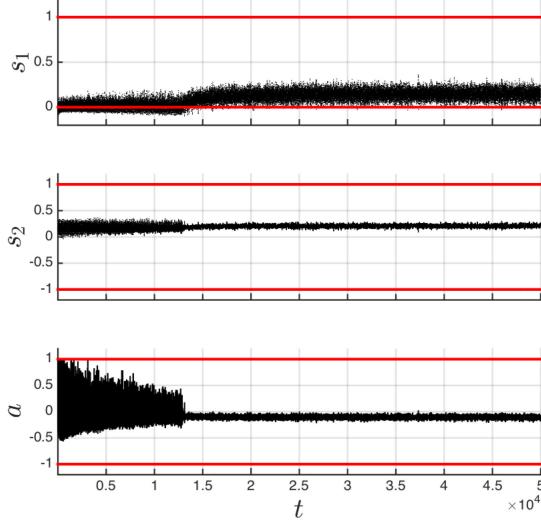


Fig. 1. State and input trajectories for the example detailed in Section VII-A.

$$715 \quad \text{s.t.} \quad x_{k+1} = Ax_k + Bu_k + b \quad (52c)$$

$$\begin{bmatrix} 0 \\ -1 \end{bmatrix} + \underline{x} - s_k \leq x_k \leq \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \bar{x} + s_k \quad (52d)$$

$$-1 \leq u_k \leq 1, \quad x_0 = s \quad (52e)$$

716 where the NMPC parameters subject to the RL scheme are

$$\theta = (V_0, \underline{x}, \bar{x}, b, f, A, B) \quad (53)$$

717 and where the ℓ_1 relaxation uses the weight $w^\top = [10^2 \ 10^2]$.
 718 We selected a discount factor $\gamma = 0.9$. The terminal cost matrix
 719 S_N is selected as the Riccati matrix underlying the LQR con-
 720 troller locally equivalent to the MPC scheme. The objective of
 721 the MPC scheme is to drive the states to the origin, such that the
 722 MPC reference is activating the lower bound of the first state. A
 723 horizon of $N = 10$ is used. The MPC model is initially chosen
 724 as

$$A = \begin{bmatrix} 1 & 0.25 \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0.0312 \\ 0.25 \end{bmatrix} \quad (54)$$

725 while all other parameters are initialized with zero values.

726 We will consider that the “real” process is following the dy-
 727 namics

$$x_{k+1} = \begin{bmatrix} 0.9 & 0.35 \\ 0 & 1.1 \end{bmatrix} x_k + \begin{bmatrix} 0.0813 \\ 0.2 \end{bmatrix} u_k + \begin{bmatrix} e_k \\ 0 \end{bmatrix} \quad (55)$$

728 where e_k is a random, uncorrelated, uniformly distributed vari-
 729 able in the interval $[-10^{-1}, 0]$, and therefore drives the first
 730 state to violate its lower bound.

731 We use the on-policy algorithm (32) without introducing ex-
 732 ploration. The step size was selected as $\alpha = 10^{-6}$. Fig. 1 shows
 733 the resulting state and input trajectories. Fig. 2 shows the adap-
 734 tation of the MPC parameters via the on-policy algorithm (32).
 735 Fig. 3 shows the evolution of the stage cost $L(s, a)$ (including
 736 the large ℓ_1 penalties for the constraints violations), and the
 737 evolution of the TD error (32a). It can be observed that the RL

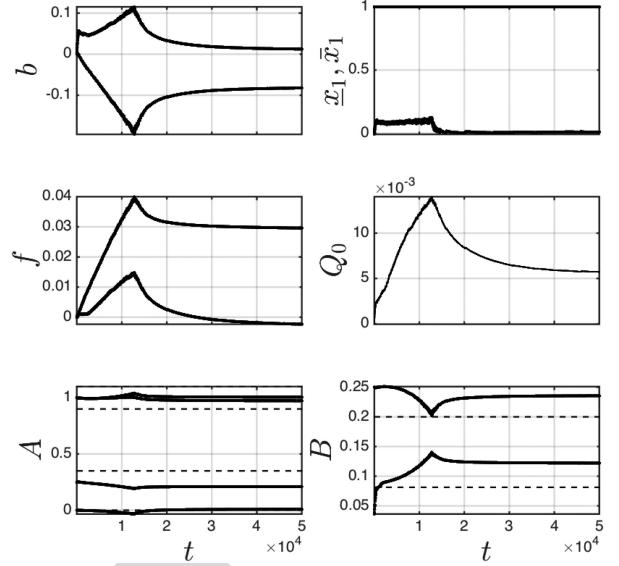


Fig. 2. Trajectory of the parameters θ for the example detailed in Section VII-A. The dashed lines report the “real system” A and B entries. One can observe that the RL scheme does not perform system identification, as the MPC model does not converge to the “real system.”

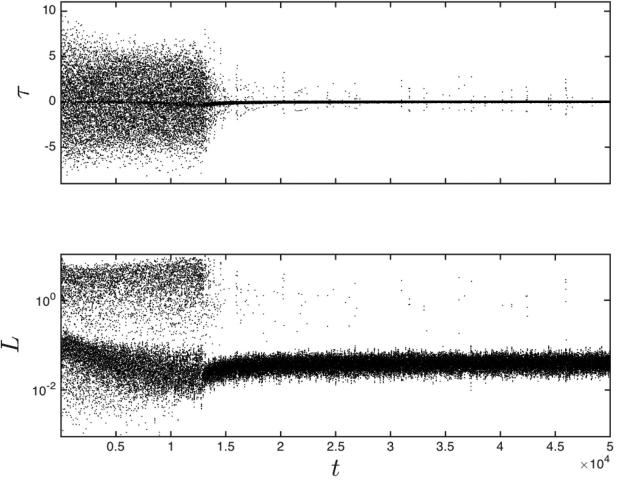


Fig. 3. Evolution of the stage cost (52b) and TD error (32a) achieved for the example detailed in Section VII-A.

algorithm manages to reduce the TD error τ to small values, 738 averaging to zero. The state trajectories often violate the state 739 bound $x_1 \geq 0$ in the beginning, resulting in large control actions 740 (see Fig. 1), but the RL adjusts the MPC parameters in order to 741 avoid these expensive violations. The adaptation of the param- 742 eters is a combination of modifying the MPC model bias b (mostly on 743 the first state, subject to the process noise).

We have also deployed this example without letting the RL 744 scheme adapt the model parameters. The MPC performance is 745 then increased mostly via tightening the bounds, and does not 746 reach the performance shown in Fig. 2. For the sake of brevity, 747 we do not report these results here.

TABLE I
MODEL PARAMETERS

<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	<i>g</i>
0.5616	0.3126	48.43	0.507	55	0.1538	55
<i>h</i>	<i>M</i>	<i>C</i>	<i>U_{A₂}</i>	<i>C_p</i>	λ	λ_s
0.16	20	4	6.84	0.07	38.5	36.6
<i>F₁</i>	<i>X₁</i>	<i>F₃</i>	<i>T₁</i>	<i>T₂₀₀</i>		
10	5 %	50	40	25		

The units are omitted and are consistent with the physical quantities they correspond to.

B. Evaporation Process

We consider an example from the process industry, i.e., the evaporation process modeled in [34], [39] and used in [5], [42] in the context of economic MPC. The model equations include states (X_2, P_2) (concentration and pressure) and controls (P_{100}, F_{200}) (pressure and flow). The model further depends on concentration X_1 , flow F_2 , and temperatures T_1, T_{200} , which are assumed to be constant in the control model. In reality, these quantities are stochastic with variance $\sigma_{X_1} = 1$, $\sigma_{F_1} = 2$, $\sigma_{T_1} = 8$, $\sigma_{T_{200}} = 5$, and mean centered on the nominal value. Bounds $(25, 40) \leq (X_2, P_2) \leq (100, 80)$ on the states and $u_l = (100, 100) \leq (P_{100}, F_{200}) \leq (400, 400) = u_u$ on the controls are present. In particular, the bound $X_2 \geq 25$ is introduced in order to ensure sufficient quality in the product. All state bounds are relaxed and translated into cost terms as in Section IV-A. The system dynamics are given by [5]

$$M\dot{X}_2 = F_1 X_1 - F_2 X_2, \quad C\dot{P}_2 = F_4 - F_5 \quad (56)$$

where

$$\begin{aligned} T_2 &= aP_2 + bX_2 + c, & T_3 &= dP_2 + e \\ \lambda F_4 &= Q_{100} - F_1 C_p (T_2 - T_1), & T_{100} &= fP_{100} + g \\ Q_{100} &= U_{A_1} (T_{100} - T_2), & U_{A_1} &= h(F_1 + F_3) \\ Q_{200} &= \frac{U_{A_2} (T_3 - T_{200})}{1 + U_{A_2} / (2C_p F_{200})}, & F_{100} &= \frac{Q_{100}}{\lambda_s} \\ \lambda F_5 &= Q_{200}, & F_2 &= F_1 - F_4 \end{aligned}$$

and the model parameters are given in Table I. The economic objective is given by

$$L(x, u) = 10.09(F_2 + F_3) + 600F_{100} + 0.6F_{200}.$$

We introduce functions $\lambda_\theta, V_\theta^f, l_\theta$ as fully parametrized quadratic functions defined by Hessian H_\dagger , gradient h_\dagger and constant $c_\dagger, \dagger = \{\lambda, V^f, l\}$ to formulate the ENMPC controller

$$\begin{aligned} \min_z \quad & \lambda_\theta(x_0) + \gamma^N (V_\theta^f(x_N) + w^\top \sigma_f) \\ & + \sum_{k=0}^{N-1} \gamma^k (l_\theta(x_k, u_k) + w^\top \sigma_k) \\ \text{s.t. } & x_{k+1} = f_\theta(x_k, u_k), \quad x_0 = s \\ & g(u_k) \leq 0, \quad h_\theta(x_k) \leq \sigma_k. \end{aligned}$$

The model is parametrized as the nominal model with the addition of a constant, i.e., $f_\theta(x, u) = f(x, u) + c_f$. The control

constraints are fixed and the state constraints are parametrized as simple bounds, i.e., $h(x) = (x - x_l, x_u - x)$. The vector of parameter, therefore, reads as

$$\theta = (H_\lambda, h_\lambda, c_\lambda, H_{V^f}, h_{V^f}, c_{V^f}, H_l, h_l, c_l, c_f, x_l, x_u).$$

Constants $w = 1$ are fixed and assumed to reflect the known cost of violating the state constraints.

We use the batch policy update (32) with $\alpha = 10^{-4}$ and we update the parameters with the learned ones every $N_{\text{upd}} = 2000$ time steps. In order to induce enough exploration, we use an ϵ -greedy policy which is greedy 90% of the samples, while in the remaining 10%, we perturb the optimal feedback as

$$a = \text{sat}(u_0^* + e, u_l, u_u), \quad e \sim \mathcal{N}(0, 1)$$

where $\text{sat}(\cdot, u_l, u_u)$ saturates the input between its lower and upper bounds $u_l = (100, 100), u_u = (400, 400)$, respectively. We initialize the ENMPC scheme by tuning the cost using the economic-based approach proposed in [42], which is based on the nominal model. For the model, we use $c_f = 0$ and the bounds are initialized at their nominal values. While at every step we do check that $H_l, H_{V^f} \succ 0$, during the learning phase the parameters never violate this constraint. As shown in Fig. 4, the algorithm converges to a constant parameter value while reducing the average TD error.

We ran a closed-loop simulation to compare the performance of RL-tuned NMPC with the one using the economic-based tuning proposed in [42]. We show in Fig. 5 that the difference in cost and concentration X_2 between the two ENMPC schemes. It can be seen that RL keeps X_2 at higher values in order to reduce the violation of the quality constraint. This entails an improvement in the cost, which is about 7% in the considered scenario.

It is important to stress here that both the cost and the model parametrization do not have the same structure as the real cost and model. Therefore, the action-value function and, consequently, the policy can be learned only approximately. Another important remark concerns the role of the storage function λ_θ . RL exploits this function to keep the stage cost positive definite. If we invert the cost modification (38) in order to recover an approximation of the economic stage cost, we obtain an indefinite one.

We ran an additional simulation in which the model used in simulations is deterministic and coincides with the one used for predictions in NMPC. We initialize the learning phase with the nominally tuned parameters from [42]. In this case, RL keeps the parameters essentially unaltered, which suggests that the nominal tuning was already optimal.

Consider the naive initial guess $H_l = I, x_l = (25, 40), x_u = (100, 80)$, while all other parameters are 0. In order to obtain parameters, which are close to convergence in each batch, we set $N_{\text{upd}} = 20000$. In this case, with 10^5 samples we did not yet obtain convergence. Even though convergence of the RL scheme is much slower, the parameters are approaching a steady value and the closed-loop performance is improved with respect to the initial guess.

While our simulation results are promising, we ought to specify here that we have observed some potential

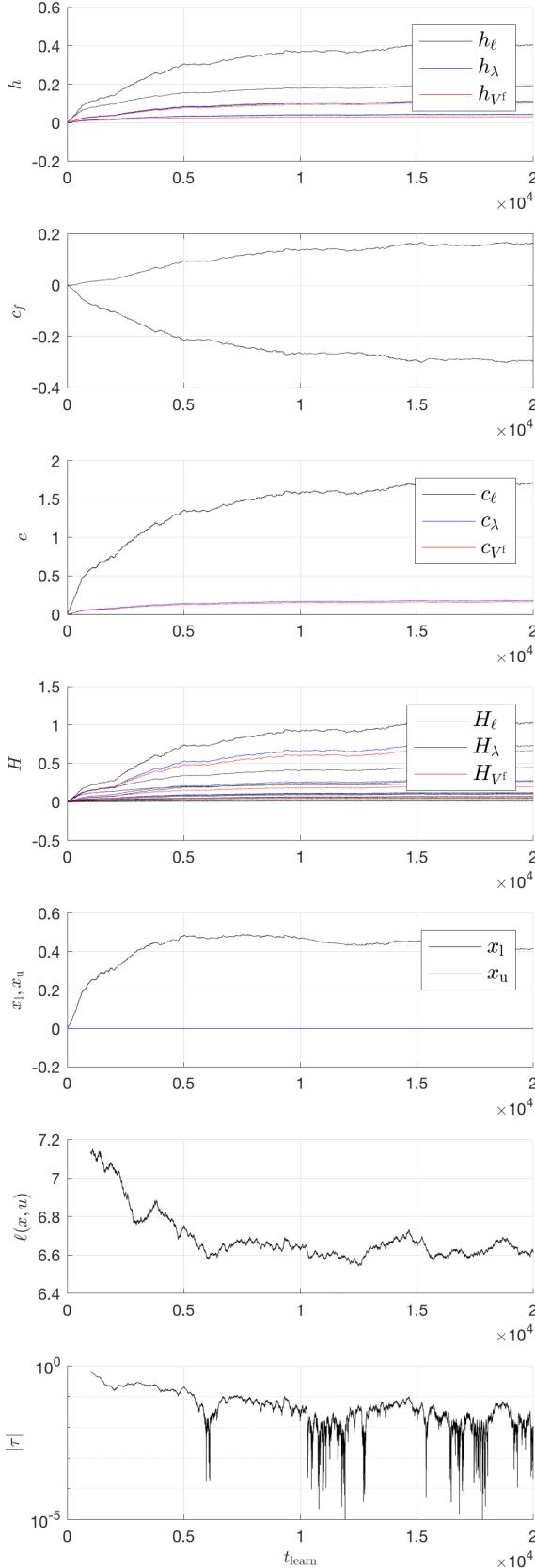


Fig. 4. Evolution of the parameters (increment w.r.t. the initial guess value) and of the TD error (averaged over the preceding 1000 samples) during learning.

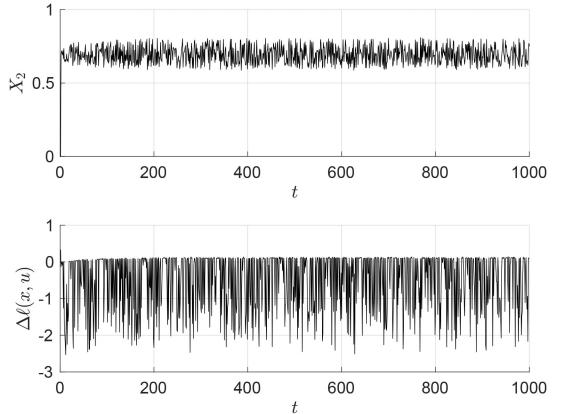


Fig. 5. Closed-loop simulations: Difference between the RL-tuned NMPC, and the nominal economic tuning [42]. Top plot: Difference in concentration X_2 . Bottom plot: Difference in economic cost.

difficulties related—to the best of our knowledge—to using a gradient method in adjusting the ENMPC parameters and using a Q -learning method as a proxy for learning the optimal policy. These observations arguably call for exploring the effectiveness of deploying second-order methods, such as LSTD-type methods [36], and policy gradient approaches [32] to alleviate the difficulties observed in some cases.

VIII. CONCLUSION

In this paper, we propose to use ENMPC schemes to support the parametrization of the value functions and/or the policy, which is an essential component of RL. We show that the ENMPC schemes can generate the optimal policy for the real system even if the underlying model is wrong, by adjusting the stage and terminal cost alone. We also show how a positive stage and terminal cost can be used in the ENMPC scheme to learn the optimal control policy, resulting in an ENMPC scheme that is stable by construction. The resulting ENMPC delivers the optimal control policy for the real system if that policy is itself stabilizing. We additionally detail how some classic RL methods can be deployed in practice to adjust the parameters of the ENMPC scheme. The methods are illustrated in simulations.

Future work will propose improvements in the RL algorithms specific for ENMPC, and propose an efficient combination of the existing classic model-tuning techniques and the RL-based tuning.

REFERENCES

- [1] P. Abbeel, A. Coates, M. Quigley, and A. Y. Ng, “An application of reinforcement learning to aerobatic helicopter flight,” in *Advances in Neural Information Processing Systems 19*. Cambridge, MA, USA: MIT Press, 2007, p. 2007.
- [2] M. Agarwal, “Feasibility of on-line reoptimization in batch processes,” *Chem. Eng. Commun.*, vol. 158, no. 1, pp. 19–29, 1997.
- [3] M. Agarwal, “Iterative set-point optimization of batch chromatography,” *Comp. Chem. Eng.*, vol. 29, no. 6, pp. 1401–1409, 2005.
- [4] R. Amrit, J. Rawlings, and D. Angeli, “Economic optimization using model predictive control with a terminal cost,” *Annu. Rev. Control*, vol. 35, pp. 178–186, 2011.

- [5] R. Amrit, J. B. Rawlings, and L. T. Biegler, "Optimizing process economics online using model predictive control," *Comput. Chem. Eng.*, vol. 58, pp. 334–343, 2013.
- [6] D. Angeli, R. Amrit, and J. Rawlings, "On average performance and stability of economic model predictive control," *IEEE Trans. Autom. Control*, vol. 57, no. 7, pp. 1615–1626, Jul. 2012.
- [7] A. Aswani, H. Gonzalez, S. S. Sastry, and C. Tomlin, "Provably safe and robust learning-based model predictive control," *Automatica*, vol. 49, no. 5, pp. 1216–1226, 2013.
- [8] F. Berkenkamp, M. Turchetta, A. Schoellig, and A. Krause, "Safe model-based reinforcement learning with stability guarantees," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., New York, NY, USA: Curran Associates, Inc., 2017, pp. 908–918.
- [9] D. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1, 3rd ed. Belmont, MA, USA: Athena Scientific, 2005.
- [10] C. Büskens and H. Maurer, "Sensitivity analysis and real-time optimization of parametric nonlinear programming problems," *Online Optimization of Large Scale Systems*. Berlin Heidelberg, Germany: Springer, 2001, pp. 3–16.
- [11] L. Chisci, J. A. Rossiter, and G. Zappa, "Systems with persistent disturbances: Predictive control with restricted constraints," *Automatica*, vol. 37, pp. 1019–1028, 2001.
- [12] M. Diehl, R. Amrit, and J. B. Rawlings, "A Lyapunov function for economic optimizing model predictive control," *IEEE Trans. Autom. Control*, vol. 56 no. 3, pp. 703–707, Mar. 2011.
- [13] J. F. Forbes and T. E. Marlin, "Design cost: A systematic approach to technology selection for model-based real-time optimization systems," vol. 20, no. 67, pp. 717–734, 1996.
- [14] J. F. Forbes, T. E. Marlin, and J. F. MacGregor, "Model adequacy requirements for optimizing plant operations," *Comp. Chem. Eng.*, vol. 18, no. 6, pp. 497–510, 1994.
- [15] W. Gao and S. Engell, "Comparison of iterative set-point optimisation strategies under structural plant-model mismatch," vol. 38, pp. 12–17, 2005.
- [16] L. Grüne and J. Pannek, *Nonlinear Model Predictive Control*. London, U.K.: Springer, 2011.
- [17] L. Hewing, A. Liniger, and M. N. Zeilinger, "Cautious NMPC with Gaussian process dynamics for miniature race cars," 2017, arXiv:1711.06586.
- [18] J. Fernandez and J. Garcia, "A comprehensive survey on safe reinforcement learning," *J. Mach. Learn. Res.*, vol. 16, pp. 1437–1480, 2013.
- [19] J. Peters, J. Kober, and J. A. Bagnell, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, pp. 1238–1274, 2013.
- [20] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, "Learning-based model predictive control for safe exploration and reinforcement learning," 2018, arXiv:1803.08287.
- [21] J. Löfberg, "Oops! I cannot do it again: Testing for recursive feasibility in MPC," *Automatica*, vol. 48, no. 3, pp. 550–555, 2012.
- [22] A. Marchetti, B. Chachuat, and D. Bonvin, "Modifier-adaptation methodology for real-time optimization," *Ind. Eng. Chem. Res.*, vol. 48, no. 13, pp. 6022–6033, 2009.
- [23] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 26, no. 6, pp. 789–814, 2000.
- [24] M. A. Müller, D. Angeli, and F. Allgöwer, "On necessity and robustness of dissipativity in economic model predictive control," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1671–1676, Jun. 2015.
- [25] C. J. Ostafew, A. P. Schoellig, and T. D. Barfoot, "Robust constrained learning-based NMPC enabling reliable mobile robot path tracking," *Int. J. Robot. Res.*, vol. 35, no. 13, pp. 1547–1563, 2016.
- [26] J. B. Rawlings and R. Amrit, "Optimizing process economic performance using model predictive control," in *Proc. IFAC Conf. Nonlinear Model Predictive Control Pavia*, 2009, pp. 119–138.
- [27] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Madison, WI, USA: Nob Hill, 2009.
- [28] P. D. Roberts, "An algorithm for steady-state system optimization and parameter estimation," *Int. J. Syst. Sci.*, vol. 10, no. 7, pp. 719–734, 1979.
- [29] P. D. Roberts, "Coping with model-reality differences in industrial process optimisation, a review of integrated system optimisation and parameter estimation (isope)," *Comput. Industry*, vol. 26, no. 3, pp. 281–290, 1995.
- [30] P. O. M. Scokaert and J. B. Rawlings, "Feasibility issues in linear model predictive control," *AICHE J.*, vol. 45, no. 8, pp. 1649–1659, 1999.
- [31] D. Silver *et al.*, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, pp. 484–503, 2016.
- [32] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proc. 31st Int. Conf. Int. Conf. Mach. Learn.*, vol. 32, 2014, pp. I–387–I–395.
- [33] D. Simon and J. Löfberg, "Stability analysis of model predictive controllers using mixed integer linear programming," in *IEEE 55th Conference Decis. Control*, 2016, pp. 7270–7275.
- [34] C. Sonntag, O. Stursberg, and S. Engell, "Dynamic optimization of an industrial evaporator using graph search with embedded nonlinear programming," in *Proc. 2nd IFAC Conf. Anal. Des. Hybrid Syst.*, 2006, pp. 211–216.
- [35] P. Sopasakis, D. Herceg, P. Patrinos, and A. Bemporad, "Stochastic economic model predictive control for Markovian switching systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 524–530, 2017.
- [36] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.
- [37] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Proc. 12th Int. Conf. Neural Inform. Process.*, 1999, pp. 1057–1063.
- [38] P. Tatjewski, "Iterative optimizing set-point control—the basic principle redesigned," vol. 35, pp. 49–54, 2002.
- [39] F. Y. Wang and I. T. Cameron, "Control studies on a model evaporation process—Constrained state driving with conventional and higher relative degree systems," *J. Process Control*, vol. 4, pp. 59–75, 1994.
- [40] S. Wang, W. Chaovallitwongse, and R. Babuska, "Machine learning algorithms in bipedal robot control," *Trans. Syst. Man Cybern. Part C*, vol. 42, no. 5, pp. 728–743, Sep. 2012.
- [41] M. Zanon, T. Charalambous, H. Wymeersch, and P. Falcone, "Optimal scheduling of downlink communication for a multi-agent system with a central observation post," *IEEE Control Syst. Lett.*, vol. 2, no. 1, pp. 37–42, Jan. 2018.
- [42] M. Zanon, S. Gros, and M. Diehl, "A tracking MPC formulation that is locally equivalent to economic MPC," *J. Process Control*, vol. 45, pp. 30–42, 2016.

Sébastien Gros received the Ph.D. degree from École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, in 2007.



He was part of an R&D group hosted at Strathclyde University, Glasgow, U.K., focusing on wind turbine control. In 2011, he joined the university of KU Leuven, Leuven, Belgium, where his main research focus was on optimal control and fast NMPC for complex mechanical systems. He joined the Department of Signals and Systems at Chalmers University of Technology, Göteborg, Sweden, in 2013, where he became Associate Professor in 2017. He is now Full Professor with Norwegian University of Science and Technology, Norway, and Affiliate Professor with Chalmers. His research interests include numerical methods, real-time optimal control, reinforcement learning, and the optimal control of energy-related applications.

Mario Zanon received the master's degree in mechatronics from the University of Trento, Trento, Italy, and the Diplôme d'Ingénieur degree from the Ecole Centrale Paris, Paris, France, in 2010. After research stays at the KU Leuven, Leuven, Belgium, University of Bayreuth, Bayreuth, Germany, Chalmers University, Göteborg, Sweden, and the University of Freiburg, Freiburg, Germany, he received the Ph.D. degree in electrical engineering from the KU Leuven in November 2015.



He held a Postdoc Researcher Position with Chalmers University until the end of 2017 and is now an Assistant Professor with the IMT School for Advanced Studies Lucca, Lucca, Italy. His research interests include numerical methods for optimization, economic MPC, reinforcement learning, and the optimal control and estimation of nonlinear dynamic systems, in particular for aerospace and automotive applications.

GENERAL INSTRUCTION

- **Authors:** We cannot accept new source files as corrections for your paper. If possible, please annotate the PDF proof we have sent you with your corrections and upload it via the Author Gateway. Alternatively, you may send us your corrections in list format. You may also upload revised graphics via the Author Gateway.

Queries

- Q1. Author: We have made a number of changes to the wording of the paper, English grammar, and readability. Please check the whole text for correctness and please suggest any improvements.
- Q2. Author: Please confirm or add details for any funding or financial support for the research of this article.
- Q3. Author: Please provide the expansion of LQR, LMI, and MPC at first mention.
- Q4. Author: Please check Ref. [15] for correctness.
- Q5. Author: Please provide the subject in which Sébastien Gros received the Ph.D. degree.
- Q6. Author: Please check the sentence “He was part...” for correctness.
- Q7. Author: Please provide the subject in which Mario Zanon received the Diplôme d’Ingénieur degree.

Data-Driven Economic NMPC Using Reinforcement Learning

Sébastien Gros¹ and Mario Zanon¹

Abstract—Reinforcement learning (RL) is a powerful tool to perform data-driven optimal control without relying on a model of the system. However, RL struggles to provide hard guarantees on the behavior of the resulting control scheme. In contrast, nonlinear model predictive control (NMPC) and economic NMPC (ENMPC) are standard tools for the closed-loop optimal control of complex systems with constraints and limitations, and benefit from a rich theory to assess their closed-loop behavior. Unfortunately, the performance of (E)NMPC hinges on the quality of the model underlying the control scheme. In this paper, we show that an (E)NMPC scheme can be tuned to deliver the optimal policy of the real system even when using a wrong model. This result also holds for real systems having stochastic dynamics. This entails that ENMPC can be used as a new type of function approximator within RL. Furthermore, we investigate our results in the context of ENMPC and formally connect them to the concept of dissipativity, which is central for the ENMPC stability. Finally, we detail how these results can be used to deploy classic RL tools for tuning (E)NMPC schemes. We apply these tools on both, a classical linear MPC setting and a standard nonlinear example, from the ENMPC literature.

Index Terms—Adaptive nonlinear model predictive control (NMPC), economic NMPC (ENMPC), reinforcement learning, strict dissipativity.

I. INTRODUCTION

R EINFORCEMENT learning (RL) is a powerful tool for tackling Markov decision processes (MDP) without depending on a model of the probability distributions underlying the state transitions. Indeed, most RL methods rely purely on observed state transitions, and realizations of the stage cost in order to increase the performance of the control policy. RL has drawn an increasing attention thanks to its striking accomplishments ranging from computers beating Chess and Go masters [31], to robot learning to walk or fly without supervision [1], [40].

Manuscript received October 16, 2018; revised April 8, 2019; accepted April 16, 2019. Date of publication; date of current version. Recommended by Associate Editor M. Alamir. (Corresponding author: Mario Zanon.)

S. Gros is with the Department of Engineering Cybernetics, Norwegian University of Science and Technology, 7491 Trondheim, Norway (e-mail: sebastien.gros@ntnu.no).

M. Zanon is with the IMT School for Advanced Studies 55100 Lucca, Italy (e-mail: mario.zanon@imtlucca.it).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TAC.2019.2913768

Most RL methods are based on learning the optimal control policy for the real system either directly, or indirectly. Indirect methods rely on learning an approximation of the optimal action-value function underlying the MDP, typically using variants of temporal-difference learning [19]. Since the action-value function is, in general, unknown *a priori*, a generic function approximator is typically used to approximate it. A common choice in the RL community is to use a deep neural network (DNN).

Direct RL methods seek to learn the optimal policy directly. Most direct RL methods are based on the stochastic or deterministic policy gradient methods, see [32], [37]. Both rely on carrying an approximation of the action-value function, or at least of the value function underlying the policy. Similar to indirect methods, direct RL methods also typically use DNNs to approximate the optimal policy and the associated (action) value function.

Unfortunately, the closed-loop behavior of a system subject to an approximate optimal policy supported by a DNN or a generic function approximation can be difficult to formally analyze. It can therefore be difficult to generate certificates of the behavior of a system controlled by a generic RL algorithm. This issue is especially salient when dealing with safety-critical systems. The development of safe RL methods, which aims at tackling this issue, is an open field of research [18].

Nonlinear model predictive control (NMPC) is a formal control method based on solving at every time instant an optimal control problem to generate the optimal control policy. The optimal control problem seeks to minimize a sum of stage costs over a prediction horizon, subject to state trajectories provided by a model of the real system, the current observed state of the real system, and possibly state and input constraints to be respected. The optimal control problem then delivers an entire input and state sequence spanning the prediction horizon. However, only the first input is applied to the real system. At the next time instant, the entire optimal control problem is solved again using a new estimation of the state of the system.

If the system model underlying the NMPC scheme is perfect, and with the addition of an adequate terminal cost, the NMPC scheme delivers the optimal control policy. Classic NMPC is based on a stage cost lower bounded by \mathcal{K}_∞ functions, while economic NMPC (ENMPC) accepts a generic stage cost [4], [12], [26]. A rich and mature theory exists in the literature to analyze the properties of classic NMPC when operating in closed loop with a real system, establishing desirable key properties, such as recursive feasibility and stability [16], [23], [27].

ENMPC has recently attracted the attention of the research community, and a stability theory has been developed fairly recently [4], [6], [12], [24], [26], but is arguably still under development. Since it can tackle the system constraints directly and since it benefits from a rich theory analyzing its behavior, (E)NMPC is arguably an ideal candidate for safety-critical applications.

Unfortunately, the performance of (E)NMPC schemes relies on having a good model of the system to be controlled. A data-driven adaption of the NMPC model to better fit the real system is a fairly obvious approach to tackle this issue. However, since the model does not necessarily match the real system, fitting the model to the data does not necessarily result in the (E)NMPC scheme delivering the optimal policy, and can even be counterproductive. Some attempts have been recently proposed to tackle this problem, such as in [17].

The problem of optimizing a system based on a model having the wrong structure is well known in the field of real-time optimization [2], [3], [13], [14], [22], and has been addressed via the modifier approach [15], [22], [28], [29], [38], whereby the cost function of the optimization problem is adapted rather than the model.

In this paper, we propose to use (E)NMPC schemes instead of DNNs to support the parametrization to approximate the (action) value functions and the policy. Similarly to the idea originated in RTO, we show that the NMPC scheme can deliver the optimal control policy even if the underlying model is incorrect, by adapting the stage cost, terminal cost, and constraints only. This observation is applicable to any NMPC scheme, such as classic NMPC, ENMPC, robust and stochastic ENMPC. Furthermore, we establish strong connections between this cost adaptation and the concept of strict dissipativity, which is fundamental to the stability theory of ENMPC.

One practical outcome of the theory proposed in this paper is that all RL techniques can be directly used to tune the NMPC scheme to increase its performance on the real system. Because the theory proposed is very generic, this observation holds for a stochastic system being controlled by an (E)NMPC scheme based on a deterministic model or a robust NMPC scheme using a scenario tree. Another practical outcome of the theory proposed in this paper is that if a stage cost attached to a given system yields a stabilizing optimal control policy, then an ENMPC scheme with a positive stage cost can, in principle, be tuned to deliver the optimal policy. A last practical outcome of the proposed theory is that using (E)NMPC, as a parametrization for RL instead of DNN, allows one to use the rich theory underlying (E)NMPC schemes in the context of RL, and deliver certificates on the behavior of the policy resulting from the learning process.

The combination of learning and control techniques has been proposed in [7], [8], [20], [25]. To the best of our knowledge, however, our paper is the first work proposing to use NMPC as a function approximator in RL and investigating the connection between RL and economic MPC.

The paper is structured as follows. Section II establishes the fundamental result of the paper, showing that an (E)NMPC scheme based on the wrong model can, under some conditions, nonetheless deliver the optimal control policy. Section III details

how these results can be used in practice. Section IV details how some classic RL techniques can be deployed to adjust the (E)NMPC parameters. Section V further develops the theory and details its connection to the fundamental concept of strict dissipativity underlying the stability theory of ENMPC, and details its consequences for using ENMPC as a parametrization for RL. Section VI deploys the theory in the case of LQR with Gaussian noise, for which all the objects discussed in the theory can be built explicitly, and their practical meaning assessed. Section VII proposes some illustrative examples.

II. OPTIMAL POLICY BASED ON AN INEXACT MODEL

We will consider that the real system we want to control is described by a discrete MDP having the (possibly) stochastic state transition dynamics

$$\mathbb{P}[s_+ | s, a] \quad (1)$$

where s, a is the current state-input pair and s_+ is the subsequent one. We will label $L(s, a)$ the stage cost associated to the MDP, possibly infinite for some state-input pairs s, a , which, we will assume, can take the form

$$L(s, a) = l(s, a) + \mathcal{I}_\infty(h(s, a)) + \mathcal{I}_\infty(g(a)) \quad (2)$$

where we use the indicator function

$$\mathcal{I}_\infty(x) = \begin{cases} \infty & \text{if } x_i > 0 \text{ for some } i \\ 0 & \text{otherwise} \end{cases}. \quad (3)$$

In (2), function l captures the cost given to different state-input pairs, while the constraints

$$g(a) \leq 0, \quad h(s, a) \leq 0 \quad (4)$$

capture undesirable state and inputs, and infinite values are given to L when (4) is violated. Note that we have separated pure input constraints and mixed constraints for reasons that will be made clear later on.

With the addition of a discount factor $0 < \gamma \leq 1$, (1) and (2) yield the optimal policy $\pi_\star(s)$. Note that the notation (1) is standard in the literature on MDPs, while the control literature typically uses the notation $s_+ = f(s, a, \zeta)$, where ζ is a stochastic variable and f a possibly nonlinear function. The action-value function Q_\star and value function V_\star associated with the MDP are defined by the Bellman equations [9]

$$Q_\star(s, a) = L(s, a) + \gamma \mathbb{E}[V_\star(s_+) | s, a] \quad (5a)$$

$$V_\star(s) = Q_\star(s, \pi_\star(s)) = \min_a Q_\star(s, a). \quad (5b)$$

Throughout the paper we will assume that the MDP underlying the real system, the associated stage cost L , and the discount factor γ yield a well-posed problem, i.e., the value functions defined by (5) are well posed, and finite over some sets.

We then consider a model of the real system having the state transition dynamics

$$\mathbb{P}[\hat{s}_+ | s, a] \quad (6)$$

which typically do not match (1) perfectly. Note that (6) trivially includes deterministic models as a special case. Consider a stage

183 cost defined as

$$\hat{L}(s, a) = \begin{cases} Q_*(s, a) - \gamma V^+(s, a) & \text{if } |V^+(s, a)| < \infty \\ \infty & \text{otherwise} \end{cases} \quad (7)$$

184 where $V^+(s, a) = \mathbb{E}[V_*(\hat{s}_+) | s, a]$ and where the expectation
185 is taken over the distribution (6). It is useful here to specify that
186 the conditional formulation of the stage cost \hat{L} proposed in (7)
187 is a technicality dedicated to having a well-defined \hat{L} even when
188 both the action value function and value function take infinite
189 values.

190 We establish next the central theorem of this paper, stating
191 that under some conditions, the optimal policy π_* that minimizes
192 the stage cost L for the true dynamics (1) is also generated by
193 using model (6) combined with the stage cost \hat{L} . Hence it is
194 possible to generate the optimal policy based on a wrong model
195 by modifying the stage cost. It may be useful to specify here that
196 our approach further in the paper will be to bypass the possibly
197 difficult evaluation of (7), and replace it by learning \hat{L} directly
198 from the data, see Section IV.

199 *Theorem 1:* Consider the optimal value function

$$\hat{V}_N(s) = \min_{\pi} \mathbb{E} \left[\gamma^N V_*(\hat{s}_N^\pi) + \sum_{k=0}^{N-1} \gamma^k \hat{L}(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right] \quad (8)$$

200 associated to the stage cost (7), the state transition model (6), and
201 the terminal cost $V_*(s)$ over an optimization horizon N . Here
202 we define $\hat{s}_{0,\dots,N}^\pi$ as the (possibly stochastic) trajectories of the
203 state transition model $\mathbb{P}[\hat{s}_+ | s, a]$ under a policy π , starting
204 from $\hat{s}_0^\pi = s$. We will label $\hat{\pi}$ as the optimal policy associated
205 to $\hat{V}_N(s)$ and $\hat{Q}_N(s, a)$ the associated action-value functions.
206 Consider the set \mathcal{S} such that

$$|\mathbb{E}[V_*(\hat{s}_k^\pi)]| < \infty, \quad \forall s \in \mathcal{S}, \quad \forall k. \quad (9)$$

207 Then the following identities hold on \mathcal{S} .

- 208 i) $\hat{V}_N(s) = V_*(s)$.
- 209 ii) $\hat{\pi}(s) = \pi_*(s)$.
- 210 iii) $\hat{Q}_N(s, a) = Q_*(s, a)$ for the inputs a such that
211 $|\mathbb{E}[V_*(\hat{s}_+)]| < \infty$.

212 *Proof:* Let us consider the N -step value function \hat{V}_N^π asso-
213 ciated to the stage cost \hat{L} , the state transition model $\mathbb{P}[\hat{s}_+ | s, a]$
214 and a policy π , defined as

$$\hat{V}_N^\pi(s) = \mathbb{E} \left[\gamma^N V_*(\hat{s}_N^\pi) + \sum_{k=0}^{N-1} \gamma^k \hat{L}(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right]. \quad (10)$$

215 Assumption (9) ensures that, at least for $\pi = \pi_*$, all the terms
216 in the sum in (10) have a finite expected value for $s \in \mathcal{S}$, such
217 that $\hat{V}_N^\pi(s)$ is well defined and finite over \mathcal{S} for some policies
218 π . Using a telescopic sum, we can rewrite (10) as

$$\hat{V}_N^\pi(s) = Q_*(s, \pi(s)) + \mathbb{E} \left[\sum_{k=1}^{N-1} \gamma^k A_*(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right] \quad (11)$$

219 where the advantage function A_* is defined as

$$A_*(s, a) = \begin{cases} Q_*(s, a) - V_*(s) & \text{if } |Q_*(s, a)| < \infty \\ \infty & \text{otherwise} \end{cases}. \quad (12)$$

Using the Bellman equalities

$$\min_{\pi} Q_*(s, \pi(s)) = Q_*(s, \pi_*(s)) = V_*(s) \quad (13a)$$

$$\min_{\pi} A_*(s, \pi(s)) = A_*(s, \pi_*(s)) = 0 \quad (13b)$$

we observe that all terms in (11) are minimized by the policy
221 π_* , such that the following equalities hold on \mathcal{S} :

$$\hat{V}_N(s) = \min_{\pi} \hat{V}_N^\pi(s) = \hat{V}_N^{\pi_*}(s) = V_*(s) \quad (14)$$

where the first equality holds by definition, the second holds
223 because π_* is the minimizer of \hat{V}_N^π , and the last equality holds
224 from (13). It follows that $\hat{V}_N(s) = V_*(s)$ holds on \mathcal{S} for any
225 N , hence the identities (i) and (ii) hold. We furthermore observe
226 that on \mathcal{S} and for any input a such that $|\mathbb{E}[V_*(\hat{s}_+)]| < \infty$,
227 the following equalities hold:

$$\begin{aligned} \hat{Q}_N(s, a) &= \hat{L}(s, a) + \gamma \mathbb{E} [\hat{V}_{N-1}(\hat{s}_+) | s, a] \\ &= Q_*(s, a) + \gamma \mathbb{E} [\hat{V}_{N-1}(\hat{s}_+) - V_*(\hat{s}_+) | s, a] \\ &= Q_*(s, a) \end{aligned} \quad (15)$$

which yields statement (iii). ■ 229

Note that if the transition model is exact, i.e., $\mathbb{P}[\hat{s}_+ | s, a] =$
230 $\mathbb{P}[s_+ | s, a]$, then the stage cost defined in (7) satisfies $\hat{L}(s, a) =$
231 $L(s, a), \forall s \in \mathcal{S}$, with \mathcal{S} defined in (9). It is interesting to discuss
232 here the case $N \rightarrow \infty$ for which, under some conditions, the
233 terminal cost can be dismissed. We detail this in the following
234 Corollary. 235

Corollary 1: Under the assumptions of Theorem 1 and under
236 the additional assumption

$$\lim_{N \rightarrow \infty} \mathbb{E} [\gamma^N V_*(\hat{s}_N^\pi)] = 0 \quad (16)$$

then

$$\hat{V}_\infty(s) = \lim_{N \rightarrow \infty} \min_{\pi} \mathbb{E} \left[\sum_{k=0}^{N-1} \gamma^k \hat{L}(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right] = V_*(s) \quad (17)$$

and the equalities $\hat{\pi}(s) = \pi_*(s)$ and $\hat{Q}_\infty(s, a) = Q_*(s, a)$ hold
239 for $|\mathbb{E}[V_*(\hat{s}_+)]| < \infty$. 240

Proof: Let us consider the N -step value function \hat{V}_N^π asso-
241 ciated to the stage cost \hat{L} , the state transition model $\mathbb{P}[\hat{s}_+ | s, a]$
242 and a policy π defined as

$$\hat{V}_\infty^\pi(s) = \lim_{N \rightarrow \infty} \mathbb{E} \left[\sum_{k=0}^{N-1} \gamma^k \hat{L}(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right] \quad (18a)$$

$$\begin{aligned} &= Q_*(s, \pi(s)) + \lim_{N \rightarrow \infty} \mathbb{E} \left[-\gamma^N V_*(\hat{s}_N^\pi) \right. \\ &\quad \left. + \sum_{k=1}^{N-1} \gamma^k A_*(\hat{s}_k^\pi, \pi(\hat{s}_k^\pi)) \right] \end{aligned} \quad (18b)$$

where (18b) holds as, by assumption, all terms in the summa-
244 tions in (18) have finite expected values. This entails that

$$\hat{V}_\infty^{\pi_*}(s) = \lim_{N \rightarrow \infty} V_*(s) + \mathbb{E} \left[-\gamma^N V_*(\hat{s}_N^{\pi_*}) \right]. \quad (19)$$

246 Using (16), we finally observe that the following inequalities
 247 hold on S :

$$\hat{V}_\infty(s) = \min_{\pi} \hat{V}_\infty^\pi(s) = \hat{V}_\infty^{\pi_*}(s) = V_*(s) \quad (20)$$

248 and are justified similarly to (14). \blacksquare

- 249 Let us next make a few observations regarding Theorem 1.
- 250 1) Assumption (9) requires that the model trajectories under
 251 policy π_* are contained within the set where the value
 252 function V_* is finite with a unitary probability.
 - 253 2) Assumption (16) can be construed as some form of
 254 stability condition on the model dynamics under pol-
 255 icy π_* . This observation is especially clear in the case
 256 $\gamma = 1$, which then imposes the condition $\lim_{N \rightarrow \infty} \mathbb{E}[V_*(\hat{s}_N^\pi)] = 0$.
 - 257 3) Condition (16) is not required in Theorem 1, however,
 258 it is highly desirable to fulfil it in practice when a finite
 259 horizon is required, so that the terminal cost in (8) has a
 260 limited impact on the optimal policy $\hat{\pi}$.
 - 261 4) Though very similar, Theorem 1 and Corollary 1 cover
 262 different cases, since taking the limit for $N \rightarrow \infty$ in
 263 Theorem 1 does not require Assumption (16) to hold.
 - 264 5) Theorem 1 proposes a modified stage cost and a termi-
 265 nal cost such that the finite-horizon problem (8) delivers
 266 the optimal policy π_* . It ought to be noted that prob-
 267 lem (8) would actually use $\pi_*(\hat{s}_k)$ to select the inputs at
 268 every stage k of the prediction $\hat{s}_{0,\dots,N-1}$. However, iden-
 269 tities (i)–(iii) of Theorem 1 do not necessarily require this
 270 restriction, i.e., it is sufficient to find a stage cost and a
 271 terminal cost such that the resulting optimal control prob-
 272 lem yields π_* as its *initial* policy (at stage $k = 0$) only to
 273 get identities (i)–(iii).

275 III. ENMPC AS A FUNCTION APPROXIMATOR

276 We will now detail how the theory presented above applies
 277 to using a parametrized ENMPC scheme to approximate the
 278 optimal policy and value functions π_* and V_* , Q_* , even if the
 279 model underlying the ENMPC scheme is not highly accurate.
 280 We will detail in Section IV how the ENMPC parameters can
 281 be adjusted to achieve this approximation.

282 The constraints (4) can be used in the ENMPC scheme to
 283 explicitly exclude undesirable states and inputs. Since the EN-
 284 MPC scheme will be based on an imperfect model f_θ , it will
 285 seek the minimization of the modified stage cost $\hat{L}(s, a)$ rather
 286 than the original one $L(s, a)$. As a result, while the pure input
 287 constraints $g(a) \leq 0$ are arguably fixed, the mixed constraints
 288 used in the ENMPC scheme ought to be modified as well, in
 289 order to capture the domain where $\hat{L}(s, a)$ is finite. Since \hat{L} is
 290 not known *a priori*, the domain where it is finite depends, among
 291 other things, on the discrepancy between f_θ and (1), and will
 292 have to be learned. We will therefore consider introducing the
 293 parametrized mixed constraints $h_\theta(s, a) \leq 0$ in the ENMPC
 294 scheme, where θ will be parameters that can be adjusted via
 295 RL tools. Equation (26) and Corollary 2 below provide a more
 296 formal explanation of these observations.

297 Since RL tools cannot handle infinite penalties, we will need
 298 to consider a relaxed version of L and of the mixed constraints

h_θ . We can now formulate the parametrized ENMPC scheme
 299 that will serve as a function approximation in the RL tools. \blacksquare

300 We will consider a parametrization of the value function V_*
 301 using the following ENMPC scheme parametrized by θ : \blacksquare

$$V_\theta(s) = \min_{u, x, \sigma} \lambda_\theta(x_0) + \gamma^N (V_\theta^f(x_N) + w_f^\top \sigma_N) \quad (21a)$$

$$+ \sum_{k=0}^{N-1} \gamma^k (l_\theta(x_k, u_k) + w^\top \sigma_k) \quad (21a)$$

$$\text{s.t. } x_{k+1} = f_\theta(x_k, u_k), \quad x_0 = s \quad (21b)$$

$$g(u_k) \leq 0 \quad (21c)$$

$$h_\theta(x_k, u_k) \leq \sigma_k, \quad h_\theta^f(x_N) \leq \sigma_N. \quad (21d)$$

302 Problem (21) is a classic ENMPC formulation when $\gamma = 1$
 303 and $\lambda_\theta = 0$ [16], [27]. Note that we have used x, u in order
 304 to clearly distinguish the ENMPC prediction from the actual
 305 closed-loop state and control trajectory. \blacksquare

306 We observe that the ENMPC scheme (21) holds a model
 307 parametrization f_θ , a constraint parametrization h_θ as discussed
 308 above, and a parametrization of the stage cost l_θ and terminal
 309 cost V_θ^f . The extra cost λ_θ is discussed in detail in Section V-
 310 A. Reasonable guesses for these functions are to use $l_\theta = l$,
 311 $h_\theta = h$, and any classic heuristic to build the terminal cost
 312 approximation V_θ^f , such as a quadratic cost stemming from the
 313 LQR approximation of the ENMPC scheme. RL tools are then
 314 used to modify these initial guesses toward higher closed-loop
 315 performances. \blacksquare

316 The ℓ_1 relaxation of the mixed constraints (21d) relying on
 317 slack variables σ_k is fairly standard in practical implementa-
 318 tion of (EN)MPC schemes. For w, w_f large enough, the solution
 319 to (21) is identical to the unrelaxed one whenever a feasible
 320 trajectory exists for the initial state s [30]. In this case, we refer
 321 to the relaxation as *exact*. The specific role of the relaxation will
 322 be detailed in Section IV-C. Function λ_θ in (21a) is not required
 323 anywhere in the following developments, but will play a central
 324 role in forming nominal stability guarantees of the ENMPC
 325 scheme (21), see Section V. We define the policy
 326

$$\pi_\theta(s) = u_0^* \quad (22)$$

327 where u_0^* is the first element of the input sequence u_0^*, \dots, u_{N-1}^*
 328 solution of (21) for a given s . We additionally define the action-
 329 value function Q_θ

$$Q_\theta(s, a) = \min_{u, x} \quad (21a) \quad (23a)$$

$$\text{s.t. } (21b) - (21d) \quad (23b)$$

$$u_0 = a. \quad (23c)$$

330 Note that the proposed parametrization trivially satisfies the
 331 fundamental equalities underlying the Bellman equations, i.e.,
 332

$$\pi_\theta(s) = \arg \min_a Q_\theta(s, a), \quad V_\theta(s) = \min_a Q_\theta(s, a). \quad (24)$$

333 Let us make some key observations on (21) and (22). The
 332 deterministic model (21b) can be construed as a special case of
 333

334 the stochastic state transition (6), using

$$\mathbb{P}[\hat{s}_+ | s, a] = \delta(\hat{s}_+ - f(s, a)) \quad (25)$$

335 where δ is the Dirac distribution. Moreover, suppose that we
336 can select θ such that $\lambda_\theta = 0$ and such that the stage cost and
337 constraints in (21) satisfy

$$\hat{L}(s, a) = l_\theta(s, a) + \mathcal{I}_\infty(h_\theta(s, a)) + \mathcal{I}_\infty(g_\theta(a)) \quad (26a)$$

$$V_*(s) = V_\theta^f(s) + \mathcal{I}_\infty(h_\theta^f(s)) \quad (26b)$$

338 for \hat{L} given by (7), where $l_\theta, V_\theta^f < \infty$.

339 *Corollary 2 (of Theorem 1):* Assume that the NMPC
340 scheme (21) is parametrized using a rich enough parametrization
341 with an exact relaxation (i.e., w, w_f large enough). Then,
342 the NMPC scheme (21) delivers the optimal policy π_* and
343 value functions V_*, Q_* for any state s for which assumption
344 (9) is satisfied.

345 *Proof:* By assumption, there exists θ such that (26) holds,
346 then Theorem 1 directly yields the desired result. ■

347 Note that assumption (9) entails that \mathcal{S} is the forward-invariant
348 set of the dynamics (21b) under the policy π_* , associated to the
349 condition $|V_*(s)| < \infty$.

350 Unfortunately, a parameter θ satisfying (26) is clearly not
351 guaranteed to exist, and does not exist in most nontrivial practical
352 cases. Indeed, the modified stage cost \hat{L} defined by (7)
353 can be highly intricate, and satisfying (26) can require a very
354 elaborate parametrization of the ENMPC cost and constraints.
355 Even if a θ satisfying (26) does exist, finding it is arguably
356 highly difficult as evaluating (7) can be extremely demanding
357 and requires the knowledge of the real stochastic state transition
358 (6). In this paper, we propose to circumvent this difficulty by
359 1) relying on a limited parametrization of the ENMPC scheme,
360 at the price of not achieving $\pi_\theta = \pi_*$ exactly, and 2) deploying
361 RL techniques in order to adjust the ENMPC parameters θ , so
362 as to avoid computing \hat{L} altogether (see Section IV). In the RL
363 context, we ought to consider the ENMPC scheme (21) as a
364 function approximator for V_*, Q_* , and π_* .

365 The choice of cost parametrization is clearly important, but
366 beyond the scope of this paper. In the example below (see
367 Section VII), we have made fairly trivial choices and obtained
368 interesting results nonetheless. It is clear, however, that a rich
369 parametrization in theory is desirable. It is also desirable that the
370 stage cost and terminal cost approximations l_θ and V_θ^f always
371 remain (quasi)convex in order to facilitate the computation of
372 the NMPC solution. In contrast, the parametrization of the initial
373 cost λ_θ can be unrestricted. In the future, we will investigate
374 rich function approximations, including positive sums of convex
375 functions and sum-of-square approaches.

376 A. Robust NMPC Using Scenario Trees

377 Robust NMPC implemented via scenario tree approaches also
378 readily fits in the framework presented here. Indeed, the scenario
379 tree can be construed as a stochastic process with a discrete
380 probability distribution. More specifically, the stochastic model

state transition for a scenario tree dynamics reads as

$$\mathbb{P}[\hat{s}_+ | s, a] = \sum_{i=1}^{N_s} W_i \delta(\hat{s}_+ - f_i(s, a)) \quad (27)$$

382 where f_{1,\dots,N_s} are the N_s different dynamics underlying the
383 scenario tree, and W_{1,\dots,N_s} are the associated probabilities,
384 with $W_i \geq 0$ and $\sum_{i=1}^{N_s} W_i = 1$. All the observations made in
385 Section III are then also valid in this context. One ought then to
386 see the discrete probability distribution w_i as part of the param-
387 eters to be adjusted in the (E)NMPC scheme. 387

B. Model Parametrization

389 As detailed in Section III, the ENMPC scheme (21) can, in
390 principle, capture the optimal policy π_* without having to adjust
391 the model (21b). This observation, however, does not preclude
392 an adaptation of the model (21b) in order to drive the NMPC pol-
393 icy π_θ toward the optimal one π_* , and one can easily argue that
394 allowing such an adaptation introduces additional freedom for
395 the NMPC scheme (21) to better approximate the optimal policy
396 π_* . The interplay between the cost and constraints adaptation
397 and the model adaptation is the object of current research. 397

IV. RL FOR ENMPC

399 Theorem 1 guarantees that it is, in theory, possible to generate
400 the optimal policy and value functions using an ENMPC scheme
401 based on a possibly inaccurate model. In practice, one has to
402 rely on ad hoc parametrization θ of the NMPC scheme, yielding
403 the value function $Q_\theta(s, a)$, $V_\theta(s)$ and the policy $\pi_\theta(s)$. The
404 goal is then to adjust the parameters θ such that the ENMPC
405 scheme policy fits the optimal policy as closely as possible. 405

406 Because computing \hat{L} given by (7) is difficult and requires the
407 knowledge of the true dynamics, we will rely on RL techniques
408 to adjust the parameters θ . We will focus here on classical RL
409 approaches. These techniques typically require the sensitivities
410 of the value functions. We briefly detail next how to compute
411 these sensitivities for (21)–(23).

A. Sensitivities of the ENMPC Scheme

412 We detail next how to evaluate the gradients of functions
413 $Q_\theta, V_\theta, \pi_\theta$. To that end, let us define the Lagrange function
414 associated to the ENMPC Problem (23) as
415

$$\begin{aligned} \mathcal{L}_\theta(s, y) = & \lambda_\theta(x_0) + \gamma^N V_\theta^f(x_N) + \chi_0^\top (x_0 - s) + \mu_N^\top h_\theta^f(x_N) \\ & + \sum_{k=0}^{N-1} \chi_{k+1}^\top (f_\theta(x_k, u_k) - x_{k+1}) + \nu_k^\top g_\theta(u_k) \\ & + \gamma^k L_\theta(x_k, u_k) + \mu_k^\top h_\theta(x_k, u_k) + \zeta^\top (u_0 - a) \end{aligned}$$

416 where χ, μ, ν, ζ are the multipliers associated to constraints
417 (21b)–(21d), and (23b) and (23c), respectively, and we will
418 note $y = (x, u, \chi, \mu, \nu, \zeta)$ the primal-dual variables associated
419 to (23).

420 Note that, for $\zeta = 0$, $\mathcal{L}_\theta(s, y)$ is the Lagrange function associated
421 to the NMPC problem (21). We observe that [10]

$$\nabla_\theta Q_\theta(s, a) = \nabla_\theta \mathcal{L}_\theta(s, y^*) \quad (29)$$

422 holds for y^* given by the primal-dual solution of (23). The gradient (29) is, therefore, straightforward to build as a byproduct
423 of solving the NMPC problem (23). We additionally observe
424 that

$$\nabla_\theta V_\theta(s) = \nabla_\theta \mathcal{L}(s, y^*) \quad (30)$$

426 for y^* given by the primal-dual solution to (21) completed with
427 $\zeta^* = 0$. Finally, the gradient of the NMPC policy with respect
428 to the parameters θ is given by [10]

$$\nabla_\theta \pi_\theta(s) = -\nabla_\theta \xi_\theta(s, y^*) \nabla_y \xi_\theta(s, y^*)^{-1} \frac{\partial y}{\partial u_0} \quad (31)$$

429 for y^* given by the primal-dual solution to (21) with $\zeta^* =$
430 0, and where $\xi_\theta(s, y)$ gathers the primal-dual KKT conditions
431 underlying the NMPC scheme (21).

432 We remark that (29) and (30) are well defined for any s, θ
433 such that no inequality constraint in the ENMPC schemes (21)
434 and (23), respectively, is weakly active. When an inequality
435 constraint is weakly active, (29) and (30) may be defined only
436 up to the subgradients generated by the possible active sets. This
437 technical issue is fairly straightforward to circumvent in practice
438 by use of interior-point techniques when solving (21) and
439 (23). Additionally, the policy gradient (31) is only valid if the
440 ENMPC (21) satisfies the linear independence constraint qualifi-
441 cation, and the second-order sufficient conditions [10], which
442 are typically satisfied by properly formulated ENMPC schemes.

443 We detail next how TD learning is deployed on the NMPC
444 scheme (23), both in an on and off policy fashion.

445 B. Q-Learning for (E)NMPC

446 A classical approach to Q -learning [36] is based on parameter
447 updates driven by the temporal difference with instantaneous
448 policy updates

$$\tau_k = L_\theta(s_k, a_k) + \gamma V_\theta(s_{k+1}) - Q_\theta(s_k, a_k) \quad (32a)$$

$$\theta \leftarrow \theta + \alpha \tau_k \nabla_\theta Q_\theta(s_k, a_k) \quad (32b)$$

449 where

$$L_\theta(s_k, a_k) = l_\theta(x_k, a_k) + w^\top \max(0, h_\theta(x_k, a_k)) \quad (33)$$

450 and where a_k is selected according to the NMPC policy $\pi_\theta(s)$,
451 with the possible addition of occasional random exploratory
452 moves [36]. The scalar $\alpha > 0$ is a step size commonly used in
453 stochastic gradient-based approaches. A version of Q -learning
454 with batch policy updates reads as

$$\tau_k = L_{\tilde{\theta}}(s_k, a_k) + \gamma V_{\tilde{\theta}}(s_{k+1}) - Q_{\tilde{\theta}}(s_k, a_k) \quad (34a)$$

$$\tilde{\theta} \leftarrow \tilde{\theta} + \alpha \tau_k \nabla_{\tilde{\theta}} Q_{\tilde{\theta}}(s_k, a_k) \quad (34b)$$

455 where a_k is selected according to the fixed NMPC policy $\pi_\theta(s)$,
456 while the learning is performed within an alternative NMPC
457 scheme based on the parameters $\tilde{\theta}$, and not applied to the
458 real system. Note that the on-policy approach (32) entails that
459 changes in the NMPC parameters θ are readily applied in closed

loop after each update (32b), while the off-policy approach (34) allows one to learn a new set of policy parameters $\tilde{\theta}$ while deploying the original NMPC scheme, based on the parameters θ , on the real system. The learned parameters $\tilde{\theta}$ can then be introduced in closed-loop at convenience, by performing the replacement $\theta \leftarrow \tilde{\theta}$, after they have converged and after a formal verification of the corresponding NMPC scheme has been carried out, see [21], [33].

It ought to be made clear here that RL methods of the type (32) or (34) yield no guarantee to find the global optimum of the parameters. This limitation pertains to most applications of RL relying on nonlinear function approximators, such as the commonly used DNN. In practice, however, RL improves the closed-loop performance over the one of the initial parameters.

C. Role of the Constraints Relaxation

We can now further discuss the ℓ_1 constraints relaxation (21d) in the light of the TD approaches (32) and (34). In the absence of constraints relaxation, the value functions take infinite values when a constraint violation occurs, and they are, therefore, meaningless in the context of RL, as only finite value functions can be used in (32)–(34). The proposed constraint relaxation ensures that the value functions retain finite value even in the presence of constraints violation, such that the RL updates (32)–(34) remain well defined and meaningful. This technical observation has a fairly simple and generic interpretation: Any form of learning is meaningless if infinite penalties are assigned to violating limitations.

In practice, violating some safety-critical constraints may be unacceptable. In that context, avoiding the violation of crucial constraints ought to be prevented from the formulation of the (E)NMPC scheme. Here, robust NMPC techniques are arguably an important tool to avoid such difficulties. The applicability of the proposed theory to the robust (E)NMPC formulations of Section III-A is, therefore, of crucial importance for safety-critical applications. The interplay of robust (E)NMPC with RL and the handling of safety-critical constraints will be the object of future publications.

D. Deterministic Policy Gradient Methods for ENMPC

It is useful to underline here that Q -learning techniques seek the fitting of Q_θ to Q_* under some norm, with the hope that $Q_\theta \approx Q_*$ will result in $\pi_\theta \approx \pi_*$. There is, however, no *a priori* guarantee that the latter approximation holds when the former does. In order to formally maximize the performance of policy π_θ , it is useful to turn to policy gradient methods. For the sake of brevity, we propose to focus on deterministic policy gradient methods here [32], based on the policy gradient equation

$$\nabla_\theta J(\pi_\theta) = \mathbb{E} [\nabla_\theta \pi_\theta(s) \nabla_a Q_{\pi_\theta}(s, a)] \quad (35)$$

where J is the expected closed-loop cost associated to running policy π_θ on the real system, and Q_{π_θ} the corresponding action-value function. Note that the expectation \mathbb{E} is taken over trajectories of the real system subject to policy π_θ . A necessary condition of optimality for policy π_θ is then

$$\nabla_\theta J(\pi_\theta) = 0. \quad (36)$$

511 Deterministic policy gradient methods are often built around the
 512 TD actor-critic approach, based on [32]

$$\begin{aligned} \tau_k &= L(s_k, a_k) + \gamma Q_w(s_{k+1}, \pi_\theta(s_{k+1})) - Q_w(s_k, a_k) \\ w &\leftarrow w + \alpha_w \tau_k \nabla_w Q_w(s_k, a_k) \\ \theta &\leftarrow \theta + \alpha_\theta \nabla_\theta \pi_\theta(s_k) \nabla_a Q_w(s_k, \pi_\theta(s_k)) \end{aligned} \quad (37)$$

513 for some $\alpha_w, \alpha_\theta > 0$ small enough, where $Q_w \approx Q_{\pi_\theta}$ is
 514 an approximation of the corresponding action-value function.
 515 Computationally efficient choices of action-value function
 516 parametrization Q_w in the ENMPC context will be the object of
 517 future publications.

518 V. RL AND STABLE ENMPC

519 The main idea in ENMPC is to optimize performance (defined
 520 by a suitably chosen cost) rather than penalizing deviations from
 521 a given reference. Since the cost is generic, the value function
 522 is not guaranteed to be positive definite and proving stability
 523 becomes challenging. The main idea for proving stability is to
 524 introduce a cost modification, called *rotation*, which does not
 525 modify the optimal solution, but yields a positive-definite value
 526 function, and hence is a Lyapunov function, such that nominal
 527 stability follows.

528 Section II focused on learning the optimal policy even in case
 529 a wrong model $\mathbb{P}[\hat{s}_+ | s, u]$ is used. In this section, we aim at
 530 enforcing nominal stability even if the corresponding stage cost
 531 \hat{L} is indefinite.¹ To this end, in the NMPC cost, we replace \hat{L}
 532 by a newly-defined stage cost \bar{L} , which we force to be positive
 533 definite. We then use a cost modification in order to recover the
 534 correct, possibly indefinite, value, and action-value functions
 535 corresponding to \bar{L} . Throughout the section, we assume that the
 536 value and action-value functions are bounded on some set.

537 In order to show how the cost modification can be imple-
 538 mented by the term λ_θ in (21), we first introduce the proposed
 539 cost modification as a generalization of the standard one, prove
 540 that it does not modify the optimal policy, and discuss how it
 541 is used to learn the optimal value and action-value functions.
 542 Then, we discuss how the cost modification relates to the cost
 543 rotation typically used in ENMPC. Finally, we illustrate how
 544 the cost modification can be used in a parametrized ENMPC
 545 scheme to be used as function approximator within RL.

546 A. Generalized Cost Rotation

547 Consider modifying the cost of the problem according to

$$\bar{L}(s, a) = \hat{L}(s, a) + \Lambda(s, a) - \gamma \mathbb{E} [\Lambda(\hat{s}_+, \hat{\pi}(\hat{s}_+)) | s, a] \quad (38a)$$

$$\bar{V}_f(s) = V_*(s) + \Lambda(s, \hat{\pi}(s)) \quad (38b)$$

548 where $\bar{V}_f(s)$ denotes the rotated terminal cost and where

$$\Lambda(s, a) \geq \Lambda(s, \hat{\pi}(s)), \quad \forall s, a \quad (38c)$$

549 holds. Moreover, we require $\Lambda(s, a)$ to be such that $\bar{L}(s, a)$ is
 550 finite whenever $\hat{L}(s, a)$ is finite.

¹We remark that, in this context, stability is obtained for the model used by NMPC for predictions. Future work will investigate obtaining stability guarantees for the real process $\mathbb{P}[s_+ | s, u]$.

It is important to underline that defining a function $\Lambda(s, a)$ strictly satisfying condition (38c) is in general hard, as it requires knowledge of $\hat{\pi}$. A simpler choice satisfying (38c) with equality is $\Lambda(s, a) = \lambda(s)$, where λ is any function satisfying the boundedness assumption on $\bar{L}(s, a)$.

Theorem 2: The modification (38) preserves the optimal control policy $\hat{\pi}$ corresponding to \bar{L} and the model $\mathbb{P}[\hat{s}_+ | s, u]$. Moreover, the optimal value and action-value functions satisfy

$$\bar{V}_N(s) = \hat{V}_N(s) + \Lambda(s, \hat{\pi}(s)) \quad (39a)$$

$$\bar{Q}_N(s, a) = \hat{Q}_N(s, a) + \Lambda(s, a). \quad (39b)$$

Proof: We will first prove the theorem for $N = 1$ and then proceed by induction. The definition of action value function reads

$$\hat{Q}_1(s, a) = \hat{L}(s, a) + \gamma \mathbb{E} [V_*(\hat{s}_+) | s, a]. \quad (40)$$

Then, we can write

$$\begin{aligned} \hat{Q}_1(s, a) &+ \Lambda(s, a) \\ &= \hat{L}(s, a) + \Lambda(s, a) + \gamma \mathbb{E} [V_*(\hat{s}_+) | s, a] \\ &= \bar{L}(s, a) + \gamma \mathbb{E} [\hat{V}_f(\hat{s}_+) | s, a] = \bar{Q}_1(s, a). \end{aligned}$$

Since by definition $\hat{Q}_1(s, a) > \hat{Q}_1(s, \hat{\pi}(s_+)), \forall a \notin \hat{\pi}(s_+)$ and by construction (38c) holds, we obtain that

$$\hat{Q}_1(s, a) > \bar{Q}_1(s, \hat{\pi}(s_+)), \quad \forall a \notin \hat{\pi}(s_+).$$

Therefore, the optimal policy is preserved and

$$\bar{V}_1(s) = \hat{V}_1(s) + \Lambda(s, \hat{\pi}(s)) = \hat{V}_*(s) + \Lambda(s, \hat{\pi}(s)).$$

We now use bootstrapping to obtain

$$\begin{aligned} \hat{Q}_{k+1}(s, a) &+ \Lambda(s, a) \\ &= \hat{L}(s, a) + \Lambda(s, a) + \gamma \mathbb{E} [V_k(\hat{s}_+) | s, a] \\ &= \bar{L}(s, a) + \gamma \mathbb{E} [\hat{V}_k(\hat{s}_+) | s, a] = \bar{Q}_{k+1}(s, a). \end{aligned}$$

Therefore, cost modification (38) preserves the policy over any horizon N , i.e., $\bar{\pi} = \hat{\pi}$, and (39) hold. ■

We will show next that, for any given policy, the modified NMPC scheme can yield any bounded value function, while this does not hold for the action-value function. Consequently any attempt at learning the optimal policy by solely relying on learning the value function and the proposed modification cannot succeed.

Consider a problem formulated using a stage cost $\check{L}(s, a)$, with the corresponding value functions associated to the optimal policy $\check{\pi}$

$$\check{V}_N(s) = \check{L}(s, \check{\pi}(s)) + \gamma \mathbb{E} [\check{V}_{N-1}(\check{s}_+) | s, \check{\pi}(s)] \quad (41a)$$

$$\check{Q}_N(s, a) = \check{L}(s, a) + \gamma \mathbb{E} [\check{V}_{N-1}(\check{s}_+) | s, a]. \quad (41b)$$

Lemma 1: Consider the set $\check{\mathcal{S}}$ such that for all $s \in \check{\mathcal{S}}$ both $\check{V}_N(s)$ and $\hat{V}_N(s)$ are bounded. Then, there exists a cost modification $\Lambda(s, a) = \lambda(s), \forall a$ such that

$$\check{V}_N(s) + \lambda(s) = \hat{V}_N(s). \quad (42)$$

581 *Proof:* The result is obtained by choosing $\lambda(s) = \hat{V}_N(s) -$
 582 $\check{V}_N(s)$. ■

583 B. Strict Dissipativity

584 Since most of the literature on dissipativity-based ENMPC
 585 focuses on deterministic formulations, in this subsection we
 586 also restrict to the deterministic case and we adopt $\gamma = 1$, as is
 587 usual in the literature on ENMPC. Strict dissipativity is typically
 588 used in ENMPC in order to obtain a positive-definite cost to
 589 construct a Lyapunov function and prove closed-loop stability.
 590 Strict dissipativity holds if there exists a function λ such that

$$\lambda(s_+) - \lambda(s) \leq -\rho(\|s - s_e\|) + L(s, a) - L(s_e, a_e) \quad (43)$$

591 with ρ a positive-definite function and

$$(s_e, a_e) = \underset{s, a}{\operatorname{argmin}} \quad L(s, a) \quad \text{s.t. } s = f(s, a).$$

592 In [4], it has been proven that strict dissipativity is sufficient
 593 for closed-loop stability, provided that λ is bounded and the
 594 constraint set is compact. In [24], it has been proven that, under
 595 a controllability assumption and if (s_e, a_e) lies in the interior of
 596 the constraint set, strict dissipativity with a bounded function λ is
 597 also necessary for closed-loop stability on a compact constraint
 598 set.

599 For simplicity and without loss of generality, we assume that
 600 $L(s_e, a_e) = 0$.

601 *Lemma 2:* If there exists a function $\Lambda(s, a)$ such that $\min_a \Lambda$
 602 $(s, a) = \Lambda(s, \hat{\pi}(s)) = \lambda(s)$, with $\lambda(s)$ satisfying the strict dissi-
 603 pativity inequality (43), then $\bar{L}(s, a) \geq \rho(\|s - s_e\|) \geq 0$ holds.

604 *Proof:* The proof is obtained by noting that

$$\begin{aligned} \bar{L}(s, a) &= L(s, a) + \Lambda(s, a) - \Lambda(\hat{s}_+, \hat{\pi}(s_+)) \\ &\geq L(s, a) + \lambda(s) - \lambda(\hat{s}_+) \geq \rho(\|s - s_e\|) \end{aligned}$$

605 where the first inequality follows from (38c), and the second
 606 inequality is a direct consequence of (43) and $L(s_e, a_e) = 0$. ■

607 Lemma 2 is of paramount importance in the context of EN-
 608 MPC because it establishes that any stabilizing optimal policy
 609 π_\star originating from any given stage cost $L(s, a)$ can be learned
 610 by using a parametrization that yields a positive-definite stage
 611 cost $\bar{L}(s, a)$. Note that the proposed cost modification is a gen-
 612 eralization of the cost rotation used for ENMPC, as we replace
 613 $\lambda(s)$ with $\Lambda(s, a)$.

614 We ought to stress here that though the results presented in
 615 this section pertain to the deterministic case, we expect them
 616 to extend to the stochastic case as well. Unfortunately a mature
 617 dissipativity theory for stochastic ENMPC has not been devel-
 618 oped yet. It would arguably not be surprising if a stochastic
 619 dissipativity criterion can be put in the form

$$\bar{L}(s, a) \geq \rho(\|s - s_e\|)$$

620 with \bar{L} defined in (38) in combination with appropriate termi-
 621 nal conditions, hence offering a generalization of [35]. These
 622 questions will be the object of future research.

623 C. Economic Reward and NMPC Parametrization

624 In the following, we will first show that the proposed cost
 625 modification can be reduced to a cost on the initial state, which

is introduced in the ENMPC scheme (21) as the term λ_θ . Second,
 626 we will explain how we can use a stability-enforcing positive-
 627 definite stage cost to approximate the value and action-value
 628 function of the economic cost.
 629

For simplicity, we use the cost modification $\Lambda(s, a) = \lambda(s)$.
 630 Note that we drop the dependence on the control in Λ for prac-
 631 tical reasons: 1) keeping it would require knowledge of the
 632 optimal policy and 2) $\lambda(s)$ is a valid choice in the sense that it
 633 satisfies (38c).
 634

We observe that the cost modification can be summarized by
 635 the initial cost $\lambda(x_0)$ by relying on the fact that for all predicted
 636 trajectories the modified cost reads as [4], [12]
 637

$$\begin{aligned} &= \gamma^N V^f(x_N) + \sum_{k=0}^{N-1} \gamma^k l(x_k, u_k) + \gamma^k \lambda(x_k) - \gamma^{k+1} \lambda(x_{k+1}) \\ &= \lambda(x_0) + \gamma^N V^f(x_N) + \sum_{k=0}^{N-1} \gamma^k l(x_k, u_k). \end{aligned}$$

In (21), we parametrize the cost modification as $\lambda_\theta(x_0)$

We now turn to the question of enforcing stability in the
 639 parametrized ENMPC scheme. In the literature on EMPC the
 640 cost modification is used to prove stability in case the stage
 641 cost is not positive definite. In our case, we are interested in
 642 the opposite: We would like to use a positive-definite stage cost
 643 to enforce stability while the cost modification is used in order
 644 to obtain value and action-value functions corresponding to the
 645 economic (nonpositive definite) cost.
 646

Thanks to Theorem 1 and Lemma 2, introducing the term λ_θ
 647 in (21) guarantees that if the optimal policy π_\star is stabilizing for
 648 the model dynamics (21b), then the ENMPC scheme (21) can
 649 deliver the optimal policy π_\star and value functions V_\star, Q_\star with
 650 a stage cost l_θ lower bounded by \mathcal{K}_∞ .
 651

VI. ANALYTICAL CASE STUDY: THE LQR CASE

One of the few cases where (26) can be constructed and sat-
 653 isfied exactly is the LQR case. We use it as a simple illustration
 654 of the meaning of \hat{L} , Assumption (16) and the cost modification
 655 $\Lambda(s, a)$. Consider a centered linear quadratic Gaussian control
 656 problem with the true dynamics and stage cost
 657

$$s_+ = As + Ba + e, \quad e \sim \mathcal{N}(0, \Sigma) \quad (44)$$

$$L(s, a) = \begin{bmatrix} s \\ a \end{bmatrix}^\top \begin{bmatrix} T & N \\ * & R \end{bmatrix} \begin{bmatrix} s \\ a \end{bmatrix}. \quad (45)$$

The associated value functions, if they exist, read as

$$V_\star = s^\top S s + V_0$$

$$Q_\star = \begin{bmatrix} s \\ a \end{bmatrix}^\top \begin{bmatrix} T + \gamma A^\top S A & N + \gamma A^\top S B \\ * & R + \gamma B^\top S B \end{bmatrix} \begin{bmatrix} s \\ a \end{bmatrix} + V_0 \quad (46)$$

where $V_0 = \gamma(1 - \gamma)^{-1} \operatorname{Tr}(S\Sigma)$. Matrix S and the associated
 659 optimal policy K_\star are delivered by the Schur complement of
 660 the quadratic form in Q_\star , i.e., the discounted LQR equations
 661

$$T + \gamma A^\top S A = S + (N + \gamma A^\top S B) K_\star \quad (47a)$$

$$(R + \gamma B^\top S B) K_\star = N^\top + \gamma B^\top S A. \quad (47b)$$

662 *A. LQR With Imperfect Model and \hat{L}*

663 We now consider the deterministic model

$$\hat{s}_+ = \hat{A}s + \hat{B}a. \quad (48)$$

664 In order to verify Theorem 1, we consider the stage cost

$$\begin{aligned} \hat{L}(s, a) &= Q_*(s, a) - \gamma V_*(\hat{s}_+) \\ &= \begin{bmatrix} s \\ a \end{bmatrix}^\top \begin{bmatrix} \hat{T} & \hat{N} \\ * & \hat{R} \end{bmatrix} \begin{bmatrix} s \\ a \end{bmatrix} + (1 - \gamma) V_0. \end{aligned}$$

665 This implies that \hat{T} , \hat{N} , \hat{R} must satisfy

$$\hat{T} + \gamma \hat{A}^\top S \hat{A} = T + \gamma A^\top S A \quad (49a)$$

$$\hat{N} + \gamma \hat{A}^\top S \hat{B} = N + \gamma A^\top S B \quad (49b)$$

$$\hat{R} + \gamma \hat{B}^\top S \hat{B} = R + \gamma B^\top S B. \quad (49c)$$

666 The resulting value function reads as $\hat{V}(s) = s^\top \hat{S}s + \hat{V}_0$,
667 where \hat{S} and the associated policy \hat{K} satisfy

$$\hat{T} + \gamma \hat{A}^\top \hat{S} \hat{A} = \hat{S} + (\hat{N} + \gamma \hat{A}^\top \hat{S} \hat{B}) \hat{K} \quad (50a)$$

$$(\hat{R} + \gamma \hat{B}^\top \hat{S} \hat{B}) \hat{K} = \hat{N}^\top + \gamma \hat{B}^\top \hat{S} \hat{A}. \quad (50b)$$

668 Using (49), we obtain that matrices $\hat{S} = S$ and $\hat{K} = K_*$ sat-
669 isfy (50).670 *B. Assumption (16) in the LQR Case*671 In order for the discrete algebraic Riccati equation
672 (DARE) (50) to deliver a valid LQR solution \hat{S} , \hat{K} (in the
673 sense of a Bellman optimality backup), Assumption (16) needs
674 to be satisfied. In the case $\gamma = 1$, this requires that $\hat{A} - \hat{B}\hat{K}_*$
675 has all its eigenvalues inside the unit circle. We illustrate this
676 fact by the following simple example:

$$A = 1, \quad B = 1, \quad T = 1, \quad R = 2, \quad N = 0$$

677 which yields $S = 2$ and $K_* = 0.5$. By using the model $\hat{A} = 2$,
678 $\hat{B} = 1$, we obtain

$$\hat{T} = -5, \quad \hat{R} = 2, \quad \hat{N} = -2$$

679 such that $\hat{A} - \hat{B}\hat{K}_* = 1.5$ and K_* corresponds to the nonsta-
680 bilizing solution $\hat{S} = 2$ of the DARE. Note, however, that the
681 DARE does have a stabilizing solution, which reads

$$\hat{S} = 7, \quad \hat{K} = 4/3 \quad \text{such that} \quad \hat{A} - \hat{B}\hat{K} = 2/3.$$

682 *C. Economic LQR and Cost Modification $\Lambda(s, a)$* 683 We now consider an economic LQR, for which $\hat{L}(s, a)$ is
684 indefinite. We introduce matrices δT , δN , δR to define

$$\Lambda(s, a) := \begin{bmatrix} s \\ a \end{bmatrix}^\top \begin{bmatrix} \delta T & \delta N \\ \delta N^\top & \delta R \end{bmatrix} \begin{bmatrix} s \\ a \end{bmatrix}$$

685 and observe that $\delta N = K_*^\top \delta R$, $\delta R \succeq 0$ must hold in order for
686 (38c) to be fulfilled, where K_* is the LQR controller gain asso-
687 ciated to the stage cost L for the true system (47b). By defining $\delta S := \delta T - K_*^\top \delta R K_*$, we obtain

$$\mathbb{E} [\Lambda(s_+, \hat{\pi}(\hat{s}_+)) | s, a] = \begin{bmatrix} s \\ a \end{bmatrix}^\top \begin{bmatrix} \hat{A}^\top \delta S \hat{A} & \hat{A}^\top \delta S \hat{B} \\ \hat{B}^\top \delta S \hat{A} & \hat{B}^\top \delta S \hat{B} \end{bmatrix} \begin{bmatrix} s \\ a \end{bmatrix}$$

such that admissible stage cost modifiers are based on the
689 quadratic forms
690

$$\begin{aligned} \delta W_L &= \begin{bmatrix} K_*^\top \delta R K_* & K_*^\top \delta R \\ \delta R K_* & \delta R \end{bmatrix} - \begin{bmatrix} \gamma \hat{A}^\top \delta S \hat{A} - \delta S & \gamma \hat{A}^\top \delta S \hat{B} \\ \gamma \hat{B}^\top \delta S \hat{A} & \gamma \hat{B}^\top \delta S \hat{B} \end{bmatrix} \\ &= \delta W_L^0 + \delta W_L^1. \end{aligned}$$

It follows that the modified action-value function reads

$$\hat{Q}(s, a) = \begin{bmatrix} s \\ a \end{bmatrix}^\top (W + \delta W_L) \begin{bmatrix} s \\ a \end{bmatrix}. \quad (51)$$

Since we are interested in stability-enforcing schemes, one
692 needs to choose δR and δS such that $W + \delta W_L \succ 0$ holds.
693 Since this expression is linear in δR and δS , the problem
694 amounts to solving an LMI.
695In the following, we state explicitly how the two contributions
696 in δW_L relate to conditions (38c) and (43) respectively. The term
697 δW_L^1 can be framed as a quadratic stage cost rotation and the
698 condition
699

$$\delta W_L^1 = - \begin{bmatrix} \gamma \hat{A}^\top \delta S \hat{A} - \delta S & \gamma \hat{A}^\top \delta S \hat{B} \\ \gamma \hat{B}^\top \delta S \hat{A} & \gamma \hat{B}^\top \delta S \hat{B} \end{bmatrix} \succ 0$$

is the strict dissipativity condition (43) for the linear-quadratic
700 case [42]. For $\delta R = 0$, we obtain $\Lambda(s, a) = \lambda(s) = s^\top \delta S s$.
701The term δW_L^0 resembles the stage cost used in [11], [41] and
702 satisfies
703

$$\Lambda^0(s, a) = \begin{bmatrix} s \\ a \end{bmatrix}^\top \delta W_L^0 \begin{bmatrix} s \\ a \end{bmatrix} = (a + K_* s)^\top \delta R (a + K_* s)$$

such that $\Lambda^0(s, a) > \Lambda^0(s, -K_* s)$, $\forall a \neq -K_* s$, $\forall s$. As dis-
704 cussed in [11], [41], the use of $\Lambda^0(s, a)$ as stage cost with a
705 zero terminal cost yields a scheme, which delivers the optimal
706 feedback for the nominal model. The related value function and
707 action-value function, however, are zero.
708

VII. NUMERICAL EXAMPLES

In this section, we propose two examples in order to illustrate
710 the theoretical developments.
711

A. Linear MPC

We first consider a simple linear MPC example to illustrate
713 the methods mentioned above. We consider the MPC scheme
714

$$\min_{x, u} V_0 + \frac{\gamma^N}{2} x_N^\top S_N x_N + \sum_{k=0}^{N-1} f^\top \begin{bmatrix} x_k \\ u_k \end{bmatrix} \quad (52a)$$

$$\sum_{k=0}^{N-1} \frac{1}{2} \gamma^k \left(\|x_k\|^2 + \frac{1}{2} \|u_k\|^2 + w^\top s_k \right) \quad (52b)$$

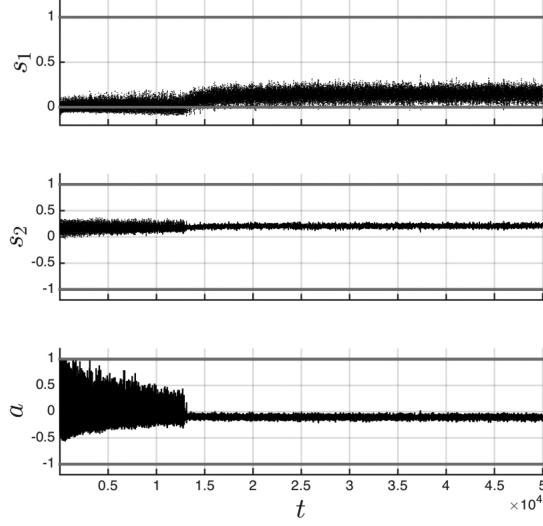


Fig. 1. State and input trajectories for the example detailed in Section VII-A.

$$715 \quad \text{s.t.} \quad x_{k+1} = Ax_k + Bu_k + b \quad (52c)$$

$$\begin{bmatrix} 0 \\ -1 \end{bmatrix} + \underline{x} - s_k \leq x_k \leq \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \bar{x} + s_k \quad (52d)$$

$$-1 \leq u_k \leq 1, \quad x_0 = s \quad (52e)$$

716 where the NMPC parameters subject to the RL scheme are

$$\theta = (V_0, \underline{x}, \bar{x}, b, f, A, B) \quad (53)$$

717 and where the ℓ_1 relaxation uses the weight $w^\top = [10^2 \ 10^2]$.
 718 We selected a discount factor $\gamma = 0.9$. The terminal cost matrix
 719 S_N is selected as the Riccati matrix underlying the LQR con-
 720 troller locally equivalent to the MPC scheme. The objective of
 721 the MPC scheme is to drive the states to the origin, such that the
 722 MPC reference is activating the lower bound of the first state. A
 723 horizon of $N = 10$ is used. The MPC model is initially chosen
 724 as

$$A = \begin{bmatrix} 1 & 0.25 \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0.0312 \\ 0.25 \end{bmatrix} \quad (54)$$

725 while all other parameters are initialized with zero values.

726 We will consider that the “real” process is following the dy-
 727 namics

$$x_{k+1} = \begin{bmatrix} 0.9 & 0.35 \\ 0 & 1.1 \end{bmatrix} x_k + \begin{bmatrix} 0.0813 \\ 0.2 \end{bmatrix} u_k + \begin{bmatrix} e_k \\ 0 \end{bmatrix} \quad (55)$$

728 where e_k is a random, uncorrelated, uniformly distributed vari-
 729 able in the interval $[-10^{-1}, 0]$, and therefore drives the first
 730 state to violate its lower bound.

731 We use the on-policy algorithm (32) without introducing ex-
 732 ploration. The step size was selected as $\alpha = 10^{-6}$. Fig. 1 shows
 733 the resulting state and input trajectories. Fig. 2 shows the adap-
 734 tation of the MPC parameters via the on-policy algorithm (32).
 735 Fig. 3 shows the evolution of the stage cost $L(s, a)$ (including
 736 the large ℓ_1 penalties for the constraints violations), and the
 737 evolution of the TD error (32a). It can be observed that the RL

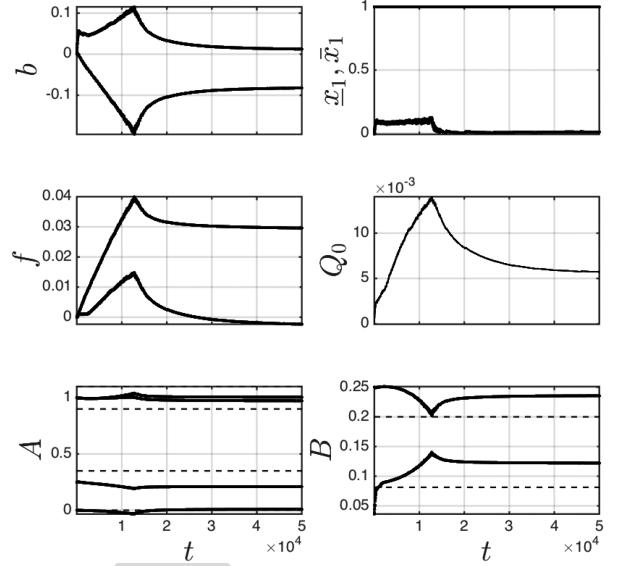


Fig. 2. Trajectory of the parameters θ for the example detailed in Section VII-A. The dashed lines report the “real system” A and B entries. One can observe that the RL scheme does not perform system identification, as the MPC model does not converge to the “real system.”

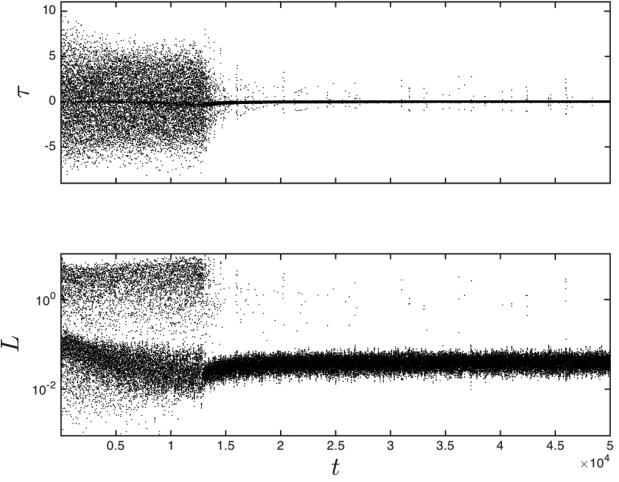


Fig. 3. Evolution of the stage cost (52b) and TD error (32a) achieved for the example detailed in Section VII-A.

algorithm manages to reduce the TD error τ to small values, 738 averaging to zero. The state trajectories often violate the state 739 bound $x_1 \geq 0$ in the beginning, resulting in large control actions 740 (see Fig. 1), but the RL adjusts the MPC parameters in order to 741 avoid these expensive violations. The adaptation of the param- 742 eters is a combination of modifying the stage gradient f and of 743 introducing a model bias b (mostly on the first state, subject to 744 the process noise). 745

We have also deployed this example without letting the RL 746 scheme adapt the model parameters. The MPC performance is 747 then increased mostly via tightening the bounds, and does not 748 reach the performance shown in Fig. 2. For the sake of brevity, 749 we do not report these results here. 750

TABLE I
MODEL PARAMETERS

<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	<i>g</i>
0.5616	0.3126	48.43	0.507	55	0.1538	55
<i>h</i>	<i>M</i>	<i>C</i>	<i>U_{A₂}</i>	<i>C_p</i>	λ	λ_s
0.16	20	4	6.84	0.07	38.5	36.6
<i>F₁</i>	<i>X₁</i>	<i>F₃</i>	<i>T₁</i>	<i>T₂₀₀</i>		
10	5 %	50	40	25		

The units are omitted and are consistent with the physical quantities they correspond to.

B. Evaporation Process

We consider an example from the process industry, i.e., the evaporation process modeled in [34], [39] and used in [5], [42] in the context of economic MPC. The model equations include states (X_2, P_2) (concentration and pressure) and controls (P_{100}, F_{200}) (pressure and flow). The model further depends on concentration X_1 , flow F_2 , and temperatures T_1, T_{200} , which are assumed to be constant in the control model. In reality, these quantities are stochastic with variance $\sigma_{X_1} = 1$, $\sigma_{F_1} = 2$, $\sigma_{T_1} = 8$, $\sigma_{T_{200}} = 5$, and mean centered on the nominal value. Bounds $(25, 40) \leq (X_2, P_2) \leq (100, 80)$ on the states and $u_l = (100, 100) \leq (P_{100}, F_{200}) \leq (400, 400) = u_u$ on the controls are present. In particular, the bound $X_2 \geq 25$ is introduced in order to ensure sufficient quality in the product. All state bounds are relaxed and translated into cost terms as in Section IV-A. The system dynamics are given by [5]

$$M\dot{X}_2 = F_1 X_1 - F_2 X_2, \quad C\dot{P}_2 = F_4 - F_5 \quad (56)$$

where

$$\begin{aligned} T_2 &= aP_2 + bX_2 + c, & T_3 &= dP_2 + e \\ \lambda F_4 &= Q_{100} - F_1 C_p (T_2 - T_1), & T_{100} &= fP_{100} + g \\ Q_{100} &= U_{A_1} (T_{100} - T_2), & U_{A_1} &= h(F_1 + F_3) \\ Q_{200} &= \frac{U_{A_2} (T_3 - T_{200})}{1 + U_{A_2} / (2C_p F_{200})}, & F_{100} &= \frac{Q_{100}}{\lambda_s} \\ \lambda F_5 &= Q_{200}, & F_2 &= F_1 - F_4 \end{aligned}$$

and the model parameters are given in Table I. The economic objective is given by

$$L(x, u) = 10.09(F_2 + F_3) + 600F_{100} + 0.6F_{200}.$$

We introduce functions $\lambda_\theta, V_\theta^f, l_\theta$ as fully parametrized quadratic functions defined by Hessian H_\dagger , gradient h_\dagger and constant $c_\dagger, \dagger = \{\lambda, V^f, l\}$ to formulate the ENMPC controller

$$\begin{aligned} \min_z \quad & \lambda_\theta(x_0) + \gamma^N (V_\theta^f(x_N) + w^\top \sigma_f) \\ & + \sum_{k=0}^{N-1} \gamma^k (l_\theta(x_k, u_k) + w^\top \sigma_k) \\ \text{s.t. } & x_{k+1} = f_\theta(x_k, u_k), \quad x_0 = s \\ & g(u_k) \leq 0, \quad h_\theta(x_k) \leq \sigma_k. \end{aligned}$$

The model is parametrized as the nominal model with the addition of a constant, i.e., $f_\theta(x, u) = f(x, u) + c_f$. The control

constraints are fixed and the state constraints are parametrized as simple bounds, i.e., $h(x) = (x - x_l, x_u - x)$. The vector of parameter, therefore, reads as

$$\theta = (H_\lambda, h_\lambda, c_\lambda, H_{V^f}, h_{V^f}, c_{V^f}, H_l, h_l, c_l, c_f, x_l, x_u).$$

Constants $w = 1$ are fixed and assumed to reflect the known cost of violating the state constraints.

We use the batch policy update (32) with $\alpha = 10^{-4}$ and we update the parameters with the learned ones every $N_{\text{upd}} = 2000$ time steps. In order to induce enough exploration, we use an ϵ -greedy policy which is greedy 90% of the samples, while in the remaining 10%, we perturb the optimal feedback as

$$a = \text{sat}(u_0^* + e, u_l, u_u), \quad e \sim \mathcal{N}(0, 1)$$

where $\text{sat}(\cdot, u_l, u_u)$ saturates the input between its lower and upper bounds $u_l = (100, 100), u_u = (400, 400)$, respectively. We initialize the ENMPC scheme by tuning the cost using the economic-based approach proposed in [42], which is based on the nominal model. For the model, we use $c_f = 0$ and the bounds are initialized at their nominal values. While at every step we do check that $H_l, H_{V^f} \succ 0$, during the learning phase the parameters never violate this constraint. As shown in Fig. 4, the algorithm converges to a constant parameter value while reducing the average TD error.

We ran a closed-loop simulation to compare the performance of RL-tuned NMPC with the one using the economic-based tuning proposed in [42]. We show in Fig. 5 that the difference in cost and concentration X_2 between the two ENMPC schemes. It can be seen that RL keeps X_2 at higher values in order to reduce the violation of the quality constraint. This entails an improvement in the cost, which is about 7% in the considered scenario.

It is important to stress here that both the cost and the model parametrization do not have the same structure as the real cost and model. Therefore, the action-value function and, consequently, the policy can be learned only approximately. Another important remark concerns the role of the storage function λ_θ . RL exploits this function to keep the stage cost positive definite. If we invert the cost modification (38) in order to recover an approximation of the economic stage cost, we obtain an indefinite one.

We ran an additional simulation in which the model used in simulations is deterministic and coincides with the one used for predictions in NMPC. We initialize the learning phase with the nominally tuned parameters from [42]. In this case, RL keeps the parameters essentially unaltered, which suggests that the nominal tuning was already optimal.

Consider the naive initial guess $H_l = I, x_l = (25, 40), x_u = (100, 80)$, while all other parameters are 0. In order to obtain parameters, which are close to convergence in each batch, we set $N_{\text{upd}} = 20000$. In this case, with 10^5 samples we did not yet obtain convergence. Even though convergence of the RL scheme is much slower, the parameters are approaching a steady value and the closed-loop performance is improved with respect to the initial guess.

While our simulation results are promising, we ought to specify here that we have observed some potential

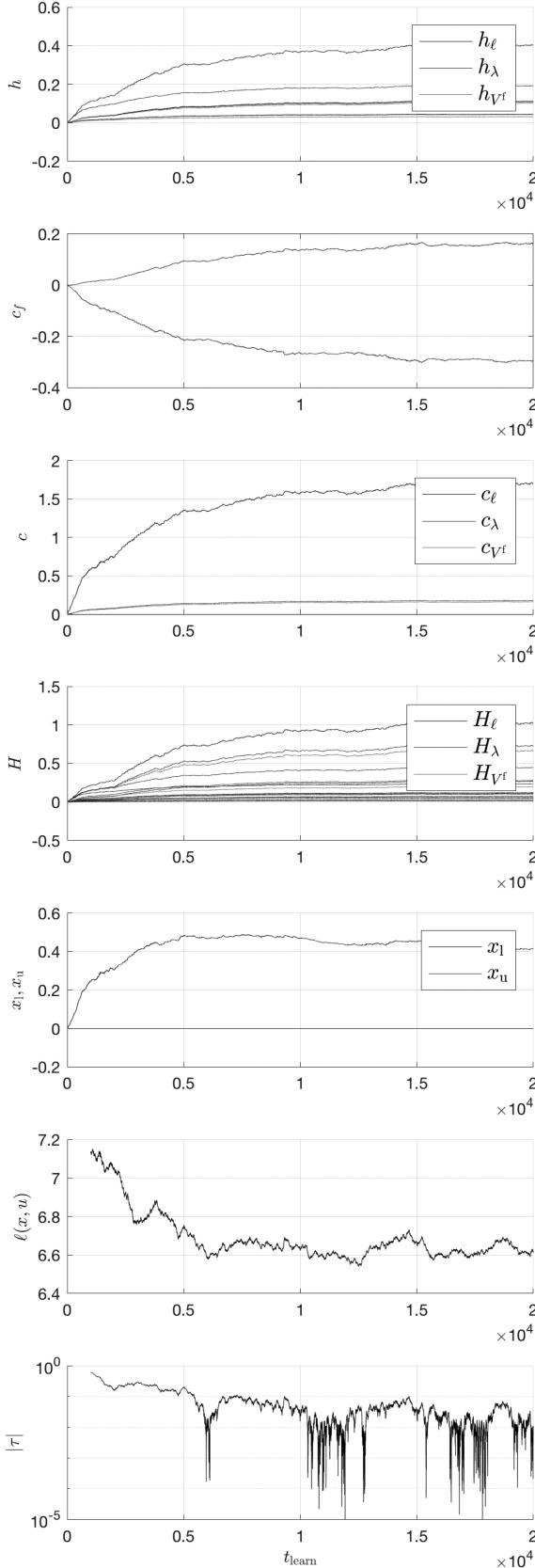


Fig. 4. Evolution of the parameters (increment w.r.t. the initial guess value) and of the TD error (averaged over the preceding 1000 samples) during learning.

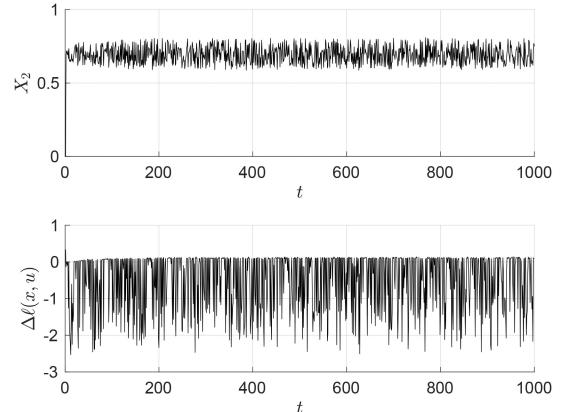


Fig. 5. Closed-loop simulations: Difference between the RL-tuned NMPC, and the nominal economic tuning [42]. Top plot: Difference in concentration X_2 . Bottom plot: Difference in economic cost.

difficulties related—to the best of our knowledge—to using a gradient method in adjusting the ENMPC parameters and using a Q -learning method as a proxy for learning the optimal policy. These observations arguably call for exploring the effectiveness of deploying second-order methods, such as LSTD-type methods [36], and policy gradient approaches [32] to alleviate the difficulties observed in some cases.

VIII. CONCLUSION

In this paper, we propose to use ENMPC schemes to support the parametrization of the value functions and/or the policy, which is an essential component of RL. We show that the ENMPC schemes can generate the optimal policy for the real system even if the underlying model is wrong, by adjusting the stage and terminal cost alone. We also show how a positive stage and terminal cost can be used in the ENMPC scheme to learn the optimal control policy, resulting in an ENMPC scheme that is stable by construction. The resulting ENMPC delivers the optimal control policy for the real system if that policy is itself stabilizing. We additionally detail how some classic RL methods can be deployed in practice to adjust the parameters of the ENMPC scheme. The methods are illustrated in simulations.

Future work will propose improvements in the RL algorithms specific for ENMPC, and propose an efficient combination of the existing classic model-tuning techniques and the RL-based tuning.

REFERENCES

- [1] P. Abbeel, A. Coates, M. Quigley, and A. Y. Ng, "An application of reinforcement learning to aerobatic helicopter flight," in *Advances in Neural Information Processing Systems 19*. Cambridge, MA, USA: MIT Press, 2007, p. 2007.
- [2] M. Agarwal, "Feasibility of on-line reoptimization in batch processes," *Chem. Eng. Commun.*, vol. 158, no. 1, pp. 19–29, 1997.
- [3] M. Agarwal, "Iterative set-point optimization of batch chromatography," *Comp. Chem. Eng.*, vol. 29, no. 6, pp. 1401–1409, 2005.
- [4] R. Amrit, J. Rawlings, and D. Angeli, "Economic optimization using model predictive control with a terminal cost," *Annu. Rev. Control*, vol. 35, pp. 178–186, 2011.

- [5] R. Amrit, J. B. Rawlings, and L. T. Biegler, "Optimizing process economics online using model predictive control," *Comput. Chem. Eng.*, vol. 58, pp. 334–343, 2013.
- [6] D. Angeli, R. Amrit, and J. Rawlings, "On average performance and stability of economic model predictive control," *IEEE Trans. Autom. Control*, vol. 57, no. 7, pp. 1615–1626, Jul. 2012.
- [7] A. Aswani, H. Gonzalez, S. S. Sastry, and C. Tomlin, "Provably safe and robust learning-based model predictive control," *Automatica*, vol. 49, no. 5, pp. 1216–1226, 2013.
- [8] F. Berkenkamp, M. Turchetta, A. Schoellig, and A. Krause, "Safe model-based reinforcement learning with stability guarantees," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., New York, NY, USA: Curran Associates, Inc., 2017, pp. 908–918.
- [9] D. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1, 3rd ed. Belmont, MA, USA: Athena Scientific, 2005.
- [10] C. Büskens and H. Maurer, "Sensitivity analysis and real-time optimization of parametric nonlinear programming problems," *Online Optimization of Large Scale Systems*. Berlin Heidelberg, Germany: Springer, 2001, pp. 3–16.
- [11] L. Chisci, J. A. Rossiter, and G. Zappa, "Systems with persistent disturbances: Predictive control with restricted constraints," *Automatica*, vol. 37, pp. 1019–1028, 2001.
- [12] M. Diehl, R. Amrit, and J. B. Rawlings, "A Lyapunov function for economic optimizing model predictive control," *IEEE Trans. Autom. Control*, vol. 56 no. 3, pp. 703–707, Mar. 2011.
- [13] J. F. Forbes and T. E. Marlin, "Design cost: A systematic approach to technology selection for model-based real-time optimization systems," vol. 20, no. 67, pp. 717–734, 1996.
- [14] J. F. Forbes, T. E. Marlin, and J. F. MacGregor, "Model adequacy requirements for optimizing plant operations," *Comp. Chem. Eng.*, vol. 18, no. 6, pp. 497–510, 1994.
- [15] W. Gao and S. Engell, "Comparison of iterative set-point optimisation strategies under structural plant-model mismatch," vol. 38, pp. 12–17, 2005.
- [16] L. Grüne and J. Pannek, *Nonlinear Model Predictive Control*. London, U.K.: Springer, 2011.
- [17] L. Hewing, A. Liniger, and M. N. Zeilinger, "Cautious NMPC with Gaussian process dynamics for miniature race cars," 2017, arXiv:1711.06586.
- [18] J. Fernandez and J. Garcia, "A comprehensive survey on safe reinforcement learning," *J. Mach. Learn. Res.*, vol. 16, pp. 1437–1480, 2013.
- [19] J. Peters, J. Kober, and J. A. Bagnell, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, pp. 1238–1274, 2013.
- [20] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, "Learning-based model predictive control for safe exploration and reinforcement learning," 2018, arXiv:1803.08287.
- [21] J. Löfberg, "Oops! I cannot do it again: Testing for recursive feasibility in MPC," *Automatica*, vol. 48, no. 3, pp. 550–555, 2012.
- [22] A. Marchetti, B. Chachuat, and D. Bonvin, "Modifier-adaptation methodology for real-time optimization," *Ind. Eng. Chem. Res.*, vol. 48, no. 13, pp. 6022–6033, 2009.
- [23] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 26, no. 6, pp. 789–814, 2000.
- [24] M. A. Müller, D. Angeli, and F. Allgöwer, "On necessity and robustness of dissipativity in economic model predictive control," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1671–1676, Jun. 2015.
- [25] C. J. Ostafew, A. P. Schoellig, and T. D. Barfoot, "Robust constrained learning-based NMPC enabling reliable mobile robot path tracking," *Int. J. Robot. Res.*, vol. 35, no. 13, pp. 1547–1563, 2016.
- [26] J. B. Rawlings and R. Amrit, "Optimizing process economic performance using model predictive control," in *Proc. IFAC Conf. Nonlinear Model Predictive Control Pavia*, 2009, pp. 119–138.
- [27] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Madison, WI, USA: Nob Hill, 2009.
- [28] P. D. Roberts, "An algorithm for steady-state system optimization and parameter estimation," *Int. J. Syst. Sci.*, vol. 10, no. 7, pp. 719–734, 1979.
- [29] P. D. Roberts, "Coping with model-reality differences in industrial process optimisation, a review of integrated system optimisation and parameter estimation (isope)," *Comput. Industry*, vol. 26, no. 3, pp. 281–290, 1995.
- [30] P. O. M. Scokaert and J. B. Rawlings, "Feasibility issues in linear model predictive control," *AICHE J.*, vol. 45, no. 8, pp. 1649–1659, 1999.
- [31] D. Silver *et al.*, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, pp. 484–503, 2016.
- [32] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proc. 31st Int. Conf. Int. Conf. Mach. Learn.*, vol. 32, 2014, pp. I–387–I–395.
- [33] D. Simon and J. Löfberg, "Stability analysis of model predictive controllers using mixed integer linear programming," in *IEEE 55th Conference Decis. Control*, 2016, pp. 7270–7275.
- [34] C. Sonntag, O. Stursberg, and S. Engell, "Dynamic optimization of an industrial evaporator using graph search with embedded nonlinear programming," in *Proc. 2nd IFAC Conf. Anal. Des. Hybrid Syst.*, 2006, pp. 211–216.
- [35] P. Sopasakis, D. Herceg, P. Patrinos, and A. Bemporad, "Stochastic economic model predictive control for Markovian switching systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 524–530, 2017.
- [36] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.
- [37] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Proc. 12th Int. Conf. Neural Inform. Process. Syst.*, 1999, pp. 1057–1063.
- [38] P. Tatjewski, "Iterative optimizing set-point control—the basic principle redesigned," vol. 35, pp. 49–54, 2002.
- [39] F. Y. Wang and I. T. Cameron, "Control studies on a model evaporation process—Constrained state driving with conventional and higher relative degree systems," *J. Process Control*, vol. 4, pp. 59–75, 1994.
- [40] S. Wang, W. Chaovallitwongse, and R. Babuska, "Machine learning algorithms in bipedal robot control," *Trans. Syst. Man Cybern. Part C*, vol. 42, no. 5, pp. 728–743, Sep. 2012.
- [41] M. Zanon, T. Charalambous, H. Wymeersch, and P. Falcone, "Optimal scheduling of downlink communication for a multi-agent system with a central observation post," *IEEE Control Syst. Lett.*, vol. 2, no. 1, pp. 37–42, Jan. 2018.
- [42] M. Zanon, S. Gros, and M. Diehl, "A tracking MPC formulation that is locally equivalent to economic MPC," *J. Process Control*, vol. 45, pp. 30–42, 2016.

Sébastien Gros received the Ph.D. degree from École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, in 2007.



He was part of an R&D group hosted at Strathclyde University, Glasgow, U.K., focusing on wind turbine control. In 2011, he joined the university of KU Leuven, Leuven, Belgium, where his main research focus was on optimal control and fast NMPC for complex mechanical systems. He joined the Department of Signals and Systems at Chalmers University of Technology, Göteborg, Sweden, in 2013, where he became Associate Professor in 2017. He is now Full Professor with Norwegian University of Science and Technology, Norway, and Affiliate Professor with Chalmers. His research interests include numerical methods, real-time optimal control, reinforcement learning, and the optimal control of energy-related applications.

Mario Zanon received the master's degree in mechatronics from the University of Trento, Trento, Italy, and the Diplôme d'Ingénieur degree from the Ecole Centrale Paris, Paris, France, in 2010. After research stays at the KU Leuven, Leuven, Belgium, University of Bayreuth, Bayreuth, Germany, Chalmers University, Göteborg, Sweden, and the University of Freiburg, Freiburg, Germany, he received the Ph.D. degree in electrical engineering from the KU Leuven in November 2015.



He held a Postdoc Researcher Position with Chalmers University until the end of 2017 and is now an Assistant Professor with the IMT School for Advanced Studies Lucca, Lucca, Italy. His research interests include numerical methods for optimization, economic MPC, reinforcement learning, and the optimal control and estimation of nonlinear dynamic systems, in particular for aerospace and automotive applications.