PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

Iterative Online Optimal Feedback Control

Yuqing Chen and David J. Braun* *Member, IEEE*

***Abstract*—This paper proposes a data-driven iterative feedback control method to efficiently solve finite time horizon, nonlinear, input constrained optimal control problems. The proposed method introduces a *novel approach to combine an inexact system model with measured state information* to *reduce the cost* and *provide near-optimal control* by approximately solving the optimal control problem along the trajectory of the real system, as opposed to solving it along the trajectory predicted by the inexact model. We present a new algorithm that implements the proposed method, establish the convergence and optimality properties of the proposed algorithm, and compare it to optimal feedback control and model-predictive control that solve the same optimal control problem along the trajectory predicted by the inexact model. Finally, we illustrate the generality of the proposed algorithm by approximately solving a challenging optimal control problem with unknown and changing dynamics.**

***Index Terms*—Optimal control; Optimization algorithms; Nonlinear systems; Uncertain systems.**

I. Introduction

Optimal control theory provides a mathematical formalism to control dynamical systems; find the control inputs that minimize a user defined cost, assuming that the *exact model* of the system is known [1]–[3]. However, *models are inexact*, regardless of whether obtained from first principles, offline system identification, or online learning, and it is the inexactness of the model that leads to sub-optimality, *increased cost* and *reduced robustness to model uncertainty*, compared to the optimal controller that is based on the exact model.

Model predictive control (MPC) [4], [5] is an industry-standard *online* optimal control approach [6]; it provides one way to combine an inexact model with measured state information. MPC methods repetitively solve finite time horizon optimal control problems to approximate a computationally intractable optimal feedback controller [1]. Nominal MPC methods assume *no knowledge about the model uncertainty*. These methods, include quasi-infinite horizon MPC [7] which can be used to *efficiently* solve *control constrained* problems while possessing *inherent robustness* to model uncertainty [8]. Robust MPC methods presume some knowledge about the uncertainty. For example, min-max MPC [9], [10] minimizes the cost under worst-case uncertainty, while tube-based MPC [11]–[14] uses feedback to minimize the error between the measured trajectory and the predicted optimal trajectory, assuming a known worst-case uncertainty. *Iterative* MPC combines the idea of iterative learning control (ILC) [15], [16] with MPC to control repetitive tasks or iteratively executed non-repetitive tasks. The basic idea of ILC methods is to use the measured state information from the previous iteration to improve the controller for the next iteration. Classical ILC methods may be used without model information to solve tracking problems with a-priori known reference trajectory, but cannot be used to solve nonlinear optimal control problems where no reference trajectory is defined [17]. The combination of ILC and MPC was first proposed in [18], where the task was to track a reference trajectory, the measured state from the previous iteration was used to reduce the tracking error in the next iteration, and nominal linear [19], robust [20], and nonlinear [21] MPC was used to implement the online *feedback controller* in every iteration. More recently, iterative learning MPC has also been extended to solve a more general class of *reference free* infinite horizon online optimal control problems [22]. Common to all aforementioned approaches is that they solve the optimal control problem along the trajectory predicted by an inexact model which leads to increased cost and reduced robustness to model uncertainty.

In this paper, we propose a data-driven optimal feedback control method to solve *finite time horizon, nonlinear, input constrained optimal control problems*. The main contribution of the proposed method is the novel way it combines the inexact model with measured state information, which enables *efficient online* implementation of a *near-optimal feedback controller* such that *the inexact model is never used to make a future prediction*.

We present the first algorithmic implementation of the proposed method where *the model uncertainty is assumed unknown* (Section III, Algorithm 1). We show that, under common hypotheses on the system dynamics and the cost, the proposed algorithm: (i) converges with monotonically decreasing cost, and (ii) provides a locally optimal feedback controller that satisfies the necessary conditions of optimality along the measured trajectory of the controlled system as opposed to satisfying the same conditions along the trajectory predicted by the inexact model (Section IV, Theorems 1, 2, and 3).

The proposed algorithm can be used to implement control constrained online feedback controllers which reduce the user defined cost through repetitive execution of the task. Through repetition of the task, the controllers converge to a locally optimal feedback controller if the

Y. Chen is with the Singapore University of Technology and Design (SUTD), 8 Somapah Road, 487372 Singapore.

D. J. Braun is with the Department of Mechanical Engineering at Vanderbilt University.

* E-mail david.braun@vanderbilt.edu.

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

model is exact, or terminate with a near-optimal controller that reduces the cost along the measured trajectory if the model is inexact (Section V).

It has been recently shown [23] that a number of reinforcement learning methods [24]–[28] can approximately solve *control constrained* optimal control problems when the value function (optimal future cost) and the control policy (optimal feedback control law) are continuous. The key idea of the aforementioned infinite horizon methods is to learn the value function and the control policy online *along the trajectory of the real system* [28]. These methods use a non-quadratic control cost to replace hard control constraints with soft constraints, and consider *infinite horizon stabilization problems*, because the infinite horizon assumption, together with the assumption of a time-invariant dynamics, time-invariant running cost, and no terminal cost, ensure that the *value function and the control policy are time-invariant*, and require less data to learn compared to a time-varying value function and a time-varying control policy in finite horizon optimal control. Extensions of this idea to efficiently solve finite horizon unconstrained or control constrained stabilization problems remains challenging [29]–[34], because this extension requires considerable amount of measured data to approximate the optimal value function, co-state, and control policy [34]. Although the proposed online optimal control method is not a reinforcement learning method that attempts to learn the complex time and state dependent value function, co-state, or control policy from data, it extends the key idea of the aforementioned infinite horizon reinforcement learning methods to finite horizon control constrained problems, as it calculates the time-varying optimal control policy under hard control constraints along the measured trajectory of the controlled system.

The paper is structured as follows: In Section II, we present the optimal control problem. In Section III, we provide the optimal control algorithm. In Section IV, we establish the convergence and optimality properties of the proposed algorithm, first assuming that the model of the controlled system is exact, and then assuming that the model is inexact. In Section V, we solve three examples to demonstrate the reduced cost, the increased robustness to parametric model uncertainty, and the applicability of the proposed algorithm to online solve a complex optimal control problem with an unknown and changing dynamics. In Section VI we conclude with a brief summary of the results. Finally, in Section VII Appendix, we present the proofs supporting the results of this paper.

We use the following notation: Let $\mathbb{N}$ denote the natural numbers, $\mathbb{R}_+$ denote positive real numbers, $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ represent the sets of n dimensional vectors and $n \times m$ dimensional matrices, $\Re\lambda(\cdot)$ denote the real part of the eigenvalue of a matrix. Let $\mathbf{f}(\cdot)$ represent a function, $\mathbf{f}_\mathbf{x} = \partial\mathbf{f}(\mathbf{x},\cdot)/\partial\mathbf{x}$ denote the partial derivative of a function, and $\mathcal{I}[\cdot]$ represent a functional. Let $\|\mathbf{x}\|$ be the 2-norm of $\mathbf{x} \in \mathbb{R}^n$, $\|\mathbf{x}(\cdot)\|_{\mathcal{L}^p}$ be the p-norm of a Lebesgue integrable function $\mathbf{x}(\cdot) \in \mathcal{L}^p$, and let $\mathcal{B}_r(\mathbf{u}) = \{\mathbf{u}+\delta\mathbf{u} \in \mathbb{R}^m : \|\delta\mathbf{u}\| \leq r, r > 0\}$ denote a closed ball centered at $\mathbf{u}$ with radius r. Function arguments will be suppressed at places to simplify the presentation.

II. Problem Statement

We consider the Lagrange problem of optimal control[1] in continuous time

$$\begin{aligned} &\min_{\mathbf{u}(\cdot)\in\mathcal{U}} \mathcal{I}[\mathbf{u}(\cdot)] = \int_0^T \mathcal{L}(\mathbf{x}(t),\mathbf{u}(t),t)dt \\ &\text{subject to :}\quad \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t),\mathbf{u}(t),t) \text{ and } \mathbf{x}(0) = \mathbf{x}_0 \end{aligned} \tag{1}$$

where $t \in \mathbb{T} = [0,T]$ is the time interval, $T \in \mathbb{R}_+$ is the terminal time, $\mathbf{x}(t) = [x_1(t),...,x_n(t)]^\top \in \mathbb{R}^n$ is the state and $\mathbf{u}(t) = [u_1(t),...,u_m(t)]^\top \in \mathbb{R}^m$ is the control input at time t, $\mathbf{u}(\cdot)$ is the control function, $\mathcal{U}$ is the set of admissible control functions, $\mathcal{I}[\mathbf{u}(\cdot)]$ is the total cost, $\mathcal{L}(\mathbf{x}(t),\mathbf{u}(t),t)$ is the running cost, $\mathbf{f}(\mathbf{x}(t),\mathbf{u}(t),t)$ represents the dynamics of the controlled system while $\mathbf{x}_0$ is the initial state. The set of admissible control functions $\mathcal{U}$, is the set of measurable functions $\mathbf{u}(\cdot) : \mathbb{T} \to \mathbb{U}$ that satisfy

$$\forall t \in \mathbb{T} : \mathbf{u}(t) \in \mathbb{U} = \{\mathbf{u} \in \mathbb{R}^m : \mathbf{A}\mathbf{u} \preceq \mathbf{b}\} \tag{2}$$

where $\mathbf{A} \in \mathbb{R}^{p\times m}$ and $\mathbf{b} \in \mathbb{R}^m$ such that $\mathbb{U}$ is a compact, convex, and nonempty set [36].

We make the following standing assumptions:

Assumption 1 (Dynamics)**.** *(i) For any $\mathbf{u}(\cdot) \in \mathcal{U}$, there exists a uniformly bounded solution of $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t),\mathbf{u}(t),t)$; there exists $X \in (0,\infty)$ such that:*

$$\forall t \in \mathbb{T} : \mathbf{x}(t) \in \mathbb{X} = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\| \leq X\}.$$

(ii) $\mathbf{f}$, $\mathbf{f}_\mathbf{x}$ and $\mathbf{f}_\mathbf{u}$ are continuous in t and locally Lipschitz continuous in $(\mathbf{x},\mathbf{u})$ for all $(\mathbf{x},\mathbf{u},t) \in \mathbb{X}\times\mathbb{U}\times\mathbb{T}$.

Assumption 2 (Cost)**.** *(i) $\mathcal{L}$, $\mathcal{L}_\mathbf{x}$, $\mathcal{L}_\mathbf{u}$, $\mathcal{L}_{\mathbf{xx}}$, $\mathcal{L}_{\mathbf{xu}}$, $\mathcal{L}_{\mathbf{uu}}$ are continuous for all $(\mathbf{x},\mathbf{u},t) \in \mathbb{X}\times\mathbb{U}\times\mathbb{T}$.*

(ii) The running cost is convex with respect to $\mathbf{u}$:

$$\forall(\mathbf{x},\mathbf{u},t) \in \mathbb{X}\times\mathbb{U}\times\mathbb{T} : \mathcal{L}_{\mathbf{uu}}(\mathbf{x},\mathbf{u},t) \succeq \mathbf{0}.$$

Assumption 3 (Inexact dynamics)**.** *(i) The state $\mathbf{x}(t)$ of the system is measurable.*

(ii) The exact model of the system $\mathbf{f}$ is not known.

(iii) The inexact model of the system is defined by

$$\dot{\mathbf{x}}(t) = \hat{\mathbf{f}}(\mathbf{x}(t),\mathbf{u}(t),t). \tag{3}$$

(iv) Assumption 1 is valid for the inexact model.

(v) The error between $\mathbf{f}_\mathbf{x}$, $\mathbf{f}_\mathbf{u}$ and $\hat{\mathbf{f}}_\mathbf{x}$, $\hat{\mathbf{f}}_\mathbf{u}$ is bounded

$$\|\hat{\mathbf{f}}_\mathbf{x} - \mathbf{f}_\mathbf{x}\| \leq \epsilon_\mathbf{x}, \quad \|\hat{\mathbf{f}}_\mathbf{u} - \mathbf{f}_\mathbf{u}\| \leq \epsilon_\mathbf{u}$$

[1]The Bolza problem of optimal control with nonzero terminal cost, or the Mayer problem of optimal control with only terminal cost, can be transformed into a Lagrange problem under suitable regularity assumptions, see [35] (Chapter 1.9).

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

where for all $(\mathbf{x}, \mathbf{u}, t) \in \mathbb{X} \times \mathbb{U} \times \mathbb{T} : \epsilon_{\mathbf{x}}, \epsilon_{\mathbf{u}} \in [0, \infty)$.

Assumption 4 (Existence of the solution)**.** *The optimal control problem* (1) *has at least one solution.*

Assumptions 1–2 guarantee that there exist at least one feasible control function and state trajectory, and that the cost is bounded for all feasible control functions and state trajectories. These assumptions restrict the dynamics and the cost [37]; for example, dynamics with finite escape time, and running costs which are not convex with respect to controls, do not satisfy these assumptions. Nevertheless, Assumptions 1–2 do not guarantee existence of the optimal control. Existence of the optimal control, Assumptions 4, can be guaranteed by imposing extra conditions on the dynamics and the cost, see [38], [39] (Chapter 4) and [35] (Chapters 9 and 11-16).

III. Iterative Optimal Feedback Control

We propose a novel **I**terative **O**nline **O**ptimal **F**eedback **C**ontrol (**IOOFC**) algorithm to approximately solve the optimal control problem (1), using the inexact model of the system (3). Algorithm 1 belongs to the class of successive approximation methods [40], [41], which include differential dynamic programing [42]–[45], the iterative linear quadratic regulator [46], in addition to a number of strong-variation methods based on the maximum principle [17], [36], [47]–[50].

Similar to [17], [36], [40]–[50], Algorithm 1 iteratively solves an approximation of the original optimal control problem to calculate the control inputs that reduce the cost in every iteration, but different from the aforementioned methods, Algorithm 1 calculates the control inputs online, along the measured trajectory of the system, instead of calculating the control inputs using the trajectory predicted by the inexact model. Consequently, Algorithm 1 terminates with control that satisfies the local necessary conditions of optimality along the measured trajectory, instead of terminating with a controller that satisfies the same condition along the trajectory predicted by the inexact model.

The algorithm is described below in **Step 0–Step 4**. At every iteration denoted with superscript i, and every time step denoted with subscript j, the algorithm updates the control input from the previous iteration to obtain the control input in the next iteration $\mathbf{u}_j^i = \mathbf{u}_j^{i-1} + \delta\mathbf{u}_j^i \in \mathbb{U}$. The control update is a feedback control input $\delta\mathbf{u}_j^i = \delta\mathbf{u}^i(\mathbf{x}^i(t_j), t_j)$ calculated by minimizing a local approximation of the Hamiltonian (6) (quadratic with respect to control and linear with respect to the state) evaluated at the measured state of the system $\mathbf{x}^i(t_j)$. The approximation of the Hamiltonian with respect to the state is calculated using two linear differential equations (4), (5), solved between subsequent iterations. The algorithm relies on a convergence control method which ensures decreasing cost in every iteration and termination only if the necessary conditions of local optimality are satisfied along the measured trajectory of the controlled system (or if the initial control input cannot be improved).

Step 0 (Initialization):

Set $i = 0$. Choose a feasible initial control function $\mathbf{u}^0(\cdot) \in \mathcal{U}$. Execute this control on the system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}^0(t), t)$. Measure the corresponding state trajectory $\mathbf{x}^0(\cdot)$ and compute the cost $\mathcal{I}^0 = \mathcal{I}[\mathbf{u}^0(\cdot)]$ (1).

Step 1 (Offline computation):

Set $i = i + 1$ and $\alpha^i = 1$.

Compute $\mathbf{R}^{i-1}(\cdot)$, $\mathbf{r}^{i-1}(\cdot)$ by backward integration of

$$\dot{\mathbf{R}}^{i-1} + \mathbf{R}^{i-1}\hat{\mathbf{f}}_{\mathbf{x}} + \hat{\mathbf{f}}_{\mathbf{x}}^{\top}\mathbf{R}^{i-1} + \mathcal{L}_{\mathbf{xx}} = \mathbf{0}, \tag{4}$$

$$\dot{\mathbf{r}}^{i-1} + \mathcal{L}_{\mathbf{x}} + \hat{\mathbf{f}}_{\mathbf{x}}^{\top}\mathbf{r}^{i-1} = \mathbf{0}, \tag{5}$$

with terminal condition $\mathbf{R}^{i-1}(T) = \mathbf{0}$ and $\mathbf{r}^{i-1}(T) = \mathbf{0}$. In (4) and (5), the partial derivatives $\mathcal{L}_{\mathbf{x}}$, $\mathcal{L}_{\mathbf{xx}}$, $\hat{\mathbf{f}}_{\mathbf{x}}$ are evaluated along the control function $\mathbf{u}^{i-1}(\cdot)$ applied in previous iteration, and the state trajectory $\mathbf{x}^{i-1}(\cdot)$ *measured* in previous iteration.

Step 2a (Online computation at $t = t_j$):

At the current i^{th} iteration, and j^{th} time point $t = t_j = j\frac{T}{n_T} \in [t_0 = 0, t_1, t_2, ..., t_{n_T} = T]$, measure the state $\mathbf{x}_j^i = \mathbf{x}^i(t_j)$, and calculate the corresponding *feedback* control update by solving the constrained quadratic program (cQP)

$$\begin{aligned}\delta\mathbf{u}_j^i &= \delta\mathbf{u}(\mathbf{x}_j^i, t_j) \\ &= \underset{\delta\mathbf{u}\in\delta\mathbb{U}_j^i}{\operatorname{argmin}} \frac{1}{2}\delta\mathbf{u}^{\top}\mathcal{L}_{\mathbf{uu}}^{\mathrm{R}}\delta\mathbf{u} + \alpha^i\delta\mathbf{u}^{\top}\mathbf{b}(\mathbf{x}_j^i, t_j),\end{aligned} \tag{6}$$

$$\mathbf{b}(\mathbf{x}_j^i, t_j) = (\mathcal{L}_{\mathbf{ux}} + \hat{\mathbf{f}}_{\mathbf{u}}^{\top}\mathbf{R}_j^{i-1})(\mathbf{x}_j^i - \mathbf{x}_j^{i-1}) + \mathcal{L}_{\mathbf{u}} + \hat{\mathbf{f}}_{\mathbf{u}}^{\top}\mathbf{r}_j^{i-1}$$

where $\mathbf{x}_j^{i-1} = \mathbf{x}^{i-1}(t_j)$ is the state at the previous iteration and current time point t_j, $\mathcal{L}_{\mathbf{uu}}^{\mathrm{R}}$ is a *strictly* positive definite Hessian calculated by regularized Cholesky factorization of $\mathcal{L}_{\mathbf{uu}}$ [51] (Chapter 3.4)

$$\mathcal{L}_{\mathbf{uu}}^{\mathrm{R}} = \mathrm{rCholesky}(\mathcal{L}_{\mathbf{uu}}) \succeq \mathbf{I}\lambda_{\min}^i \text{ where } \lambda_{\min}^i > 0, \tag{7}$$

$\delta\mathbb{U}_j^i$ is the admissible set of control updates

$$\delta\mathbb{U}_j^i = \{\delta\mathbf{u} \in \mathbb{R}^m : \mathbf{A}\delta\mathbf{u} \leq \mathbf{b} - \mathbf{A}\mathbf{u}_j^{i-1}\},$$

and $\alpha^i \in (0, 1]$ is a convergence control parameter. (The convergence control parameter is set to $\alpha^i = 1$ in Step 1, and is modified $\alpha^i \in (0, 1]$ in Step 4, if such modification is required to decrease the cost.)

Step 2b (Online control of the system at $t = t_j$):

Update the control input

$$\mathbf{u}_j^i = \mathbf{u}_j^{i-1} + \delta\mathbf{u}_j^i = \mathbf{u}^{i-1}(t_j) + \delta\mathbf{u}(\mathbf{x}^i(t_j), t_j),$$

and use the updated input to control the system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}_j^i, t_j)$. Proceed to the next time point $t = t_{j+1}$. Repeat Step 2a and Step 2b until $t = T$.

Step 3 (Convergence control):

Calculate the reduction of the cost in the current iteration

$$\Delta\mathcal{I}^i = \mathcal{I}[\mathbf{u}^i(\cdot)] - \mathcal{I}[\mathbf{u}^{i-1}(\cdot)] \tag{8}$$

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

and check if the following condition is satisfied

$$\Delta\mathcal{I}^i \leq -c\|\delta\mathbf{u}^i(\cdot)\|^2_{\mathcal{L}^2} \tag{9}$$

where $c \in (0, \infty)$ is a user defined positive number.

If (9) is satisfied then store the control $\mathbf{u}^i(\cdot)$, the state $\mathbf{x}^i(\cdot)$, and the cost $\mathcal{I}^i$, and proceed to Step 4.

If (9) is not satisfied then use backtracking [51] (Chapter 3.1) to decrease the convergence control parameter $\alpha^i = \gamma\alpha^i \in (0, 1]$ where $\gamma \in (0, 1)$ is a pre-defined constant, and go to Step 2a.

Step 4 (Stopping criterion): Terminate the algorithm if there exist i^* such that for all $i \geq i^* : \|\delta\mathbf{u}^i(\cdot)\|^2_{\mathcal{L}^2} < \epsilon$. Otherwise set $i = i + 1$ and go to Step 1.

The pseudo-code is given in Algorithm 1.

Algorithm 1: Iterative Online Optimal Feedback Control

Input: Feasible $\mathbf{u}^0(\cdot) \in \mathbb{U}$, system, model $\hat{\mathbf{f}}(\mathbf{x}, \mathbf{u}, t)$
Output: $\mathbf{u}^*(\cdot)$
Initialize: $cnvg = false$, $i = 1$, $c \in (0, \infty)$, $\gamma \in (0, 1)$, $0 < \epsilon \ll 1$, $\Delta i \in \mathbb{N}$, $\mathbf{x}^0 \leftarrow \big(\text{system}, \mathbf{u}^0(\cdot)\big)$, $\mathcal{I}^0 \leftarrow \mathcal{I}[\mathbf{u}^0(\cdot)]$
while $conv = false$ **do**
  for $j = n_T : -1 : 2$ **do**
    Evaluate: $\hat{\mathbf{f}}_{\mathbf{x}}$, $\mathcal{L}_{\mathbf{x}}$, $\mathcal{L}_{\mathbf{xx}} \leftarrow (\mathbf{x}_j^{i-1}, \mathbf{u}_j^{i-1}, t_j)$
    Offline compute: $\mathbf{R}_{j-1}^{i-1}$, $\mathbf{r}_{j-1}^{i-1} \leftarrow$ (4), (5)
  end
  $\alpha^i = 1$, $iter = true$
  while $iter = true$ **do**
    for $j = 1 : n_T$ **do**
      Measure: $\mathbf{x}_j^i \leftarrow$ system
      Evaluate: $\hat{\mathbf{f}}_{\mathbf{u}}$, $\mathcal{L}_{\mathbf{u}}$, $\mathcal{L}_{\mathbf{ux}}, \mathcal{L}_{\mathbf{uu}} \leftarrow (\mathbf{x}_j^{i-1}, \mathbf{u}_j^{i-1}, t_j)$
      Regularize: $\mathcal{L}^{\mathbf{R}}_{\mathbf{uu}} = \text{rCholesky}(\mathcal{L}_{\mathbf{uu}})$
      Online compute: $\delta\mathbf{u}_j^i \leftarrow$(6)
      Online control: system $\leftarrow \mathbf{u}_j^i = \mathbf{u}_j^{i-1} + \delta\mathbf{u}_j^i$
    end
    Convergence control: $\Delta\mathcal{I}^i \leftarrow$ (8)
    if $\Delta\mathcal{I}^i \leq -c\|\delta\mathbf{u}^i(\cdot)\|^2_{\mathcal{L}^2}$ **then**
      $iter = false$
    else
      Backtrack: $\alpha^i = \gamma\alpha^i$
    end
  end
  if $\forall i \in \{i^*, i^* + \Delta i\} : \|\delta\mathbf{u}^i(\cdot)\|^2_{\mathcal{L}^2} < \epsilon$ **then**
    $\mathbf{u}^* = \mathbf{u}^i(\cdot)$, $cnvg = true$
  end
  $i = i + 1$
end

The following three remarks summarize the main features of Algorithm 1.

Remark 1 (Online optimal control using measured states)**.** *The feedback control input is obtained by using the measured state in the current iteration when solving cQP* (6) *while the time dependent parameters* $\mathbf{R}, \mathbf{r}$ *in cQP* (6) *are obtained by solving the linear ordinary differential equations* (4) *and* (5) *along the measured trajectory in the previous iteration. The key advantage of this approach is that the inexact model* (3) *is never used to make a future prediction; only the parameters of the inexact model are used:* $\hat{\mathbf{f}}_{\mathbf{u}}$ *is used as a parameter in the cQP* (6) *while* $\hat{\mathbf{f}}_{\mathbf{x}}$ *is used as a parameter in the linear differential equations* (4) *and* (5)*.*

According to Remark 1, the proposed algorithm bypasses the inevitable error accumulation seen in model predictive control that uses inexact model-based predicted trajectories to calculate the control inputs. Recent tube-based robust model predictive control methods promote the use of feedback to mitigate this limitation [11]–[14]. In our formulation, feedback is reserved to address unforeseen perturbations, while online optimization along the measured trajectory of the system is used to mitigate the detrimental effect of inexact model-based future prediction.

Remark 2 (Terminal solution)**.** *Unless the initial user defined control input* $\mathbf{u}^0(\cdot)$ *cannot be improved, the exit criterion (9) and the adaptation of the convergence control parameter* $\alpha \in (0, 1]$ *in* **Step 3** *ensure that the cost decreases in every iteration until the necessary conditions of local optimality are satisfied along the measured trajectory of the controlled system. When the algorithm terminates, the measured trajectory does not change anymore, and the parameters of the cQP obtained in the previous iteration* $\mathbf{R}, \mathbf{r}$*, are the optimal parameters for the next iteration. Consequently, Algorithm 1 provides an online computed locally optimal constrained feedback controller when the model is exact, and an approximately optimal constrained feedback controller when the model is inexact. In both cases, the control constraints are rigorously taken into account.*

The convergence and optimality properties of the algorithm are detailed in Section IV.

Remark 3 (Computational cost)**.** *The proposed algorithm has both online and offline computational costs which depend on the number of states* n*, number of control inputs* m*, and the number of discrete time points* n_T *used to discretize the time horizon (see* **Step 2a** *and Algorithm 1).*

(i) The online computational cost comes from solving the cQP (6) *forward in time. The worst-case computational cost per time step in solving* (6) *is [51]*

$$\frac{\textit{Online cost}}{\textit{Time step}} \propto \mathcal{O}(m^3). \tag{10}$$

(ii) The offline computational cost comes from solving the linear differential equations (4) *and* (5) *between itera-*

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

tions. The worst-case computational cost of solving these equations is

$$\frac{\textit{Offline cost}}{\textit{Iteration}} \propto \mathcal{O}(n_T n^2). \tag{11}$$

Remark 3 shows that the online computational cost (10) does not depend on the time horizon. This means lower online computational cost compared to nonlinear model predictive control, where the online computational cost at least linearly scales with the reduced time horizon optimal control problem solved at each time step [48], [52]–[54]: *Online cost/Time step* $\propto \mathcal{O}(n_t m^3)$ where $1 \ll n_t \leq n_T$. We further note that similar to offline successive approximation methods [36], [49], the proposed online algorithm is not subject to the "curse of dimensionality" becasue the offline computational cost (11) of solving two linear differential equations (4) and (5) quadratically scales with the number of states, as opposed to the exponentially scaling in dynamic programming. Due to both (10) and (11), Algorithm 1 is suitable for iterative online control of high-dimensional systems using large number of control inputs (see Sections V-B and V-C).

Finally, we note that Algorithm 1 presents a discrete time computation of the control inputs implementable on sample time digital control systems. Although, this discrete time controller does not provide the exact solution of the original continuous time optimal control problem (1), it can provide an approximate implementation of the continuous time controller under fast enough sampling, and sufficient regularity of the closed-loop dynamics [55]. Discretization of continuous time controllers is known as controller emulation [56] which is a standard approach in MPC setting, see [8] (Remark 22).

IV. Convergence and Local Optimality

Proving convergence of successive approximation methods [17], [36], [40]–[49] has been nontrivial. One of the first convergence proofs was given in [57] for linear quadratic optimal control problems. Convergence proofs for methods applicable to more general classes of constrained nonlinear optimal control problems are given for dynamic programing based methods [43], [45], [58] and also for methods based on the maximum principle [36], [49].

In this section, we prove the convergence of Algorithm 1 when the model is exact, and we provide a sufficient conditions for Algorithm 1 to terminate with a near-optimal controller, that satisfies the local necessary conditions of optimality along the measured trajectory of the controlled system, when the model is inexact. We will characterize the local optimality of the control inputs in both cases. The main results are summarized in Theorems 1–3. The following two propositions will be used to prove these Theorems. We will use $\mathbf{u}^i = \mathbf{u}^i(\cdot)$ and $\mathbf{x}^i = \mathbf{x}^i(\cdot)$ to simplify the notation.

Proposition 1 (Relation between the cost variation and the control update)**.** *At every iteration $i \in \mathbb{N}$ of Algorithm 1, there exist $c_{\mathbf{u}} \in (0, \infty)$ and $c_\epsilon \in [0, \infty)$, independent of $i \in \mathbb{N}$, such that the variation of the cost satisfies*

$$\begin{aligned} \Delta\mathcal{I}^i =& \mathcal{I}[\mathbf{u}^i] - \mathcal{I}[\mathbf{u}^{i-1}] \\ \leq& -(\frac{\lambda^i_{\min}}{\alpha^i} - c_{\mathbf{u}})\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2} + c_\epsilon\|\delta\mathbf{u}^i\|_{\mathcal{L}^1} \end{aligned} \tag{12}$$

where $\delta\mathbf{u}^i = \mathbf{u}^i - \mathbf{u}^{i-1}$ is the control update, $\alpha^i \in (0, 1]$ is the convergence control parameter in (6) *while $\lambda^i_{\min} > 0$ is the smallest eigenvalue of the regularized Hessian of the running cost $\mathcal{L}^R_{\mathbf{uu}}$ in* (7)*.*

Proof. See Appendix VII-B. ■

Proposition 2 (Relation between the cost variation and the model error)**.** *The constants $c_{\mathbf{u}}(\epsilon_{\mathbf{x}}, \epsilon_{\mathbf{u}}) \in (0, \infty)$ and $c_\epsilon(\epsilon_{\mathbf{x}}, \epsilon_{\mathbf{u}}) \in [0, \infty)$ in* (12) *are monotonically increasing with respect to the model error $\epsilon_{\mathbf{x}}$ and $\epsilon_{\mathbf{u}}$ defined in Assumption 3.*

Proof. See Appendix VII-C. ■

Remark 4. *When the model is exact, $c_\epsilon = 0$,* (12) *reduces to a quadratic inequality between the cost variation and the control update, which is similar to (27) in [36]. Such quadratic inequality relation was used to prove the convergence of the successive approximation based optimal control algorithm presented in [36].*

A. Exact model

The following Theorem proves the convergence of Algorithm 1 when the model is exact.

Theorem 1 (Convergence)**.** *When the model is exact $\epsilon_{\mathbf{x}} = \epsilon_{\mathbf{u}} = 0$, there exist a sequence of convergence control parameters $\{\alpha^i\} \in (0, 1]$ such that:*

(i) The sequence of state trajectories and control functions $\{\mathbf{x}^i\}$, $\{\mathbf{u}^i\}$ converge to $\mathbf{x}^\infty$, $\mathbf{u}^\infty$ in $\mathcal{L}^2$:

$$\lim_{i\to\infty}\|\mathbf{u}^i - \mathbf{u}^\infty\|_{\mathcal{L}^2} = 0, \quad \lim_{i\to\infty}\|\mathbf{x}^i - \mathbf{x}^\infty\|_{\mathcal{L}^2} = 0.$$

(ii) The cost $\{\mathcal{I}^i\}$ decreases monotonically and converges:

$$\forall i \in \mathbb{N} : \ \mathcal{I}^i > \mathcal{I}^{i+1} \ \text{ and } \ \mathcal{I}^0 \geq \lim_{i\to\infty}\mathcal{I}^i \to \mathcal{I}^\infty > -\infty.$$

Proof. According to Proposition 2, $c_{\mathbf{u}}$ and c_ϵ assume their minimum values when the model is exact

$$\epsilon_{\mathbf{x}} = \epsilon_{\mathbf{u}} = 0 : c_{\mathbf{u}} = c_{\mathbf{u}\min} > 0 \ \text{ and } \ c_\epsilon = 0.$$

Consequently, (12) in Proposition 1 reduces to

$$\Delta\mathcal{I}^i \leq -(\frac{\lambda^i_{\min}}{\alpha^i} - c_{\mathbf{u}\min})\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2}.$$

(i) Given a small enough convergence control parameter

$$\forall i \in \mathbb{N} : \alpha^i \in \left(0, \frac{\lambda^i_{\min}}{c_{\mathbf{u}\min} + c}\right] \cap (0, 1],$$

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

the cost in (1) will decrease because

$$\forall i \in \mathbb{N} : \Delta\mathcal{I}^i \leq -c\|\delta\mathbf{u}^i\|_{\mathcal{L}^2}^2 \tag{13}$$

where $\Delta\mathcal{I}^i = 0$ only if

$$\|\delta\mathbf{u}^i\|_{\mathcal{L}^2} = \|\mathbf{u}^i - \mathbf{u}^{i-1}\|_{\mathcal{L}^2} = 0.$$

Summing up the cost decrements in (13), we obtain

$$\sum_{i=0}^{\infty}\|\delta\mathbf{u}^i\|_{\mathcal{L}^2}^2 \leq c^{-1}\sum_{i=0}^{\infty} -\Delta\mathcal{I}^i \leq c^{-1}\big(\mathcal{I}^0 - \inf_{\mathbf{u}(\cdot)\in\mathcal{U}}\mathcal{I}\big).$$

Assumptions 1 and 2 together with $c > 0$ guarantee that the cost is bounded

$$-\infty < \inf_{\mathbf{u}(\cdot)\in\mathcal{U}}\mathcal{I} \leq \mathcal{I}^0 < \infty. \tag{14}$$

Consequently,

$$\sum_{i=0}^{\infty}\|\delta\mathbf{u}^i\|_{\mathcal{L}^2}^2 < \infty \Rightarrow \lim_{i\to\infty}\|\delta\mathbf{u}^i\|_{\mathcal{L}^2}^2 = 0. \tag{15}$$

Based on (15), the control update converges to zero in $\mathcal{L}^2$. Furthermore, according to Proposition 8 in Appendix VII-A,

$$\|\delta\mathbf{x}^i\|_{\mathcal{L}^2}^2 \leq \frac{1}{2}c_1^2T^2e^{2c_1T}\|\delta\mathbf{u}^i\|_{\mathcal{L}^2}^2$$

where $c_1 \in (0,\infty)$ is a constant independent of $i \in \mathbb{N}$. Consequently,

$$\lim_{i\to\infty}\|\delta\mathbf{u}^i\|_{\mathcal{L}^2} = 0, \quad \lim_{i\to\infty}\|\delta\mathbf{x}^i\|_{\mathcal{L}^2} = 0.$$

We conclude that $\{\mathbf{x}^i\}$, $\{\mathbf{u}^i\}$ converge to $\mathbf{x}^\infty$, $\mathbf{u}^\infty$ in $\mathcal{L}^2$. This proves the first part of the theorem.

(ii) The second part of the theorem follows from (13) and (14); the cost sequence $\{\mathcal{I}^i\}$ is monotonically decreasing and is bounded from below, thereby it converges. ■

The next Theorem proves local optimality of the constrained feedback controller provided by Algorithm 1.

Theorem 2 (Optimality)**.** *When the model is exact $\epsilon_{\mathbf{x}} = \epsilon_{\mathbf{u}} = 0$, Algorithm 1 converges to $\mathbf{x}^\infty$ and $\mathbf{u}^\infty$ that satisfy the first order necessary conditions of local optimality of the optimal control problem (1).*

Proof. When the model is exact $\mathbf{f} = \hat{\mathbf{f}}$, the Hamiltonian of the optimal control problem (1) is given by

$$\mathcal{H}(\mathbf{x},\mathbf{u},\boldsymbol{\lambda},t) = \mathcal{L}(\mathbf{x},\mathbf{u},t) + \boldsymbol{\lambda}^\top\mathbf{f}(\mathbf{x},\mathbf{u},t)$$

where $\boldsymbol{\lambda}$ denotes the co-state. The first order necessary conditions of local optimality for (1) are given by [2]:

$$\dot{\boldsymbol{\lambda}}^{\text{opt}} = -\mathcal{H}_{\mathbf{x}}(\mathbf{x}^{\text{opt}},\mathbf{u}^{\text{opt}},\boldsymbol{\lambda}^{\text{opt}},t),\ \boldsymbol{\lambda}^{\text{opt}}(T) = \mathbf{0} \tag{16}$$

and

$$\begin{aligned}&\exists r > 0, \forall\mathbf{u} \in \mathcal{B}_r(\mathbf{u}^{\text{opt}}) \cap \mathbb{U} :\\ &\mathcal{H}_{\mathbf{u}}(\mathbf{x}^{\text{opt}},\mathbf{u}^{\text{opt}},\boldsymbol{\lambda}^{\text{opt}},t)(\mathbf{u}-\mathbf{u}^{\text{opt}}) \geq 0.\end{aligned} \tag{17}$$

When Algorithm 1 converges $i \to \infty$, equation (5) becomes

$$\dot{\mathbf{r}}^\infty = -\mathcal{H}_{\mathbf{x}}(\mathbf{x}^\infty,\mathbf{u}^\infty,\mathbf{r}^\infty,t),\ \ \mathbf{r}^\infty(T) = \mathbf{0} \tag{18}$$

while the constrained quadratic program (6) implies [59]:

$$\begin{aligned}&\forall\mathbf{u} \in \mathcal{B}_r(\mathbf{u}^\infty) \cap \mathbb{U} :\\ &\frac{\partial[\frac{1}{2}\delta\mathbf{u}^\top\mathcal{L}^{\text{R}}_{\mathbf{uu}}\delta\mathbf{u} + \alpha^\infty\delta\mathbf{u}^\top\mathbf{b}(\mathbf{x}^\infty,t)]}{\partial\delta\mathbf{u}}\bigg|_{\delta\mathbf{u}=\mathbf{0}}(\mathbf{u}-\mathbf{u}^\infty) \geq 0.\end{aligned} \tag{19}$$

Because $\alpha^\infty \in (0,1]$, and $\mathcal{H}_{\mathbf{u}}(\mathbf{x}^\infty,\mathbf{u}^\infty,\mathbf{r}^\infty,t) = \mathcal{L}_{\mathbf{u}} + \mathbf{f}_{\mathbf{u}}^\top\mathbf{r}^\infty = \mathbf{b}(\mathbf{x}^\infty,t)$, (19) is equivalent to

$$\mathcal{H}_{\mathbf{u}}(\mathbf{x}^\infty,\mathbf{u}^\infty,\mathbf{r}^\infty,t)(\mathbf{u}-\mathbf{u}^\infty) \geq 0. \tag{20}$$

Comparing (16), (17) with (18), (20), we conclude

$$\|\boldsymbol{\lambda}^{\text{opt}} - \mathbf{r}^\infty\|_{\mathcal{L}^2} = 0,$$
$$\|\mathbf{u}^{\text{opt}} - \mathbf{u}^\infty\|_{\mathcal{L}^2} = 0, \quad \|\mathbf{x}^{\text{opt}} - \mathbf{x}^\infty\|_{\mathcal{L}^2} = 0.$$

Consequently, $\mathbf{x}^\infty$ and $\mathbf{u}^\infty$ satisfy the local necessary conditions of optimality for (1) in $\mathcal{L}^2$. ■

B. Inexact model

The following Proposition provides the relation between the control update $\delta\mathbf{u}$ and the convergence control parameter α, as $\alpha \to 0^+$. This relation will be used to investigate the properties of Algorithm 1 when the model is inexact.

Proposition 3 (Relation between the control update and the convergence control parameter)**.** *When $\alpha^i \to 0^+$, there is a linear relation between the control update $\delta\mathbf{u}^i$ defined by the cQP* (6) *and the convergence control parameter*

$$\|\delta\mathbf{u}^i\|_{\mathcal{L}^1} = c^i_{\delta\mathbf{u}}\alpha^i \tag{21}$$

where $c^i_{\delta\mathbf{u}} \in [0,\infty)$ is a constant that may change in each iteration. (ii) If $\Delta\mathcal{I}^i \neq 0$ then $c^i_{\delta\mathbf{u}} \neq 0$.

Proof. See Appendix VII-D. ■

The following Theorem characterizes the termination and sub-optimality of Algorithm 1.

Theorem 3 (Termination and sub-optimality)**.** *If the model is inexact $\epsilon_{\mathbf{x}} \neq 0$, $\epsilon_{\mathbf{u}} \neq 0$, and the model error is bounded by $0 < c_\epsilon(\epsilon_{\mathbf{x}},\epsilon_{\mathbf{u}}) \leq c^i_{\epsilon\max} = \lambda^i_{\min}c^i_{\delta\mathbf{u}}T^{-1}$ then there exists a sequence of convergence control parameters $\{\alpha^i\} \in (0,\alpha^i_{\max}] \cap (0,1]$ where $\alpha^i_{\max} > 0$, such that:*

(i) Algorithm 1 terminates with a feasible control after finitely many iterations:

$$\lim_{i\to i^*+1}\|\mathbf{u}^i - \mathbf{u}^*\|_{\mathcal{L}^2}^2 = 0, \ \ \mathbf{u}^* \in \mathcal{U}.$$

(ii) The cost $\{\mathcal{I}^i\}$ decreases monotonically and terminates at a cost $\mathcal{I}^*$ smaller or equal to the cost at initialization:

$$\forall i \leq i^* :\ \mathcal{I}^i > \mathcal{I}^{i+1} \ \text{ and } \ \mathcal{I}^0 \geq \lim_{i\to i^*+1}\mathcal{I}^i \to \mathcal{I}^* > -\infty.$$

(iii) When Algorithm 1 terminates, $\mathbf{x}^*$ and $\mathbf{u}^*$ satisfy the first order conditions of optimality defined using the

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

inexact model $\hat{\mathbf{f}}$ (3) *and evaluated along the measured trajectory* $\mathbf{x}^*$ *of the controlled system* $\mathbf{f}$ (1).

Proof. According to Proposition 2, $c_{\mathbf{u}}$ and c_ϵ are monotonically increasing functions of the model error. Therefore, these parameters do not assume their minimum values and are both nonzero in (12):

$$\epsilon_{\mathbf{x}} \neq 0, \epsilon_{\mathbf{u}} \neq 0 : c_{\mathbf{u}} > c_{\mathbf{u}\min} > 0 \text{ and } c_\epsilon > 0.$$

We use Schwarz's inequality to obtain

$$\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^1} \leq T\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2}. \tag{22}$$

Using (22) in (12), and assuming a sufficiently small convergence control parameter $0 < (c + c_{\mathbf{u}})\alpha^i < \lambda^i_{\min}$, we obtain

$$\Delta\mathcal{I}^i + c\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2} \leq -(\frac{\lambda^i_{\min}}{\alpha^i} - c_{\mathbf{u}} - c)\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2} + c_\epsilon\|\delta\mathbf{u}^i\|_{\mathcal{L}^1}$$
$$\leq -\frac{1}{T}(\frac{\lambda^i_{\min}}{\alpha^i} - c_{\mathbf{u}} - c)\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^1} + c_\epsilon\|\delta\mathbf{u}^i\|_{\mathcal{L}^1} \leq 0. \tag{23}$$

(i) According to (23), if there exists

$$\alpha^i \in (0, \alpha^i_{\max}] \cap (0, 1], \tag{24}$$

where

$$\alpha^i_{\max} = \frac{\lambda^i_{\max}\|\delta\mathbf{u}^i\|_{\mathcal{L}^1}}{(c_{\mathbf{u}} + c)\|\delta\mathbf{u}^i\|_{\mathcal{L}^1} + c_\epsilon T} > 0, \tag{25}$$

the cost will decrease

$$\Delta\mathcal{I}^i \leq -c\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2} \tag{26}$$

and $\Delta\mathcal{I}^i = 0$ only holds if

$$\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2} = \|\mathbf{u}^i - \mathbf{u}^{i-1}\|^2_{\mathcal{L}^2} = 0.$$

The sufficient condition for (24), (25) and (26) to hold is

$$0 < c_\epsilon(\epsilon_{\mathbf{u}}, \epsilon_{\mathbf{x}}) \leq c^i_{\epsilon\max} = \frac{1}{T}\left(\frac{\lambda^i_{\min}}{\alpha^i} - c_{\mathbf{u}} - c\right)\|\delta\mathbf{u}^i\|_{\mathcal{L}^1}. \tag{27}$$

According to Proposition 3, when $\alpha^i \to 0^+$, the model error (27) is limited by

$$0 < c_\epsilon(\epsilon_{\mathbf{u}}, \epsilon_{\mathbf{x}}) \leq \lim_{\alpha\to 0^+} c^i_{\epsilon\max} = \frac{1}{T}\lambda^i_{\min} c^i_{\delta\mathbf{u}}.$$

Finally, according to Proposition 2, $c_\epsilon(\epsilon_{\mathbf{u}}, \epsilon_{\mathbf{x}})$ is a monotonically increasing function of the model error. Consequently, (27) defines a limit to $\epsilon_{\mathbf{x}}$ and $\epsilon_{\mathbf{u}}$ that guarantees decrease of the cost in the i^{th} iteration. This proves the first part of the theorem.

(ii) The second part of the theorem follows from (26).

(iii) When Algorithm 1 terminates at $i > i^*$, the measured trajectory $\mathbf{x}^*$ and the corresponding feedback control function $\mathbf{u}^*$ satisfy the following conditions

$$\dot{\mathbf{x}}^* = \mathbf{f}(\mathbf{x}^*, \mathbf{u}^*, t), \tag{28}$$
$$\dot{\mathbf{r}}^* = -\hat{\mathcal{H}}_{\mathbf{x}}(\mathbf{x}^*, \mathbf{u}^*, \mathbf{r}^*, t), \quad \mathbf{r}^*(T) = \mathbf{0},$$
$$\exists r > 0, \forall \mathbf{u} \in \mathcal{B}_r(\mathbf{u}^*) \cap \mathbb{U} : \hat{\mathcal{H}}_{\mathbf{u}}(\mathbf{x}^*, \mathbf{u}^*, \mathbf{r}^*, t)(\mathbf{u} - \mathbf{u}^*) \geq 0$$

where

$$\hat{\mathcal{H}}(\mathbf{x}, \mathbf{u}, \mathbf{r}, t) = \mathcal{L}(\mathbf{x}, \mathbf{u}, t) + \mathbf{r}^\top\hat{\mathbf{f}}(\mathbf{x}, \mathbf{u}, t)$$

is the Hamiltonian, $\hat{\mathbf{f}}$ is the inexact model (3), and $\mathbf{r}$ is defined by (5). We note that (28) provides the first order necessary conditions of optimality derived using the inexact model (3) but evaluated along the measured trajectory $\mathbf{x}^*$ of the controlled system $\mathbf{f}$. ■

Theorem 3 (iii) implies that the control input at termination $\mathbf{u}^*$ differs from the offline computed inexact model-based optimal control input, becasue $\mathbf{u}^*$ satisfies the co-state equation in (28) along the measured trajectory of the system $\mathbf{f}$ as opposed to satisfying the co-state equation along the trajectory predicted by the inexact model $\hat{\mathbf{f}}$. The implication of this feature is shown by three examples in Section V.

V. Evaluation

We solve three examples using Algorithm 1.

(1) In Section V-A, we solve a scalar linear quadratic regulator (LQR) with an inexact model. We use this example to analytically show that the online computed feedback controller reduces the cost and improves robustness to parametric model uncertainty compared to the *offline computed optimal feedback controller*, when both controllers use the same inexact model.

(2) In Section V-B, we solve fifty large scale, finite time horizon, constrained linear quadratic regulators (cLQR) using inexact (open-loop unstable) models. In these examples, Algorithm 1 simultaneously reduces the cost, and has lower online computationally requirement than an MPC method developed to solve cLQR problems.

(3) In Section V-C, we solve a complex optimal control problem with unknown and changing dynamics. We use this example to demonstrate the usefulness of the proposed online optimal control method in a challenging application.

In all three examples we compare the solution provided by Algorithm 1 to the *ideal benchmark*, which is the optimal solution obtained using the exact model.

A. Linear Quadratic Regulator

Consider a scalar unconstrained linear quadratic regulator [60]–[62]

$$\min_{u(\cdot)} \mathcal{I}[u(\cdot)] = \min_u \frac{1}{2}\int_0^\infty (qx^2 + u^2)dt \tag{29}$$
$$\text{subject to: } \dot{x} = f(x, u) = ax + bu,\ x(0) = x_0 \tag{30}$$

where $x \in \mathbb{R}$ is the state, $u \in \mathbb{R}$ is the control input, $q \geq 0$ is the weight of the state in the running cost while $a, b \neq 0 \in \mathbb{R}$ are the model parameters.

Assumption 5 (Inexact model)**.** *The exact model of the system* (30) *is not known. The inexact model is given by*

$$\dot{x} = \hat{f}(x, u) = \hat{a}x + \hat{b}u \tag{31}$$

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

where $\hat{a}, \hat{b} \neq 0 \in \mathbb{R}$.

Next, we analytically derive the controller which can be online computed using Algorithm 1. We define the Hamiltonian function using the inexact model

$$\hat{\mathcal{H}} = \frac{1}{2}qx^2 + \frac{1}{2}u^2 + r(\hat{a}x + \hat{b}u)$$

and derive the co-state equation and the control input

$$\dot{r} = -\hat{\mathcal{H}}_x = -qx - \hat{a}r, \quad \lim_{t\to\infty} r(t) = 0,$$
$$u^* = \underset{u}{\operatorname{argmin}}\, \hat{\mathcal{H}} = \underset{u}{\operatorname{argmin}} \left[\frac{1}{2}u^2 + r\hat{b}u\right]$$

where the state x is measured from the system (30). The analytical solution of this problem is given by

$$u^*(x) = -k^*x = -k(a, b, q; \hat{a}, \hat{b})x \tag{32}$$

where

$$k(a, b, q; \hat{a}, \hat{b}) = \frac{1}{b}\left(\frac{a+\hat{a}}{2} + \sqrt{\left(\frac{a+\hat{a}}{2}\right)^2 + \hat{b}bq}\right). \tag{33}$$

The exact model-based controller is given by

$$u^{\text{opt}}(x) = -k_{\text{opt}}x = -k(a, b, q; a, b)x. \tag{34}$$

The inexact model-based optimal controller is given by

$$\hat{u}(x) = -\hat{k}x = -k(\hat{a}, \hat{b}, q; \hat{a}, \hat{b})x. \tag{35}$$

Remark 5. *Controller* (32) *requires the exact parameters of the system* a *and* b*, and as such, it cannot be directly implemented in practice. However, controller* (32) *can be online computed using Algorithm 1 which requires the measured state of the system* x *and the inexact model* (31) *but does not require the exact parameters* a *and* b*, or the exact model of the system (30). This makes* (32) *online computable and implementable in practice.*

Figure 1 shows the aforementioned three controllers together with the solution computed using Algorithm 1 (black line) and the solution computed using the MPC method [63] (gray line). Figure 1 shows that the MPC method recovers the inexact model-based optimal controller (35), as ideally expected. According to Fig. 1, Algorithm 1 is more advantageous than the inexact model-based optimal controller (35) in this example.

Figure 2 shows the cost of the three controllers (32), (34) and (35), together with the cost obtained using Algorithm 1 (black dots) and the MPC method [63] (gray dots). The MPC method behaves as ideally expected; it recovers the cost of the offline computed inexact model-based optimal controller (35). We also observe that Algorithm 1 significantly improves the robustness of the closed-loop system to model parameter uncertainty compared to the inexact model-based optimal controller (35).

The following two Lemmas substantiate the aforementioned observations for scalar LQR problems.

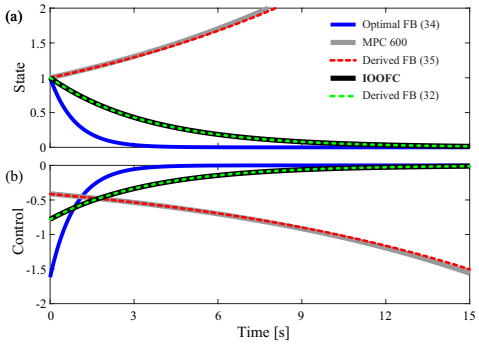


Fig. 1: (a) State. (b) Control. The model parameters are given by: $q = 1$, $a = 0.5$, $\hat{a} = -1$, $b = \hat{b} = 1$ and $T = 15$. The model predictive controller uses a $\Delta t = 0.01$ time step and a $T_{\text{MPC}} = 6$ ($n_t = 600$) time horizon. With this setting, the MPC 600 solution (gray lines) is nearly identical to the offline computed inexact model-based optimal controller $\hat{u}(x)$ (35) (red lines). Algorithm 1 recovers $u^*(x)$ given by (32) (black lines). The optimal controller $u^{\text{opt}}(x)$ (34) that uses the exact but practically unattainable model is shown for reference (blue line).

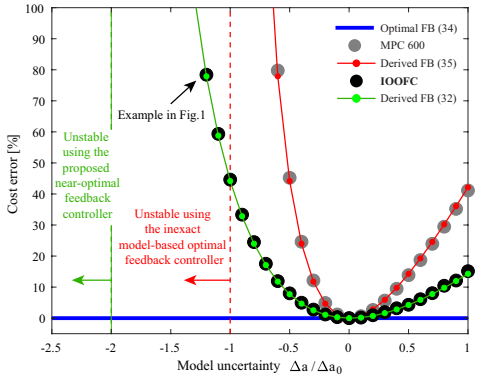


Fig. 2: Normalized cost error $(\mathcal{I}-\mathcal{I}_{\text{opt}})/\mathcal{I}_{\text{opt}} \times 100\%$ as a function of the model parameter uncertainty where $\Delta a = \hat{a} - a$ and $\Delta a_0 = \frac{1}{2}(a + \frac{q}{a})$. The model parameters are given by: $q = 1$, $a = 0.5$, $b = \hat{b} = 1$, and $T = 15$. The model predictive controller uses a $\Delta t = 0.01$ time step and a $T_{\text{MPC}} = 6$ ($n_t = 600$) time horizon. With this setting, the MPC solution is nearly identical to the offline computed inexact model-based optimal controller $\hat{u}(x)$ (35). On the other hand, Algorithm 1 recovers $u^*(x)$ in (32).

Lemma 1 (Improved robustness)**.** *The online computable controller (32) is more robust to model parameter uncertainty than the inexact model-based optimal controller (35), when both controllers are computed using the same inexact model (31) and are applied to the system (30):*

$$(\hat{a}, \hat{b}) \in \mathbb{A}_{\hat{\lambda}=a-b\hat{k}<0} \subset \mathbb{A}_{\lambda^*=a-bk^*<0} \subset \mathbb{R}^2 \tag{36}$$

where $\mathbb{A}_{\lambda<0}$ *defines the set of model parameters which lead to negative closed-loop eigenvalue* $\lambda < 0$.

Proof. For a system (30) with stable open-loop dynamics $a < 0, b > 0, q \geq 0$, the closed-loop stability is guaranteed by both controllers (32), (35) for any $\hat{a} \in \mathbb{R}$ and $\hat{b} > 0$.

For a system (30) with zero open-loop dynamics $a =$

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

$0, b > 0, q > 0$, the closed-loop stability is guaranteed by both controllers (32), (35) for any $\hat{a} \in \mathbb{R}$ and $\hat{b} > 0$.

For a system (30) with unstable open-loop dynamics $a > 0, b > 0, q > 0$, the closed-loop stability of the system controlled with (35) is guaranteed under the following conditions

$$\mathbb{A}_{\hat{\lambda}<0} = \Big\{ \hat{a} \in \mathbb{R}, \hat{b} > 0 : \quad (37)$$
$$\begin{array}{ll} \hat{a} > 0, & (a^2 \leq b^2 q) \vee \left(a^2 > b^2 q, \frac{\hat{a}}{\hat{b}} > \frac{a}{2b} - \frac{bq}{2a}\right) \\ \hat{a} \leq 0, & \left(a^2 < b^2 q, \frac{\hat{a}}{\hat{b}} > \frac{a}{2b} - \frac{bq}{2a}\right) \end{array} \Big\}$$

while the closed-loop stability of the system controlled with (32) is guaranteed by

$$\mathbb{A}_{\lambda^*<0} = \left\{ \hat{a} \in \mathbb{R}, \hat{b} > 0 : \hat{a} > 0 \vee \hat{a} \leq 0, \frac{\hat{a}}{\hat{b}} > -\frac{bq}{a} \right\}. \quad (38)$$

Based on (37) and (38), (36) holds if

$$\frac{a}{2b} - \frac{bq}{2a} > -\frac{bq}{a} \Leftrightarrow a^2 + b^2 q > 0.$$

This completes the proof. ■

Lemma 2 (Reduced cost)**.** *Within the set of inexact models that lead to finite cost $(\hat{a}, \hat{b}) \in \mathbb{A}_{\hat{\lambda}<0}$, for any inexact open-loop dynamics $\hat{a}$ there exists a subset of inexact control dynamics $|b - \hat{b}| < \epsilon$ such that the proposed feedback controller (32) has lower cost compared to the model-based feedback controller (35):*

$$\begin{aligned} &\exists \epsilon > 0 \text{ such that } \forall (\hat{a}, |\hat{b} - b| < \epsilon) \in \mathbb{A}_{\hat{\lambda}<0} : \\ &\qquad \mathcal{I}[u^{opt}(\cdot)] \leq \mathcal{I}[u^*(\cdot)] \leq \mathcal{I}[\hat{u}(\cdot)]. \end{aligned} \quad (39)$$

The equalities hold with $\hat{a} = a$ and $\hat{b} = b$.

Proof. According to (32)–(35), for all $a, b > 0, q \geq 0$ and $(\hat{a}, \hat{b} = b) \wedge (\hat{a} = a, \hat{b}) \in \mathbb{A}_{\hat{\lambda}<0}$, the following relations hold

$$\begin{aligned} &(\hat{a} < a, \hat{b} = b) \vee (\hat{a} = a, \hat{b} > b) : \ \hat{k} < k^* < k_{\text{opt}}, \\ &(\hat{a} > a, \hat{b} = b) \vee (\hat{a} = a, \hat{b} < b) : \ k_{\text{opt}} < k^* < \hat{k}. \end{aligned} \quad (40)$$

Due to the continuity of k in (33) with respect to $\hat{b}$,

$$\exists \epsilon > 0 \text{ such that } \forall (\hat{a}, |\hat{b} - b| < \epsilon) \in \mathbb{A}_{\hat{\lambda}<0}, \ (40) \text{ holds.} \quad (41)$$

Also, for any controller $u(x) = -kx$ that assures closed-loop stability $bk - a > 0$, the cost (29) is given by:

$$\forall (\hat{a}, \hat{b}) \in \mathbb{A}_{\hat{\lambda}<0} : \mathcal{I}[u(x)] = \frac{x_0^2}{4} \frac{k^2 + q}{bk - a} < \infty. \quad (42)$$

This cost is strictly convex with respect to k and has a unique global minimum for $k = k_{\text{opt}}$. Consequently, (40)–(42) imply (39) for the inexact parameters in (41). ■

Remark 6. *There are partially model-free RL algorithms that do not require the open-loop dynamics of the system [62], and model-free RL algorithms that work without any information about the system model [64]. Both of these algorithms are developed for unconstrained linear quadratic optimal control problems, and can be used to recover the exact optimal solution of* (29)-(30) *shown in Figs. 1 and 2 (blue lines). This is possible because for unconstrained linear quadratic optimal control problems the exact analytical form of the value function (quadratic with respect to the state) and the exact analytical form of the optimal controller (linear with respect to the state) are known, and this information is used in [62], [64] to replace the system model. The same is achievable if one can learn the exact system model. Extending this idea to high-dimensional, multiple-input, control constrained and nonlinear optimal control problems remains challenging.*

The simple case study problem considered in this section was analytically solvable, and had a purpose to exemplify the benefits of Algorithm 1. Although the analytical derivation presented in this section is not afforded for more complex problems, Algorithm 1 may still be advantageous, as it provides an online computed near-optimal feedback controller that does not rely on inexact model-based future prediction. In what follows, we support this point with two general numerical examples. The formal extension of Lemma 1 and 2 to more complex systems remains the topic of future research.

B. Large-scale constrained LQR example

We consider a finite time horizon control constrained linear quadratic regulator

$$\min_{\mathbf{u}(\cdot) \in \mathcal{U}} \mathcal{I}[\mathbf{u}(\cdot)] = \frac{1}{2} \int_0^T (\mathbf{x}^\top \mathbf{Q} \mathbf{x} + \mathbf{u}^\top \mathbf{R} \mathbf{u}) dt \quad (43)$$

$$\text{subject to : } \ \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \ \text{ and } \ \mathbf{x}(0) = \mathbf{x}_0 \quad (44)$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, $\mathbf{Q} \in \mathbb{R}^{n \times n}$, $\mathbf{R} \in \mathbb{R}^{m \times m}$, $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$, and

$$\forall t \in \mathbb{T} : \mathbf{u}(t) \in \mathbb{U} = \{ \mathbf{u} \in \mathbb{R}^m : -\mathbf{u}_{\max} \preceq \mathbf{u} \preceq \mathbf{u}_{\max} \}. \quad (45)$$

We make the following assumption:

Assumption 6 (LQR with inexact model)**.** *(i) $\mathbf{Q}$ is positive semi-definite, $\mathbf{R}$ is positive definite.*

(ii) The system is unknown and open-loop unstable

$$\max \Re\lambda(\mathbf{A}) > 0.$$

(iii) The inexact model of the system is given by

$$\dot{\mathbf{x}} = \hat{\mathbf{A}}_\epsilon \mathbf{x} + \hat{\mathbf{B}}_\epsilon \mathbf{u}$$

where $\hat{\mathbf{A}}_\epsilon = \mathbf{A} - \epsilon \mathbf{I} \in \mathbb{R}^{n \times n}$, $\hat{\mathbf{B}}_\epsilon = (1 - \epsilon)\mathbf{B} \in \mathbb{R}^{n \times m}$ and $\epsilon \in [0, 1)$.

(iv) For $\forall \epsilon \in [0, 1)$: $(\hat{\mathbf{A}}_\epsilon, \hat{\mathbf{B}}_\epsilon)$ are stabilizable, and $(\hat{\mathbf{A}}_\epsilon, \mathbf{Q}^{\frac{1}{2}})$ are detectable.

Assumptions 6 (i,iv) ensure that there exist nonzero initial conditions for which the system can be stabilized with control inputs that satisfy (45), see [65]. Assumptions

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

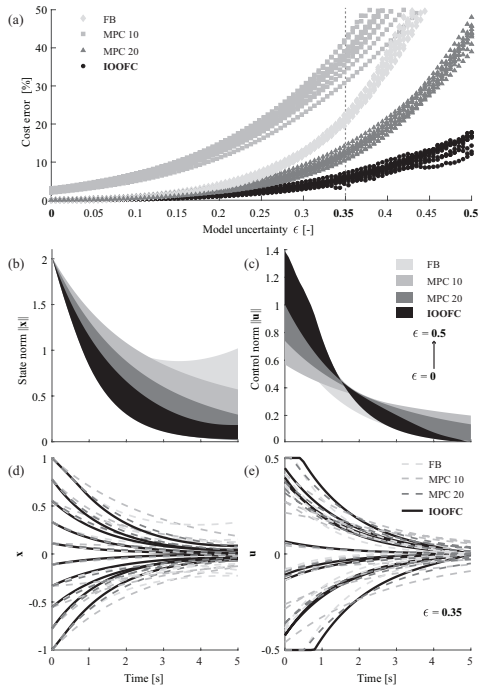


Fig. 3: (a) Normalized cost error $= (\mathcal{I} - \mathcal{I}_{\text{opt}})/\mathcal{I}_{\text{opt}} \times 100\%$ for the online optimization (black circles), the MPC method (gray squares $n_t = 10$ and dark gray triangles $n_t = 20$) and the model-based optimal feedback controller (light gray diamonds). (b,c) Norm of the state trajectories and control functions. (d,e) State and control for $\epsilon = 0.35$. The cLQR problem has $n = 10$ states and $m = 10$ control inputs. The eigenvalues of the open-loop system are $\lambda(\mathbf{A}) = \{-\frac{1}{4} + \frac{i-1}{n-1}\frac{1}{2}\}_{i\in\{1,\dots,n\}} \in [-\frac{1}{4}, \frac{1}{4}]$, the control matrix is $\mathbf{B} = \mathbf{I}_{10}$, the control constraints are $\mathbb{U} = [-\frac{1}{2}, \frac{1}{2}]^{10}$, the time step is $\Delta t = 0.1$, the time horizon is $T = 5$, the initial conditions are $\mathbf{x}_0 = \{-1+2\frac{i-1}{n-1}\}_{i\in\{1,\dots,n\}} \in [-1, 1]$ while the weights in the running cost are $\mathbf{Q} = \mathbf{I}_{10}$ and $\mathbf{R} = \mathbf{I}_{10}$. We solve this cLQR optimization problem for fifty uniformly distributed grid points in $\epsilon \in [0, \frac{1}{2}]$.

6 (ii,iii) make the open-loop unstable system (44) more difficult to stabilize as ϵ becomes larger.

Figure 3 shows the cost when the optimal control problem (43) is solved using Algorithm 1 (black), the MPC method [63] with a shorter prediction horizon (gray) and a longer prediction horizon (dark gray), and the model-based optimal feedback controller [46], [48] (light gray).

According to Fig. 3, Algorithm 1 provides lower cost compared to all three alternative methods in this example. Furthermore, Algorithm 1 provides slower online computation compared to the pre-computed optimal feedback controller (light gray lines), but faster online computation compared to the MPC methods (gray and dark gray lines). The following remark substantiates the latter assertion independent of the considered problem, and the details of the implementation used for the numerical calculation.

Remark 7 (Computational complexity)**.** *The worst-case online computational complexity of the MPC method [63] is $O(n_t m^3)$; it linearly scales with the time horizon used to calculate the control inputs $1 \leq n_t \leq n_T$. This linear scaling is what can be achieved with most efficient nonlinear MPC methods [5] (Section 2.6.1). Shorter time horizon (smaller n_t) leads to faster computation, but it also leads to larger cost (see Fig. 3 gray and dark gray) and reduced robustness to parametric model uncertainty. For example, the shortest time horizon $n_t = 1$ does not render the closed-loop response stable in this example. The worst-case online computational complexity of Algorithm 1 is $O(m^3)$ (10) which corresponds to $n_t = 1$ when compared to MPC methods. The online computational cost comes from the point-wise minimization of the Hamiltonian asserted by the Maximum Principle [2]. Consequently, Algorithm 1 does not trade off optimality to reduce the online computational cost.*

Remark 8 (Alternative RL methods)**.** *The RL methods [62], [64] recalled in Remark 6 are not applicable to* (43)-(44)*, as they are developed for unconstrained problems. Alternative partially model-free RL formulations [23], [66], or completely model-free formulations [67], [68] are developed to solve infinite horizon optimal stabilization problems where the feedback control law is time-invariant. In finite horizon optimal control, the control law is time-varying, even if the dynamics and the cost are time-invariant* (43)-(44)*. There are number of recently developed RL methods to solve finite time horizon optimal control problems that do not treat constraints [29], [32], [69], or use a non-quadratic control cost [30], [31], [33], [70] to relax the hard constraints* (45) *into soft constraints. As noted in [23], the constrained LQR problem* (43)–(45) *does not fit into the setting of these methods. Nevertheless, model-free finite horizon RL methods may be used to approximately solve* (43)–(45)*, and could, in principle, reduce the model bias compared to our proposed method, provided they can learn the time and state dependent relation between the control policy, value function, or the co-state. However, learning these functions for finite time horizon, nonlinear, input constrained, and high dimensional optimal control problems (the cLQR problem here has $n = 10$ states and $m = 10$ control inputs) requires complex function approximators, a large amount of data, and a means to ensure persistent excitation [34].*

Remark 9 (Alternative ILC methods)**.** *Iterative learning control methods [15], [16] are developed to solve tracking problems, where the desired trajectory at convergence is known a'priory, while our proposed method is developed to solve optimal control problems (1), with no notion of a desired trajectory. Some ILC methods formulate the tracking problem as an iterative optimization [18]–[20] or optimal control problem [71], [72], and consequently, these methods may appear similar to our proposed method when applied to an LQR problem with zero desired trajectory. This is mainly because, ILC methods minimize an LQR-type quadratic cost; the tracking error within each*

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

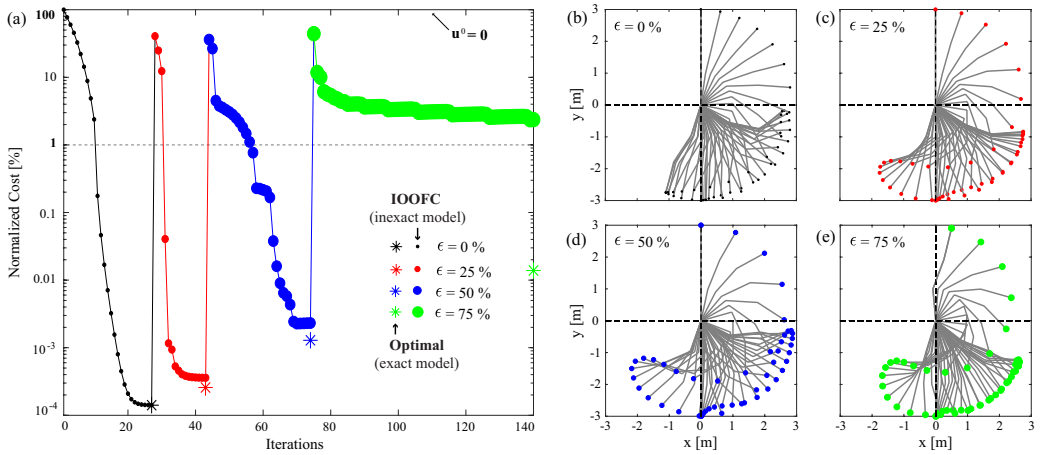


Fig. 4: (a) Normalized cost for the exact dynamics (black) and three different inexact dynamics (red, blue, and green). (b-e) Frame sequence of the motion. The robot is composed of three links with point masses at the end of the links. The physical parameters of the robot are: link lengths $l_1 = l_2 = l_3 = 1$ m; masses $m_1 = m_2 = 1$ kg and $m_3 + \Delta m$ where $m_3 = 2$ kg and $\Delta m \in \{0, 1, 2, 3\}$ kg (the mass of the system is $m_0 = 4$ kg while the masses used in the inexact models are $m_\epsilon \in [4, 5, 6, 7]$ kg); stiffness and damping parameters $\mathbf{K} = \text{diag}(12, 6, 3)$ Nm/rad, $\mathbf{B} = \text{diag}(0.5, 0.5, 0.5)$ Nms/rad; control constraints $\mathbf{u}_{\max} = \pi \times [1, 1, 1]^\top$; bandwidth of the closed-loop motor dynamics $\beta = 5$. The optimization problem has the following parameters: terminal time $T = 4$ s; terminal state $\mathbf{x}_d = [\frac{\pi}{2}, \frac{\pi}{2}, \frac{\pi}{2}, 0, 0, 0, 0, 0, 0, 0, 0, 0]$; terminal weight $\mathbf{S} = \text{diag}(1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0)$; weight of the running cost $w = 2 \times 10^{-6}$. The initial state is: $\mathbf{x}_0 = [-\frac{\pi}{2}, -\frac{\pi}{2}, -\frac{\pi}{2}, 0, 0, 0, -\frac{\pi}{2}, -\frac{\pi}{2}, -\frac{\pi}{2}, 0, 0, 0]$. The optimization problem is solved online. The control input is initialized with zero $\mathbf{u}^0(\cdot) = \mathbf{0}$.

iteration and the change of the control inputs between iterations. However, by iteratively minimizing such LQR-type cost, ILC methods actually solves a singular optimal control problem where only the tracking error is minimized at convergence [71]. This singular optimal control problem formulation is limited compared to our proposed formulation (1) even in the simplest linear quadratic setting; for example, it does not recover the solution of the regular LQR problem (43) unless the desired trajectory coincides with the optimal trajectory of the LQR problem.

C. Unknown and Changing Dynamics

In this section, we consider a challenging optimal control problem where the unknown dynamics of the system changes. One way to solve such problem is to learn the changing model of the system. Such approach is limited when the dynamics significantly changes on a short timescale. Here, we consider a scenario where the unknown dynamics abruptly changes at a time scale short to learn the new model but long enough for the online iterative optimal control to converge. We show that Algorithm 1 is robust enough to deal with such scenario; it finds the improved control under large model error.

The system we consider is a three-link robot driven by three visco-elastic actuators. The model of the robot and the actuators is given by [17]

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \dot{\mathbf{q}} \\ -\mathbf{M}(\mathbf{q})^{-1}[\mathbf{F}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{F}_{\text{A}}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{q}_m)] \\ \dot{\mathbf{q}}_m \\ -2\beta\dot{\mathbf{q}}_m - \beta^2\mathbf{q}_m + \beta^2\mathbf{u} \end{bmatrix} \tag{46}$$

where $\mathbf{x} = [\mathbf{q}, \dot{\mathbf{q}}, \mathbf{q}_m, \dot{\mathbf{q}}_m]^\top \in \mathbb{R}^{12}$ are the states, $\mathbf{q} = [q_1, q_2, q_3]^\top$ are the link angles, $\mathbf{q}_m = [q_{m1}, q_{m2}, q_{m3}]^\top$ are the motor positions, $\mathbf{M} \in \mathbb{R}^{3\times3}$ is a positive definite mass matrix, $\mathbf{F} \in \mathbb{R}^3$ denotes the inertial and gravitational forces, $\mathbf{F}_{\text{A}} = -\mathbf{K}(\mathbf{q} - \mathbf{q}_m) - \mathbf{B}\dot{\mathbf{q}}$ are the forces produced by the actuators, while $\mathbf{K} \in \mathbb{R}^{3\times3}$ and $\mathbf{B} \in \mathbb{R}^{3\times3}$ are positive definite matrices. The motors are driven by range limited control inputs

$$\forall t \in \mathbb{T} : \mathbf{u}(t) \in \mathbb{U} = \{\mathbf{u} \in \mathbb{R}^m : -\mathbf{u}_{\max} \preceq \mathbf{u} \preceq \mathbf{u}_{\max}\}.$$

The closed-loop dynamics of the motors is defined by $\beta \in (0, \infty)$ where larger β means faster change in $\mathbf{q}_m$.

We assume that the inexact model of the system (46) is known, and we consider a problem where the dynamics of the system changes. The change of the exact dynamics is achieved by attaching a mass to the end of the robot, see Fig. 4(b-e). The added mass makes the total mass of the robot change 25% three times

$$\epsilon = \frac{m_\epsilon - m_0}{m_0} \times 100\% \in \{0, 25, 50, 75\}\%$$

where m_0 is the total mass of the system while m_ϵ is the total mass used in the inexact model of the system.

The optimal control problem is defined by

$$\min_{\mathbf{u}(\cdot)\in\mathcal{U}} (\mathbf{x}(T) - \mathbf{x}_d)^\top \mathbf{S}(\mathbf{x}(T) - \mathbf{x}_d) + w \int_0^T \mathbf{u}(t)^\top \mathbf{u}(t) dt$$
$$\text{subject to: } \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \text{ and } \mathbf{x}(0) = \mathbf{x}_0 \tag{47}$$

where $\mathbf{x}_d \in \mathbb{R}^{12}$ is the desired terminal state, $\mathbf{S} \in \mathbb{R}^{12\times12}$ is the positive semi-definite terminal weight, while $w \in \mathbb{R}_+$ is the weight of the running cost.

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

Figure 4 summarizes the results obtained by Algorithm 1. When the model is exact $\epsilon = 0\%$, the cost (black dots) converges to the optimal cost (black star) within less than 30 iterations, see Figs. 4a,b. The rapid convergence is achieved with zero control at initialization.

When the model error is moderate $\epsilon \in \{25, 50\}\%$, the cost decreases and terminates in approximately 40 iterations, see Fig. 4a (red and blue dots). The frame sequences show that the robot reach the target position, see Fig. 4c,d due to the minimization of the terminal cost in (47). The final solution in these cases differs from the exact optimal solution, see Fig. 4a (red and blue stars).

Finally, when the model error is large $\epsilon = 75\%$, the cost decreases slowly, and the solution obtained after 60 iterations does not make the robot reach the target position, see Fig. 4a,d (green dots). In this case, the terminal solution significantly differs from the exact optimal solution, see Fig. 4a (green star) and Fig. 4b,e.

This example shows that Algorithm 1 can be used to robustly implement a near optimal feedback controller online, given imperfect model information for systems with both unknown and changing dynamics.

Remark 10. *Assuming exact dynamics, efficient MPC methods may be used to approximately solve* (47) *[37], [73], but the online computational cost of these methods cannot be reduce down to point-wise minimization of the Hamiltonian (see Remark 3), unless the optimal value function is known. The optimal control problem* (47) *does not fit into ILC formulations that solve tracking problems [15], [16] or assume linear [18]–[20] and exact dynamics [22]. Model-free RL algorithms mentioned in Remark 9 that could solve finite horizon constrained nonlinear optimal control problems use neural networks to approximate the control policy, value function, and the co-state [30]–[32], [70]. However, to reasonably well approximate the control policy, value function, and the co-state for input constrained, nonlinear, high-dimensional, and open-loop unstable systems, for example* (46)*, neural network approximation requires large number of basis functions and large amount of training data [29], [32], [34], [67]. These factors limit the applicability of neural network based methods in tasks where the dynamics changes in a short timescale, see Fig. 4(a).*

VI. Conclusion

In this paper we presented an iterative online feedback control algorithm to efficiently solve finite-time horizon control constrained nonlinear optimal control problems. It is proved that the proposed algorithm converges to a locally optimal feedback controller when the model is exact (Section IV Theorems 1 and 2), and that the controller satisfies the local necessary conditions of optimality along the measured trajectory of the system when the model is inexact (Section IV Theorem 3).

VII. Appendix

Here we present the propositions and proofs that support the material in Section IV.

A. Propositions

Proposition 4 (Bounded derivatives)**.** *There exist constants $c_{\mathbf{f}}, c_{\mathcal{L}} \in (0, \infty)$ such that*

$$\begin{aligned}&\forall(\mathbf{x}, \mathbf{u}, t) \in \mathbb{X} \times \mathbb{U} \times \mathbb{T} :\\ &c_{\mathbf{f}} = \max\{\|\mathbf{f}_{\mathbf{x}}\|, \|\mathbf{f}_{\mathbf{u}}\|\},\\ &c_{\mathcal{L}} = \max\{\|\mathcal{L}_{\mathbf{x}}\|, \|\mathcal{L}_{\mathbf{u}}\|, \|\mathcal{L}_{\mathbf{xx}}\|, \|\mathcal{L}_{\mathbf{xu}}\|\}.\end{aligned}$$

Proof. The proposition follows from the boundedness theorem [74] and Assumptions 1(ii) and 2(i); continuous functions on a compact set $\mathbb{X} \times \mathbb{U} \times \mathbb{T}$ are bounded. ■

Proposition 5 (Bounded inexact derivatives)**.** *Based on Proposition 4 and Assumption 3, the derivatives of the inexact dynamics are bounded.*

Proof. Using the triangle inequality, we obtain

$$\begin{aligned}\|\hat{\mathbf{f}}_{\mathbf{x}}\| &\leq \|\mathbf{f}_{\mathbf{x}}\| + \|\hat{\mathbf{f}}_{\mathbf{x}} - \mathbf{f}_{\mathbf{x}}\| \leq c_{\mathbf{f}} + \epsilon_{\mathbf{x}},\\ \|\hat{\mathbf{f}}_{\mathbf{u}}\| &\leq \|\mathbf{f}_{\mathbf{u}}\| + \|\hat{\mathbf{f}}_{\mathbf{u}} - \mathbf{f}_{\mathbf{u}}\| \leq c_{\mathbf{f}} + \epsilon_{\mathbf{u}}\end{aligned}$$

where $c_{\mathbf{f}}, \epsilon_{\mathbf{x}}, \epsilon_{\mathbf{u}} \in [0, \infty)$. ■

Proposition 6 (Local Lipschitz continuity)**.** *$\forall \mathbf{x}, \mathbf{x} + \delta\mathbf{x} \in \mathbb{X}$ and $\forall \mathbf{u}, \mathbf{u} + \delta\mathbf{u} \in \mathbb{U}$ there exist constants $c_1, c_2, c_3, c_4, c_5 \in (0, \infty)$ independent of $t \in \mathbb{T}$, such that*

$$\begin{aligned}&\|\mathbf{f}(\mathbf{x} + \delta\mathbf{x}, \mathbf{u} + \delta\mathbf{u}, t) - \mathbf{f}(\mathbf{x}, \mathbf{u}, t)\| \leq c_1(\|\delta\mathbf{x}\| + \|\delta\mathbf{u}\|),\\ &\|\mathbf{f}_{\mathbf{x}}(\mathbf{x} + \delta\mathbf{x}, \mathbf{u} + \delta\mathbf{u}, t) - \mathbf{f}_{\mathbf{x}}(\mathbf{x}, \mathbf{u}, t)\| \leq c_2(\|\delta\mathbf{x}\| + \|\delta\mathbf{u}\|),\\ &\|\mathbf{f}_{\mathbf{u}}(\mathbf{x} + \delta\mathbf{x}, \mathbf{u} + \delta\mathbf{u}, t) - \mathbf{f}_{\mathbf{x}}(\mathbf{x}, \mathbf{u}, t)\| \leq c_3(\|\delta\mathbf{x}\| + \|\delta\mathbf{u}\|),\\ &\|\mathcal{L}_{\mathbf{x}}(\mathbf{x} + \delta\mathbf{x}, \mathbf{u} + \delta\mathbf{u}, t) - \mathcal{L}_{\mathbf{x}}(\mathbf{x}, \mathbf{u}, t)\| \leq c_4(\|\delta\mathbf{x}\| + \|\delta\mathbf{u}\|),\\ &\|\mathcal{L}_{\mathbf{u}}(\mathbf{x} + \delta\mathbf{x}, \mathbf{u} + \delta\mathbf{u}, t) - \mathcal{L}_{\mathbf{u}}(\mathbf{x}, \mathbf{u}, t)\| \leq c_5(\|\delta\mathbf{x}\| + \|\delta\mathbf{u}\|).\end{aligned}$$

Proof. The proof follows from Assumptions 1(ii) and 2(i). ■

Proposition 7 (Boundedness of $\mathbf{R}$ and $\mathbf{r}$)**.** *The coefficients $\mathbf{R}$ and $\mathbf{r}$ in the cQP* (6)*, computed using (4) and (5), are bounded.*

For any admissible control function $\forall t \in \mathbb{T} : \mathbf{u}(t) \in \mathbb{U}$ and corresponding state function $\forall t \in \mathbb{T} : \mathbf{x}(t) \in \mathbb{X}$; there exist constants $c_{\mathbf{R}}, c_{\mathbf{r}} \in (0, \infty)$ such that

$$\forall t \in \mathbb{T} : \ \|\mathbf{R}(t)\| \leq c_{\mathbf{R}}, \ \|\mathbf{r}(t)\| \leq c_{\mathbf{r}}.$$

Proof. First, we define $\tau = T - t$ and

$$R(\tau) = \|\mathbf{R}(T - \tau)\| = \|\mathbf{R}(t)\|.$$

Second, we derive

$$\frac{d}{d\tau}R(\tau) = -\frac{d}{dt}\|\mathbf{R}(t)\| \leq \|\dot{\mathbf{R}}(t)\| \leq 2\|\hat{\mathbf{f}}_{\mathbf{x}}\|R(\tau) + \|\mathcal{L}_{\mathbf{xx}}\|.$$

Third, using Proposition 4 and 5, we transform the above

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

inequality into

$$\frac{d}{d\tau}R(\tau) \leq 2(c_{\mathbf{f}}+\epsilon_{\mathbf{x}})R(\tau)+c_{\mathcal{L}}, \quad R(0)=0.$$

Finally, using Gronwall's inequality, we obtain

$$\|\mathbf{R}(t)\| = R(\tau) \leq \frac{c_{\mathcal{L}}}{2(c_{\mathbf{f}}+\epsilon_{\mathbf{x}})}(e^{2(c_{\mathbf{f}}+\epsilon_{\mathbf{x}})T}-1) = c_{\mathbf{R}}. \tag{48}$$

Similarly, we derive

$$\|\mathbf{r}(t)\| \leq \frac{c_{\mathcal{L}}}{c_{\mathbf{f}}+\epsilon_{\mathbf{x}}}(e^{(c_{\mathbf{f}}+\epsilon_{\mathbf{x}})T}-1) = c_{\mathbf{r}}. \tag{49}$$

In summary, $c_{\mathbf{R}}, c_{\mathbf{r}} \in (0,\infty)$ are constants expressible as functions of $c_{\mathbf{f}}$, $c_{\mathcal{L}}$, $\epsilon_{\mathbf{x}}$ and T. ■

Proposition 8 (Boundedness of the state and control variations)**.** *The following inequality relations hold between the state and control variations*

$$\|\delta\mathbf{x}^i\|_{\mathcal{L}^1} \leq e^{c_1 T}\|\delta\mathbf{u}^i\|_{\mathcal{L}^1}, \tag{50}$$

$$\|\delta\mathbf{x}^i\|^2_{\mathcal{L}^2} \leq \frac{1}{2}c_1^2T^2e^{2c_1T}\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2}, \tag{51}$$

$$\int_0^T \|\delta\mathbf{x}^i(t)\|\|\delta\mathbf{u}^i(t)\|dt \leq \frac{\sqrt{2}}{2}c_1Te^{c_1T}\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2}. \tag{52}$$

Proof. (i) Based on Proposition 6, we obtain

$$\frac{d}{dt}\|\delta\mathbf{x}^i(t)\| \leq \|\delta\dot{\mathbf{x}}^i(t)\| \leq c_1(\|\delta\mathbf{x}^i(t)\|+\|\delta\mathbf{u}^i(t)\|). \tag{53}$$

Application of Gronwall's inequality to (53) leads to

$$\|\delta\mathbf{x}^i(t)\| \leq c_1e^{c_1t}\|\delta\mathbf{u}^i\|_{\mathcal{L}^1}. \tag{54}$$

By integrating (54) we further obtain

$$\|\delta\mathbf{x}^i\|_{\mathcal{L}^1} \leq \|\delta\mathbf{u}^i\|_{\mathcal{L}^1}\int_0^T c_1e^{c_1t}dt \leq e^{c_1T}\|\delta\mathbf{u}^i\|_{\mathcal{L}^1}.$$

This relation proves (50).

(ii) Using Schwarz's inequality, (54) leads to

$$\|\delta\mathbf{x}^i(t)\|^2 \leq c_1^2e^{2c_1t}\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^1} \leq tc_1^2e^{2c_1T}\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2}. \tag{55}$$

By integrating (55) we obtain

$$\|\delta\mathbf{x}^i\|^2_{\mathcal{L}^2} \leq c_1^2e^{2c_1T}\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2}\int_0^T tdt \leq \frac{c_1^2T^2e^{2c_1T}}{2}\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2}. \tag{56}$$

This relation proves (51).

(iii) Finally, we use Schwarz's inequality and (56) to obtain

$$\begin{aligned}\left(\int_0^T \|\delta\mathbf{x}^i(t)\|\|\delta\mathbf{u}^i(t)\|dt\right)^2 &\leq \|\delta\mathbf{x}^i\|^2_{\mathcal{L}^2}\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2} \\ &\leq \frac{c_1^2T^2e^{2c_1T}}{2}\|\delta\mathbf{u}^i\|^4_{\mathcal{L}^2}.\end{aligned} \tag{57}$$

The positive square root of (57) is (52). ■

B. Proof of Proposition 1

Proof. We define the Hamiltonian

$$\mathcal{H}(\mathbf{x},\mathbf{u},\mathbf{r},t) = \mathcal{L}(\mathbf{x},\mathbf{u},t)+\mathbf{r}^\top\mathbf{f}(\mathbf{x},\mathbf{u},t) \tag{58}$$

where $\mathbf{f}(\mathbf{x},\mathbf{u},t)$ is the dynamics in (1) while $\mathbf{r}$ is the co-state defined in (5). Using (58), we define the change in the cost between two subsequent iterations

$$\begin{aligned}\Delta\mathcal{I}^i &= \int_0^T [\mathcal{L}(\mathbf{x}^i,\mathbf{u}^i,t)-\mathcal{L}(\mathbf{x}^{i-1},\mathbf{u}^{i-1},t)]dt \\ &= \int_0^T \big[\mathcal{H}(\mathbf{x}^i,\mathbf{u}^i,\mathbf{r}^{i-1},t)-\mathcal{H}(\mathbf{x}^{i-1},\mathbf{u}^i,\mathbf{r}^{i-1},t) \\ &+ \int_0^T \big[\mathcal{H}(\mathbf{x}^{i-1},\mathbf{u}^i,\mathbf{r}^{i-1},t)-\mathcal{H}(\mathbf{x}^{i-1},\mathbf{u}^{i-1},\mathbf{r}^{i-1},t)\big]dt \\ &- \int_0^T (\mathbf{r}^{i-1})^\top\delta\dot{\mathbf{x}}^i dt = \Delta\mathcal{I}^i_1+\Delta\mathcal{I}^i_2+\Delta\mathcal{I}^i_3.\end{aligned} \tag{59}$$

(i) The first integral in (59) can be transformed into

$$\begin{aligned}\Delta\mathcal{I}^i_1 &= \int_0^T\left[\int_0^1 \mathcal{H}_{\mathbf{x}}(\mathbf{x}^{i-1}+\tau\delta\mathbf{x}^i,\mathbf{u}^i,\mathbf{r}^{i-1},t)\delta\mathbf{x}^i d\tau\right]dt \\ &= \int_0^T \mathcal{H}_{\mathbf{x}}(\mathbf{x}^{i-1},\mathbf{u}^{i-1},\mathbf{r}^{i-1},t)\delta\mathbf{x}^i dt \\ &+ \int_0^T\left[\int_0^1 \big[\mathcal{H}_{\mathbf{x}}(\mathbf{x}^{i-1}+\tau\delta\mathbf{x}^i,\mathbf{u}^i,\mathbf{r}^{i-1},t) \right. \\ &\left. - \mathcal{H}_{\mathbf{x}}(\mathbf{x}^{i-1},\mathbf{u}^{i-1},\mathbf{r}^{i-1},t)\big]\delta\mathbf{x}^i d\tau\right]dt.\end{aligned} \tag{60}$$

The last integral in (60) can be upper bounded using (58), Propositions 6 and 7

$$\begin{aligned}\Delta\mathcal{I}^i_1 &\leq \int_0^T \mathcal{H}_{\mathbf{x}}(\mathbf{x}^{i-1},\mathbf{u}^{i-1},\mathbf{r}^{i-1},t)\delta\mathbf{x}^i dt \\ &+ \int_0^T (c_{\mathbf{r}}c_2+c_4)(\frac{1}{2}\|\delta\mathbf{x}^i\|^2+\|\delta\mathbf{x}^i\|\|\delta\mathbf{u}^i\|)dt.\end{aligned} \tag{61}$$

(ii) The second term in (59) can be similarly transformed

$$\begin{aligned}\Delta\mathcal{I}^i_2 &\leq \int_0^T \mathcal{H}_{\mathbf{u}}(\mathbf{x}^{i-1},\mathbf{u}^{i-1},\mathbf{r}^{i-1},t)\delta\mathbf{u}^i dt \\ &+ \int_0^T (c_{\mathbf{r}}c_3+c_5)\frac{1}{2}\|\delta\mathbf{u}^i\|^2 dt.\end{aligned} \tag{62}$$

Also, the solution of the cQP (6) $\delta\mathbf{u}^i$ satisfies the following inequality [59]:

$$\begin{aligned}&\exists r>0, \forall\delta\mathbf{u}+\mathbf{u}^{i-1} \in \mathcal{B}_r(\mathbf{u}^{i-1})\cap\mathbb{U}: \\ &\left.\frac{\partial[\frac{1}{2}\delta\mathbf{u}^\top\mathcal{L}^{\mathrm{R}}_{\mathbf{uu}}\delta\mathbf{u}+\alpha^i(\delta\mathbf{u})^\top\mathbf{b}(\mathbf{x}^i,t)]}{\partial\delta\mathbf{u}}\right|_{\delta\mathbf{u}^i}(\delta\mathbf{u}-\delta\mathbf{u}^i) \\ &=[(\delta\mathbf{u}^i)^\top\mathcal{L}^{\mathrm{R}}_{\mathbf{uu}}+\alpha^i(\delta\mathbf{x}^i)^\top(\mathcal{L}_{\mathbf{xu}}+\mathbf{R}^{i-1}\hat{\mathbf{f}}_{\mathbf{u}}) \\ &+\alpha^i(\mathcal{L}_{\mathbf{u}}+(\mathbf{r}^{i-1})^\top\hat{\mathbf{f}}_{\mathbf{u}})](\delta\mathbf{u}-\delta\mathbf{u}^i)\geq 0.\end{aligned}$$

By setting $\delta\mathbf{u}=\mathbf{0}$, the above relation implies

$$\begin{aligned}&\mathcal{H}_{\mathbf{u}}(\mathbf{x}^{i-1},\mathbf{u}^{i-1},\mathbf{r}^{i-1},t)\delta\mathbf{u}^i = [\mathcal{L}_{\mathbf{u}}+(\mathbf{r}^{i-1})^\top\mathbf{f}_{\mathbf{u}}]\delta\mathbf{u}^i \\ &\leq -\frac{1}{\alpha^i}(\delta\mathbf{u}^i)^\top\mathcal{L}^{\mathrm{R}}_{\mathbf{uu}}\delta\mathbf{u}^i-(\delta\mathbf{x}^i)^\top(\mathcal{L}_{\mathbf{xu}}+\mathbf{R}^{i-1}\hat{\mathbf{f}}_{\mathbf{u}})\delta\mathbf{u}^i \\ &+(\mathbf{r}^{i-1})^\top(\mathbf{f}_{\mathbf{u}}-\hat{\mathbf{f}}_{\mathbf{u}})\delta\mathbf{u}^i.\end{aligned} \tag{63}$$

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

(iii) Finally, we use (58), (5), $\delta\mathbf{x}^i(0)=\mathbf{0}$ and $\mathbf{r}^{i-1}(T)=\mathbf{0}$ to simplify the third term in (59)

$$\begin{aligned}\Delta\mathcal{I}_3^i &= \int_0^T \big[\mathcal{H}_{\mathbf{x}}(\mathbf{x}^{i-1},\mathbf{u}^{i-1},\mathbf{r}^{i-1},t)\delta\mathbf{x}^i-(\mathbf{r}^{i-1})^\top\delta\dot{\mathbf{x}}^i\big]dt\\ &-\int_0^T \mathcal{H}_{\mathbf{x}}(\mathbf{x}^{i-1},\mathbf{u}^{i-1},\mathbf{r}^{i-1},t)\delta\mathbf{x}^i dt\\ &=\int_0^T[\mathcal{L}_{\mathbf{x}}+(\mathbf{r}^{i-1})^\top\mathbf{f}_{\mathbf{x}}+(\dot{\mathbf{r}}^{i-1})^\top]\delta\mathbf{x}^i dt-(\mathbf{r}^{i-1})^\top\delta\mathbf{x}^i\big|_0^T\\ &-\int_0^T \mathcal{H}_{\mathbf{x}}(\mathbf{x}^{i-1},\mathbf{u}^{i-1},\mathbf{r}^{i-1},t)\delta\mathbf{x}^i dt \qquad (64)\\ &=\int_0^T[(\mathbf{r}^{i-1})^\top(\mathbf{f}_{\mathbf{x}}-\hat{\mathbf{f}}_{\mathbf{x}})-\mathcal{H}_{\mathbf{x}}(\mathbf{x}^{i-1},\mathbf{u}^{i-1},\mathbf{r}^{i-1},t)]\delta\mathbf{x}^i dt.\end{aligned}$$

(iv) Substituting (61)–(64) into (59), we obtain

$$\begin{aligned}\Delta\mathcal{I}^i\le\int_0^T\Big[&(\mathbf{r}^{i-1})^\top[(\mathbf{f}_{\mathbf{x}}-\hat{\mathbf{f}}_{\mathbf{x}})\delta\mathbf{x}^i+(\mathbf{f}_{\mathbf{u}}-\hat{\mathbf{f}}_{\mathbf{u}})\delta\mathbf{u}^i]\\ &-\frac{1}{\alpha^i}(\delta\mathbf{u}^i)^\top\mathcal{L}^{\mathrm{R}}_{\mathbf{uu}}\delta\mathbf{u}^i-(\delta\mathbf{x}^i)^\top(\mathcal{L}_{\mathbf{xu}}+\mathbf{R}^{i-1}\hat{\mathbf{f}}_{\mathbf{u}})\delta\mathbf{u}^i\\ &+(c_{\mathbf{r}}c_2+c_4)(\frac{1}{2}\|\delta\mathbf{x}^i\|^2+\|\delta\mathbf{x}^i\|\|\delta\mathbf{u}^i\|) \qquad (65)\\ &+(c_{\mathbf{r}}c_3+c_5)\frac{1}{2}\|\delta\mathbf{u}^i\|^2\Big]dt.\end{aligned}$$

Next, we introduce $c_c=\max\{c_1,c_2,c_3,c_4,c_5\}\in(0,\infty)$, and transform (65) into the following inequality relation

$$\begin{aligned}\Delta\mathcal{I}^i\le\int_0^T\Big[&-\Big(\frac{\lambda^i_{\min}}{\alpha^i}-\frac{1}{2}(c_{\mathbf{r}}+1)c_c\Big)\|\delta\mathbf{u}^i\|^2 \qquad (66)\\ &+\frac{1}{2}(c_{\mathbf{r}}+1)c_c\|\delta\mathbf{x}^i\|^2\\ &+\big(\|\mathcal{L}_{\mathbf{xu}}\|+\|\mathbf{R}^{i-1}\|(\|\mathbf{f}_{\mathbf{u}}\|+\|\hat{\mathbf{f}}_{\mathbf{u}}-\mathbf{f}_{\mathbf{u}}\|)\big)\|\delta\mathbf{x}^i\|\|\delta\mathbf{u}^i\|\\ &+(c_{\mathbf{r}}+1)c_c\|\delta\mathbf{x}^i\|\|\delta\mathbf{u}^i\|\\ &+\|\mathbf{r}^{i-1}\|(\|\mathbf{f}_{\mathbf{x}}-\hat{\mathbf{f}}_{\mathbf{x}}\|\|\delta\mathbf{x}^i\|+\|\mathbf{f}_{\mathbf{u}}-\hat{\mathbf{f}}_{\mathbf{u}}\|\|\delta\mathbf{u}^i\|)\Big]dt.\end{aligned}$$

Finally, we substitute (48)–(52) into (66), and obtain

$$\Delta\mathcal{I}^i\le-\Big(\frac{\lambda^i_{\min}}{\alpha^i}-c_{\mathbf{u}}\Big)\|\delta\mathbf{u}^i\|^2_{\mathcal{L}^2}+c_\epsilon\|\delta\mathbf{u}^i\|_{\mathcal{L}^1} \qquad (67)$$

where

$$\begin{aligned}c_{\mathbf{u}}=&\frac{\sqrt{2}}{2}[c_{\mathcal{L}}+c_{\mathbf{R}}(c_{\mathbf{f}}+\epsilon_{\mathbf{u}})]c_cTe^{c_cT}\\ &+\frac{1}{2}\Big(1+\frac{\sqrt{2}}{2}c_cTe^{c_cT}\Big)^2(c_{\mathbf{r}}+1)c_c, \qquad (68)\\ c_\epsilon=&c_{\mathbf{r}}(e^{c_cT}\epsilon_{\mathbf{x}}+\epsilon_{\mathbf{u}})\end{aligned}$$

are constants independent of $i\in\mathbb{N}$. ■

C. Proof of Proposition 2

Proof. Substituting $c_{\mathbf{R}}$ (48) and $c_{\mathbf{r}}$ (49) into (68), we obtain the following relations

$$\begin{aligned}c_{\mathbf{u}}=&\frac{\sqrt{2}}{2}c_{\mathcal{L}}c_cTe^{c_cT}+\frac{1}{2}\Big(1+\frac{\sqrt{2}}{2}c_cTe^{c_cT}\Big)^2c_c\\ &+\frac{\sqrt{2}}{4}c_{\mathcal{L}}\frac{c_{\mathbf{f}}+\epsilon_{\mathbf{u}}}{c_{\mathbf{f}}+\epsilon_{\mathbf{x}}}(e^{2(c_{\mathbf{f}}+\epsilon_{\mathbf{x}})T}-1)c_cTe^{c_cT}\\ &+\frac{1}{2}\Big(1+\frac{\sqrt{2}}{2}c_cTe^{c_cT}\Big)^2\frac{c_{\mathcal{L}}c_c}{c_{\mathbf{f}}+\epsilon_{\mathbf{x}}}(e^{(c_{\mathbf{f}}+\epsilon_{\mathbf{x}})T}-1),\\ c_\epsilon=&\frac{1}{2}c_{\mathcal{L}}\frac{e^{c_cT}\epsilon_{\mathbf{x}}+\epsilon_{\mathbf{u}}}{c_{\mathbf{f}}+\epsilon_{\mathbf{x}}}(e^{(c_{\mathbf{f}}+\epsilon_{\mathbf{x}})T}-1).\end{aligned}$$

Assuming no model error $\epsilon_{\mathbf{x}}=\epsilon_{\mathbf{u}}=0$, we obtain

$$c_{\mathbf{u}}=c_{\mathbf{u}\min}>0 \text{ and } c_\epsilon=c_{\epsilon\min}=0.$$

Also, for any model error $\epsilon_{\mathbf{x}},\epsilon_{\mathbf{u}}\in[0,\infty)$ and $c_{\mathcal{L}},c_{\mathbf{f}}\in(0,\infty)$, we obtain

$$\frac{\partial c_{\mathbf{u}}}{\partial\epsilon_{\mathbf{x}}}>0,\ \ \frac{\partial c_{\mathbf{u}}}{\partial\epsilon_{\mathbf{u}}}>0,\ \ \frac{\partial c_\epsilon}{\partial\epsilon_{\mathbf{x}}}>0 \text{ and } \frac{\partial c_\epsilon}{\partial\epsilon_{\mathbf{u}}}>0.$$

Therefore, $c_{\mathbf{u}}$ and c_ϵ are monotonically increasing functions of $\epsilon_{\mathbf{x}}$ and $\epsilon_{\mathbf{u}}$. ■

D. Proof of Proposition 3

Proof. First, we find the solution of the cQP (6) as $\alpha\to0^+$. For this purpose, let us consider

$$\delta\mathbf{u}(\mathbf{x}+\delta\mathbf{x},t;\alpha)=\underset{\delta\mathbf{u}\in\delta\mathbb{U}}{\operatorname{argmin}}\ \frac{1}{2}\delta\mathbf{u}^\top\mathcal{L}^{\mathrm{R}}_{\mathbf{uu}}\delta\mathbf{u}+\alpha\delta\mathbf{u}^\top\mathbf{b}(\mathbf{x}+\delta\mathbf{x},t) \qquad (69)$$

where

$$\mathbf{b}(\mathbf{x}+\delta\mathbf{x},t)=(\mathcal{L}_{\mathbf{ux}}+\hat{\mathbf{f}}_{\mathbf{u}}^\top\mathbf{R})\delta\mathbf{x}+\mathcal{L}_{\mathbf{u}}+\hat{\mathbf{f}}_{\mathbf{u}}^\top\mathbf{r}. \qquad (70)$$

We note that $\mathcal{L}^{\mathrm{R}}_{\mathbf{uu}}\succeq\mathbf{I}\lambda^i_{\min}$ and $\forall i:\lambda^i_{\min}>0$ imply

$$\forall t\in\mathbb{T}:\ \lim_{\alpha\to0^+}\delta\mathbf{u}(\mathbf{x},t;\alpha)=\mathbf{0} \qquad (71)$$

while (71) and (54) imply

$$\forall t\in\mathbb{T}:\ \lim_{\alpha\to0^+}\delta\mathbf{x}(t;\alpha)=\mathbf{0}. \qquad (72)$$

Based on (71) and (72), the first term in (70) is negligible compared to the second term

$$\lim_{\alpha\to0^+}\mathbf{b}(\mathbf{x}+\delta\mathbf{x},t)=\mathbf{b}(\mathbf{x},t).$$

This relation implies $\delta\mathbf{u}(\mathbf{x}+\delta\mathbf{x},t;\alpha)=\delta\mathbf{u}(\mathbf{x},t;\alpha)$ for $\alpha\to0^+$. Furthermore, Theorem 5.4.1 in [75] asserts that the solution of a strictly convex parametric quadratic program (69)$|_{\delta\mathbf{x}=\mathbf{0}}$ is continuous, and piecewise affine with respect to $\alpha\in(0,1]$. Therefore, given the positive definiteness of $\mathcal{L}^{\mathrm{R}}_{\mathbf{uu}}$ in (69), we obtain

$$\delta\mathbf{u}(\mathbf{x}+\delta\mathbf{x},t;\alpha)=\delta\mathbf{u}(\mathbf{x},t;\alpha)=\mathbf{c}(t)\alpha \text{ if } \alpha\to0^+. \qquad (73)$$

(i) Relation (73) implies

$$\forall i\in\mathbb{N}:\delta\mathbf{u}^i(t)=\mathbf{c}^i(t)\alpha \ \text{ if } \ \alpha\to0^+.$$

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

Consequently,

$$\forall i \in \mathbb{N} : \|\delta \mathbf{u}^i\|_{\mathcal{L}^1} = \|\mathbf{c}^i\|_{\mathcal{L}^1} \alpha = c^i_{\delta \mathbf{u}} \alpha \text{ if } \alpha \to 0^+. \quad (74)$$

This proves the first part of the proposition.

(ii) In order to prove the second part of the proposition, we assume that

$$\Delta \mathcal{I}^i \neq 0 \quad \text{and} \quad c^i_{\delta \mathbf{u}} = 0. \quad (75)$$

Given $c^i_{\delta \mathbf{u}} = 0$, (74) implies

$$\|\delta \mathbf{u}^i\|_{\mathcal{L}^1} = c^i_{\delta \mathbf{u}} \alpha = 0 \quad \text{if} \quad \alpha \to 0^+. \quad (76)$$

Consequently, (76) and (50) imply

$$\|\delta \mathbf{x}^i\|_{\mathcal{L}^1} = 0 \quad \text{if} \quad \alpha \to 0^+.$$

Becasue the control and state variations are zero in the class of $\mathcal{L}^1$ functions, (8) implies that the variation of the cost is also zero $\Delta \mathcal{I}^i = 0$. This contradicts the assumption $\Delta \mathcal{I}^i \neq 0$ in (75). ■

REFERENCES

[1] R. Bellman, *Dynamic Programming*. Princeton, NJ, USA: Princeton University Press, 1 ed., 1957.

[2] L. Pontryagin, V. Boltyanskii, R. Gamkrelidze, and E. Mishchenko, *The Mathematical Theory of Optimal Processes*. John Wiley and Sons Inc., 1962.

[3] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 2. Athena Scientific, Belmont, MA, USA, 1995.

[4] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[5] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, pp. 2967–2986, 2014.

[6] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.

[7] H. Chen and F. Allgöwer, "A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability," *Automatica*, vol. 34, no. 10, pp. 1205–1217, 1998.

[8] S. Yu, M. Reble, H. Chen, and F. Allgöwer, "Inherent robustness properties of quasi-infinite horizon nonlinear model predictive control," *Automatica*, vol. 50, no. 9, pp. 2269–2280, 2014.

[9] P. O. Scokaert and D. Mayne, "Min-max feedback model predictive control for constrained linear systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 8, pp. 1136–1142, 1998.

[10] D. M. Raimondo, D. Limon, M. Lazar, L. Magni, and E. F. Camacho, "Min-max model predictive control of nonlinear systems: A unifying overview on stability," *European Journal of Control*, vol. 15, no. 1, pp. 5–21, 2009.

[11] W. Langson, I. Chryssochoos, S. V. Raković, and D. Mayne, "Robust model predictive control using tubes," *Automatica*, vol. 40, pp. 125–133, 2004.

[12] D. Q. Mayne, M. M. Seron, and S. Raković, "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41, no. 2, pp. 219–224, 2005.

[13] D. Q. Mayne, E. C. Kerrigan, E. Van Wyk, and P. Falugi, "Tube-based robust nonlinear model predictive control," *International Journal of Robust and Nonlinear Control*, vol. 21, no. 11, pp. 1341–1353, 2011.

[14] S. V. Raković, B. Kouvaritakis, M. Cannon, C. Panos, and R. Findeisen, "Parameterized tube model predictive control," *IEEE Transactions on Automatic Control*, vol. 57, no. 11, pp. 2746–2761, 2012.

[15] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *Journal of Robotic Systems*, vol. 1, no. 2, pp. 123–140, 1984.

[16] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control," *IEEE Control Systems*, vol. 26, no. 3, pp. 96–114, 2006.

[17] D. J. Braun, F. Petit, F. Huber, S. Haddadin, P. Van Der Smagt, A. Albu-Schäffer, and S. Vijayakumar, "Robots driven by compliant actuators: Optimal control under actuation constraints," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1085–1101, 2013.

[18] K. S. Lee, I.-S. Chin, H. J. Lee, and J. H. Lee, "Model predictive control technique combined with iterative learning for batch processes," *AIChE Journal*, vol. 45, no. 10, pp. 2175–2187, 1999.

[19] K. S. Lee and J. H. Lee, "Convergence of constrained model-based predictive control for batch processes," *IEEE Transactions on Automatic Control*, vol. 45, no. 10, pp. 1928–1932, 2000.

[20] J. H. Lee, K. S. Lee, and W. C. Kim, "Model-based iterative learning control with a quadratic criterion for time-varying linear systems," *Automatica*, vol. 36, no. 5, pp. 641–657, 2000.

[21] J. R. Cueli and C. Bordons, "Iterative nonlinear model predictive control. Stability, robustness and applications," *Control Engineering Practice*, vol. 16, no. 9, pp. 1023–1034, 2008.

[22] U. Rosolia and F. Borrelli, "Learning model predictive control for iterative tasks. A data-driven control framework," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 1883–1896, 2018.

[23] F. A. Yaghmaie and D. J. Braun, "Reinforcement learning for a class of continuous-time input constrained optimal control problems," *Automatica*, vol. 99, pp. 221–227, 2019.

[24] M. Abu-Khalaf and F. L. Lewis, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," *Automatica*, vol. 41, no. 5, pp. 779–791, 2005.

[25] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits and Systems Magazine*, vol. 9, pp. 32–50, 2009.

[26] K. G. Vamvoudakis and F. L. Lewis, "Online actor–critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878–888, 2010.

[27] R. Kamalapurkar, P. Walters, and W. E. Dixon, "Model-based reinforcement learning for approximate optimal regulation," *Automatica*, vol. 64, pp. 94–104, 2016.

[28] B. Kiumarsi, K. G. Vamvoudakis, H. Modares, and F. L. Lewis, "Optimal and autonomous control using reinforcement learning: A Survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2042–2062, 2018.

[29] T. Cheng, F. L. Lewis, and M. Abu-Khalaf, "A neural network solution for fixed-final time optimal control of nonlinear systems," *Automatica*, vol. 43, no. 3, pp. 482–490, 2007.

[30] T. Cheng, F. L. Lewis, and M. Abu-Khalaf, "Fixed-final-time-constrained optimal control of nonlinear systems using neural network HJB approach," *IEEE Transactions on Neural Networks*, vol. 18, no. 6, pp. 1725–1737, 2007.

[31] A. Heydari and S. N. Balakrishnan, "Finite-horizon control-constrained nonlinear optimal control using single network adaptive critics," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 1, pp. 145–157, 2013.

[32] Q. Zhao, H. Xu, and S. Jagannathan, "Neural network-based finite-horizon optimal control of uncertain affine nonlinear discrete-time systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 3, pp. 486–499, 2015.

[33] H. Zhang, X. Cui, Y. Luo, and H. Jiang, "Finite-horizon H_∞ tracking control for unknown nonlinear systems with saturating actuators," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, pp. 1200–1212, April 2018.

[34] B. Recht, "A tour of reinforcement learning: The view from continuous control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, no. nn, pp. 00–00, 2019 (accepted).

[35] L. Cesari, *Optimization—Theory and Applications*. Springer New York, 1983.

[36] Y. Sakawa and Y. Shindo, "On global convergence of an algorithm for optimal control," *IEEE Transactions on Automatic Control*, vol. 25, no. 6, pp. 1149–1153, 1980.

[37] A. Jadbabaie and J. Hauser, "On the stability of receding horizon control with a general terminal cost," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 674–678, 2005.

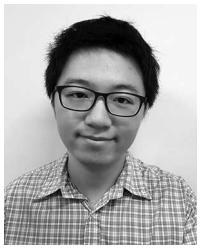
[38] A. F. Filippov, "On certain questions in the theory of optimal control," *J. SIAM Control Ser. A*, vol. 1, no. 1, p. 76–84, 1962.

[39] L. D. Berkovitz, *Optimal Control Theory*. Springer-Verlag, 1974.

DOI: 10.1109/TAC.2020.2986211

PREPRINT : "© 20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works."

[40] S. K. Mitter, "Successive approximation methods for the solution of optimal control problems," *Automatica*, vol. 3, pp. 135–149, 1966.
[41] F. L. Chernousko and A. A. Lyubishin, "Method of sucessive approximation for the soltion of optimal control problems," *Optimal Control Applications and Methods*, vol. 3, pp. 101–114, 1982.
[42] D. H. Jacobson and D. Q. Mayne, *Differential Dynamic Programming*. Elsevier, NY, USA, 1970.
[43] D. Mayne and E. Polak, "First-order strong variation algorithms for optimal control," *Journal of Optimization Theory and Applications*, vol. 16, no. 3/4, pp. 277–301, 1975.
[44] D. M. Murray and S. J. Yakowitz, "Constrained differential dynamic programming and its application to multireservoir control," *Water Resources Research*, vol. 15, no. 5, pp. 1017–1027, 1979.
[45] S. Yakowitz, "The stagewise kuhn-tucker condition and differential dynamic programming," *IEEE Transactions on Automatic Control*, vol. 31, no. 1, pp. 25–30, 1986.
[46] W. Li and E. Todorov, "Iterative linearization methods for approximately optimal control and estimation of non-linear stochastic system," *International Journal of Control*, vol. 80, no. 9, pp. 1439–1453, 2007.
[47] H. J. Pesch, "Real-time computation of feedback controls for constrained optimal control problems. part 1: Neighbouring extremals," *Optimal Control Applications and Methods*, vol. 10, no. 2, pp. 129–145, 1989.
[48] Y. Chen, L. Roveda, and D. J. Braun, "Efficiently computable constrained optimal feedback controllers," *IEEE Robotics and Automation Letters*, October 2018.
[49] Q. Li, L. Chen, C. Tai, and E. Weinan, "Maximum principle based algorithms for deep learning," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 5998–6026, 2017.
[50] P. Abbeel, M. Quigley, and A. Y. Ng, "Using inaccurate models in reinforcement learning," in *International Conference on Machine Learning*, (Pittsburgh, Pennsylvania, USA), pp. 1–8, 2006.
[51] J. Nocedal and S. Wright, *Numerical Optimization*. Springer Science & Business Media, 2006.
[52] S. J. Wright, "Interior point methods for optimal control of discrete time systems," *Journal of Optimization Theory and Applications*, vol. 77, no. 1, pp. 161–187, 1993.
[53] B. Houska, H. J. Ferreau, and M. Diehl, "An auto-generated real-time iteration algorithm for nonlinear MPC in the microsecond range," *Automatica*, vol. 47, pp. 2279–2285, 2011.
[54] M. Vukov, S. Gros, G. Horn, G. Frison, K. Geebelen, J. B. Jørgensen, J. Swevers, and M. Diehl, "Real-time nonlinear MPC and MHE for a large-scale mechatronic application," *Control Engineering Practice*, vol. 45, pp. 64–78, 2015.
[55] D. S. Laila, D. Nešić, and A. R. Teel, "Open- and closed-loop dissipation inequalities under sampling and controller emulation," *European Journal of Control*, vol. 8, no. 2, pp. 109–125, 2002.
[56] D. Nesic and A. R. Teel, "A framework for stabilization of nonlinear sampled-data systems based on their approximate discrete-time models," *IEEE Transactions on Automatic Control*, vol. 49, pp. 1103–1122, July 2004.
[57] V. V. Aleksandrov, "On the accumulation of perturbations in the linear systems with two coordinates," *Vestnik MGU*, vol. 3, pp. 67–76, 1968.
[58] B. Järmark, "A new convergence control method in Differential Dynamic Programming," *Report TRITA-REG-7502, The Royal Institute of Technology, Stockholm*, pp. 1–52, 1975.
[59] M. R. Hestenes, *Optimization Theory: The Finite Dimensional Case*. John Wiley, 1975.
[60] R. E. Kálmán, "Contributions to the theory of optimal control," *Boletín de la Sociedad Matemática Mexicana*, vol. 5, no. 2, pp. 102–119, 1960.
[61] B. Molinari, "The time-invariant linear-quadratic optimal control problem," *Automatica*, vol. 13, no. 4, pp. 347–357, 1977.
[62] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis, "Adaptive optimal control for continuous-time linear systems based on policy iteration," *Automatica*, vol. 45, no. 2, pp. 477–484, 2009.
[63] P. O. M. Scokaert and J. B. Rawlings, "Constrained linear quadratic regulation," *IEEE Transactions on Automatic Control*, vol. 43, no. 8, pp. 1163–1169, 1998.
[64] Y. Jiang and Z.-P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, no. 10, pp. 2699–2704, 2012.
[65] T. Hu, Z. Lin, and B. M. Chen, "An analysis and design method for linear systems subject to actuator saturation and disturbance," *Automatica*, vol. 38, pp. 351–359, 2002.
[66] H. Modares, F. L. Lewis, and M.-B. Naghibi-Sistani, "Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems," *Automatica*, vol. 50, pp. 193–202, Jan 2014.
[67] Y. Jiang and Z.-P. Jiang, "Global adaptive dynamic programming for continuous-time nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 60, no. 11, pp. 2917–2929, 2015.
[68] S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu, "Regret bounds for robust adaptive control of the linear quadratic regulator," in *Advances in Neural Information Processing Systems*, pp. 4188–4197, 2018.
[69] J. Fong, Y. Tan, V. Crocher, D. Oetomo, and I. Mareels, "Dual-loop iterative optimal control for the finite horizon LQR problem with unknown dynamics," *Systems & Control Letters*, vol. 111, pp. 49–57, 2018.
[70] H. Xu, Q. Zhao, and S. Jagannathan, "Finite-horizon near-optimal output feedback neural network control of quantized nonlinear discrete-time systems with input constraint," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 8, pp. 1776–1788, 2015.
[71] N. Amann, D. H. Owens, and E. Rogers, "Iterative learning control using optimal feedback and feedforward actions," *International Journal of Control*, vol. 65, no. 2, pp. 277–293, 1996.
[72] D. H. Owens and J. Hätönen, "Iterative learning control–An optimization paradigm," *Annual Reviews in Control*, vol. 29, no. 1, pp. 57–70, 2005.
[73] A. Jadbabaie, J. Yu, and J. Hauser, "Unconstrained receding-horizon control of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 46, no. 5, pp. 776–783, 2001.
[74] W. Rudin *et al.*, *Principles of Mathematical Analysis*, vol. 3. McGraw-hill New York, 1964.
[75] A. V. Fiacco, *Introduction to sensitivity and stability analysis in nonlinear programming*. Elsevier, 1983.



Yuqing Chen received his B.Eng. and M.S. degrees in control science and engineering from Harbin Institute of Technology (HIT) in 2013 and 2015 respectively.

Since 2015, he has been working toward the Ph.D. degree at the Singapore University of Technology and Design. His research is focused on optimal control of dynamical systems.

David J. Braun (M'09) received the Ph.D. degree in mechanical engineering from Vanderbilt University, Nashville, TN, USA, in 2009.

He is Assistant Professor of Mechanical Engineering at Vanderbilt University (VU). Prior to joining VU, he was visiting researcher at the Institute for Robotics and Mechatronics at the German Aerospace Center (DLR), postdoctoral research fellow at the Statistical Machine Learning and Motor Control Group (SLMC) at the University of Edinburgh, and Assistant Professor at Singapore University of Technology and Design. Dr. Braun's research interests include dynamical systems, optimal control, and robotics. He received the 2014 IEEE Transactions on Robotics Best Paper Award. He was Scientific Program Co-Chair of the 2015 IEEE International Conference on Rehabilitation Robotics, Area Chair for the 2018 Robotics Science and Systems Conference, and Associate Editor for the 2020 IEEE International Conference on Robotics and Automation.

DOI: 10.1109/TAC.2020.2986211