A tractor-trailer parking control scheme using adaptive dynamic programming

Chenyong Guan¹ · Yu Jiang²

Received: 28 November 2020 / Accepted: 9 March 2021 / Published online: 23 March 2021
© The Author(s) 2021

Abstract

This paper studies the online learning control of a truck-trailer parking problem via adaptive dynamic programming (ADP). The contribution is twofold. First, a novel ADP method is developed for systems with parametric nonlinearities. It learns the optimal control policy of the linearized system at the origin, while the learning process utilizes online measurements of the full system and is robust with respect to nonlinear disturbances. Second, a control strategy is formulated for a commonly seen truck-trailer parallel parking problem, and the proposed ADP method is integrated into the strategy to provide online learning capabilities and to handle uncertainties. A numerical simulation is conducted to demonstrate the effectiveness of the proposed methodology.

Keywords Adaptive dynamic programming · Adaptive optimal control · Autonomous vehicles

Introduction

Parking a truck-trailer is a problem frequently studied in the fields of automated and autonomous trucking, robotics, and nonlinear control (see, for example, [2,10,25,29,30]). In particular, the backward steering control of wheeled multiple vehicles have been studied using neural network [23], fuzzy logic [11,35], and other learning algorithms. Different from adaptive cruise control, lane-keeping, lane-changing, or any other control actions typically happen on the highway or on secondary roads, truck-trailer parking maneuvers mostly occur in closed off-highway environment, such as cargo yards, distribution centers, or intermodal facilities. Thus, truck-trailer parking maneuvers have a few distinctive features. First, the vehicle speed is low, and the effects of tire slip can be ignored. Second, compared with lane-keeping tasks, much higher lateral accuracy is required for trailer

parking maneuvers. Third, backing up a truck-trailer system involves dealing with a naturally unstable equilibrium [2]. Fourth, quite a few different types of uncertainties, such as wheelbase length, load balance, and worn chassis, can cause the truck-trailer dynamics to deviate from nominal models.

To address uncertainties and to apply data-driven approaches that gradually improve the controller performance, this paper resorts to the theory of adaptive dynamic programming (ADP), which is a class of approximate methods of solving optimal control problems (see [5–7,27,31,36,41–44] and the references therein). ADP avoids the inherent *curse of dimensionality* problem of classical dynamic programming [4], and it is usually considered as a class of methods in reinforcement learning [33] in that it learns the optimal solution through iterations between approximating the value function and improving the action [24]. ADP has been extensively studied for Markov decision processes (see, for example, [6,27]), as well as dynamic systems (see the review papers [21,39]). Stability issues in ADP-based control systems design are addressed in [3,20,38]. A robustification of ADP, known as robust-ADP or RADP, is developed by taking into account dynamic uncertainties [15,17]. Another related work in robust dynamic programming is reported in [8]. Sum-of-squares program [28] has been introduced to ADP with a relaxation technique to achieve global asymptotic stabilization of nonlinear polynomial system in [14]. ADP methods for different type of game theories have been

✉ Yu Jiang
yjiang@gudsen.com; yu.jiang@nyu.edu
Chenyong Guan
cy@gudsen.com

¹ Gudsen Technology Co., Ltd., 6/F, 10th Building, Jiuxiang Ling Industrial Park, Ave Xili, Nanshan District, Shenzhen 518000, Guangdong, China

² Gudsen Engineering Inc, 844 Highland Ave, #533, Needham, MA 02493, USA

studied in [32,37,40]. ADP-based tracking control design can be found in [9,13,45], just to name a few. Some recent developments of ADP in control systems can be found in [16] and the references therein.

When dealing with static uncertain nonlinearities, neural network and general universal approximation methods [12,26] are widely adopted in ADP to approximate the cost function and the control policy. However, ADP with universal approximators may have at least two shortcomings. First, a large number of basis functions are usually required. Hence, it may incur a huge computational burden and slow adaptation for the learning system. Second, when the target function to approximate is treated as a black-box, it is not trivial to manage the approximation error to avoid it from being amplified across iterations, especially when implemented online, not to mention sometimes instability can be caused due to small approximation error.

In practice, many engineering systems do not need to be treated as black-boxes, because certain knowledge about the system, although limited, could be obtained prior to design ADP-based controllers. Indeed, quite a high percentage of engineering systems, such as the truck-trailer system studied in this paper, can be parametrized with a known small set of basic functions and uncertain parameters, of which the range can also be quantified. In this way, no heavy computation is needed during policy evaluation or policy improvement. Also, the potential approximation error is eliminated theoretically. Thus, the two shortcomings with universal approximation approaches when integrated into ADP-based online learning are addressed, as long as the system in question can be parametrized. This paper will develop such an approach with detailed analysis.

In summary, the major contributions of this paper are twofold. First, a novel ADP methodology is proposed to learn the optimal solution of the uncertain linearized system, while at the same time to handle parametrized nonlinear uncertainties during online learning. Second, the proposed ADP method is incorporated into a truck-trailer parking control strategy, designed for a commonly seen truck-trailer parallel parking problem.

The remainder of this paper is organized as follows. The next section formulates the problem and introduces some basic results regarding nonlinear optimal control followed by which a novel ADP method for nonlinear systems with parametric uncertainties is developed. The subsequent section details a specific truck-trailer control problem, with analysis on its dynamics. Then a human-inspired control strategy to achieve parallel parking in the presence of parametric uncertainties is developed. This control strategy integrates the proposed ADP method. In the penultimate section, the numerical simulation results to validate the efficiency and effectiveness of the proposed method are summarized. The

final section gives concluding remarks and points out potential topics for future work.

Notation Throughout this paper, we use $\mathbb{R}$ and $\mathbb{Z}_+$ to denote the sets of real numbers and non-negative integers, respectively. Vertical bars $\|\cdot\|$ represent the Euclidean norm for vectors, or the induced matrix norm for matrices. We use $\otimes$ to indicate Kronecker product, and $\text{vec}(A)$ is defined to be the mn -vector formed by stacking the columns of $A \in \mathbb{R}^{n \times m}$ on top of one another, i.e., $\text{vec}(A) = [a_1^T \ a_2^T \ \dots \ a_m^T]^T$, where $a_i \in \mathbb{R}^n$ are the columns of A . A control law is also called a *policy*. A feedback gain matrix $K \in \mathbb{R}^{m \times n}$ is said to be *stabilizing* for linear systems $\dot{x} = Ax + Bu$ if the feedback matrix $A - BK$ is Hurwitz.

Problem formulation and mathematical preliminaries

Problem formulation

This paper studies uncertain nonlinear systems that can be represented in the following form:

$$\dot{x} = A(x) + Bu, \quad (1)$$

where $x \in \mathbb{R}^n$ is the system state, $u \in \mathbb{R}^m$ is the control input, $A(x) \in \mathbb{R}^n$ is a smooth and uncertain state-dependent vector, and $B \in \mathbb{R}^{n \times m}$ is an uncertain constant matrix. The system is assumed to be controllable at the origin.

Remark 1 Without loss of generality, we can assume

$$A(x) = Ax + \Delta A\sigma(x), \quad (2)$$

where $A = A(0) \in \mathbb{R}^{n \times n}$ and $\Delta A \in \mathbb{R}^{n \times q}$ are uncertain constant matrices, and $\sigma(x) \in \mathbb{R}^q$ is a known vector of linearly independent functions of x , vanishing at the origin. Also, (A, B) is controllable.

The control objective is to design an ADP-based control system that learns, through online data, the optimal control policy that minimizes the performance index

$$J = \int_0^\infty (x^T Qx + u^T Ru) d\tau \quad (3)$$

of the system (1) linearized at the origin. It is assumed that there exists a constant matrix C with suitable dimensions such that the weight matrix $Q \in \mathbb{R}^{n \times n}$ satisfy $Q = C^T C$ and the pair (A, C) is observable. The other weight matrix $R \in \mathbb{R}^{m \times m}$ is required to be symmetric and positive definite.

Remark 2 A and B are referred to as *uncertain* constant matrices, in the sense that their precise values are not required

to be known. In practice, it is always reasonable to have a good estimate of the range of uncertain parameters, and a stabilizing state-feedback gain K_0 , although not necessarily optimal, can be assumed.

Remark 3 The formulated problem is strongly related to the robust-ADP [15] problem but is also slightly different. In this paper, no dynamic uncertainty is considered, and the goal is to learn the optimal control policy for the linearized model. The online learning process has to be robust against the nonlinear perturbed term $\Delta A\sigma(x)$. Further, the learned control policy is optimal for the linearized model, and the closed-loop system comprised of the original system (1) and the control policy is locally asymptotically stable at the origin.

Linear optimal control and policy iteration

By linear optimal control theory [22], solutions to the problem described in “Problem formulation” can be found by solving the well-known algebraic Riccati equation (ARE)

$$A^T P + P A + Q - P B R^{-1} B^T P = 0, \quad (4)$$

if A and B are accurately known.

In addition, under the assumptions mentioned above, (4) has a unique symmetric positive definite solution $P = P^*$, and the optimal control policy is in the form of

$$u = -K^* x, \quad (5)$$

where the optimal feedback gain matrix K^* is then be determined by

$$K^* = R^{-1} B^T P^*. \quad (6)$$

One of the numerical methods for solving (4) is developed in [18] and summarized in Theorem 1 below. This methodology is related to *policy iteration* as in reinforcement learning [34], since it starts with a stabilizing feedback control policy, and during each iteration the associated LQR cost is computed and then used for improving the policy.

Theorem 1 Let $K_0 \in \mathbb{R}^{m \times n}$ be any stabilizing feedback gain matrix, and let P_k be the symmetric positive definite solution of the Lyapunov equation

$$(A - B K_k)^T P_k + P_k (A - B K_k) + Q + K_k^T R K_k = 0, \quad (7)$$

where K_k , with $k = 1, 2, \dots$, are defined recursively by

$$K_k = R^{-1} B^T P_{k-1}. \quad (8)$$

Then the following properties hold:

1. $A - B K_k$ is Hurwitz,
2. $P^* \leq P_{k+1} \leq P_k$,
3. $\lim_{k \rightarrow \infty} K_k = K^*$, $\lim_{k \rightarrow \infty} P_k = P^*$.

The iteration algorithm described in Theorem 1 has guaranteed convergence. However, it does require the perfect knowledge of the system matrices A and B . A novel ADP methodology to implement the same iteration while not using the knowledge of system matrices but online data measurements will be developed next.

Adaptive dynamic programming and parametric uncertainties

In this section, a novel approach to learn the linear optimal controller that solves the problem in “Problem formulation” will be developed. This approach makes use of the data generated from the nonlinear plant (1), without the need to identify any uncertain system parameter.

To begin with, let K_k be a stabilizing control gain matrix, and let P_k denote the symmetric, positive definite, and unique solution to the Lyapunov function (7). Next, apply the following control policy

$$u = -K_k x + e \quad (9)$$

with e an exploration noise. Then, along the trajectories of the closed-loop system comprised of (1) and (9), it yields that

$$\begin{aligned} \frac{d(x^T P_k x)}{dt} &= x^T P_k (A(x) - B K_k x + B e) \\ &\quad + (A(x) - B K_k x + B e)^T P_k x \\ &= x^T P_k (A - B K_k) x + x^T (A - B K_k)^T P_k x \\ &\quad + 2x^T P_k \Delta A \sigma(x) + 2x^T P_k B_k e. \end{aligned} \quad (10)$$

Together with (7), it follows that

$$\begin{aligned} \frac{d(x^T P_k x)}{dt} &= -x^T (Q + K_k^T R K_k) x + 2x^T P_k \Delta A \sigma(x) \\ &\quad + 2x^T P_k B_k e. \end{aligned} \quad (11)$$

Next, combining with (8), and defining $L_k = P_k \Delta A$ we have

$$\begin{aligned} \frac{d(x^T P_k x)}{dt} &= -x^T (Q + K_k^T R K_k) x + 2x^T L_k \sigma(x) \\ &\quad + 2x^T K_{k+1}^T R e. \end{aligned} \quad (12)$$

Now, given any finite time interval $[t, t + \delta t]$, we can integrate both sides of (12) with respect to time on the interval to obtain

$$x^T(t + \delta t) P_k x(t + \delta t) - x^T(t) P_k x(t)$$

$$= - \int_t^{t+\delta t} x^T (Q + K_k^T R K_k) x d\tau + 2 \int_t^{t+\delta t} x^T L_k \sigma(x) d\tau + 2 \int_t^{t+\delta t} x^T K_{k+1}^T R e \tau. \quad (13)$$

It is easy to see that the pair (P_k, K_{k+1}) satisfying (7) and (8) must satisfy (13), which illustrates a way of solving (P_k, K_{k+1}) with linear regression. Indeed, defining

$$r_k(x) = x^T (Q + K_k^T R K_k) x, \quad (14)$$

and using Kronecker product representation, (13) can be rewritten as

$$\begin{aligned} & (x^T \otimes x^T|_t^{t+\delta t}) \text{vec}(P_k) \\ & - 2 \left(\int_t^{t+\delta t} \sigma^T(x) \otimes x^T d\tau \right) \text{vec}(L_k) \\ & - 2 \left(\int_t^{t+\delta t} (Re)^T \otimes x^T d\tau \right) \text{vec}(K_{k+1}^T) \\ & = - \int_t^{t+\delta t} r_k(x) d\tau. \end{aligned} \quad (15)$$

It is not difficult to notice that if the same process of deriving (15) is applied to multiple time-interviews, we can then obtain a set of equations in the form of (15) to solve for the P_k , K_{k+1} , and L_k .

To see this, let

$$\zeta_k^{(k)} = \begin{bmatrix} x \otimes x|_{t_i}^{t_{i+1}} \\ -2 \int_{t_i}^{t_{i+1}} \sigma(x) \otimes x d\tau \\ -2 \int_{t_i}^{t_{i+1}} (Re) \otimes x d\tau \end{bmatrix}, \quad (16)$$

$$\xi_i^{(k)} = - \int_{t_i}^{t_{i+1}} r_k(x) d\tau, \quad (17)$$

$$\Theta_k = [\zeta_1^{(k)} \zeta_2^{(k)} \cdots \zeta_{l_k}^{(k)}]^T, \quad (18)$$

$$\Xi_k = [\eta_1^{(k)} \eta_2^{(k)} \cdots \eta_{l_k}^{(k)}]^T. \quad (19)$$

where $0 < t_1 < t_2 < \cdots < t_{l_{k+1}}$, with l_{k+1} a sufficiently large integer. Then we have

$$\Theta_k \text{vec}([P_k \ K_{k+1}^T \ L_k]) = \Xi_k. \quad (20)$$

Note that if the linear equation (20), together with $P_k = P_k^T$, has a unique solution, then solving them amounts to solve both (7) and (8). Hence, let us impose the following assumption.

Assumption 3 Given a stabilizing K_k and an exploration noise $e(t)$, there exists a sufficiently large integer $l_k > 0$,

such that

$$\text{rank}(\Theta_k) = \frac{n(n+1)}{2} + nq + nm. \quad (21)$$

Lemma 1 Under Assumption 3, given a stabilizing K_k , the $P_k = P_k^T$ and K_{k+1} computed from (20) must satisfy both (7) and (8).

Proof First, based on the derivations from (9) to (20), we can see (P_k, K_{k+1}) computed from (7) and (8) do satisfy (20).

Second, under Assumption 3, all the columns other than the $\frac{n(n-1)}{2}$ duplicated ones are linearly independent. That means, if we restrict P_k to be symmetric, the solution of (8) is unique. Hence, the P_k and K_{k+1} of that unique solution must satisfy (7) and (8).

Now, we are ready to give an online policy iteration scheme. Similar as in other policy-iteration-based iterative schemes, a stabilizing feedback gain matrix K_0 is assumed.

Algorithm 1 Initialization:

Find a stabilizing feedback gain matrix K_k with $k = 0$.

(2) Online Data Collection:

Apply the following control policy to system (1)

$$u = -K_k x + e. \quad (22)$$

Then construct the linear regression matrix and incrementally increase $l_k \in \mathbb{Z}_+$, until the rank condition (21) is satisfied.

(3) Policy Evaluation and Improvement:

Solve for P_k , K_{k+1} , and L_k from (20). Then, go to Step 2) with k replaced by $k + 1$.

The convergence of Algorithm 3 is guaranteed under Assumption 3 and is summarized in the theorem below

Theorem 2 Under Assumption 3 and given a stabilizing K_0 , we have

1. $A - BK_k$ is Hurwitz
2. $\lim_{k \rightarrow \infty} K_k = K^*$, $\lim_{k \rightarrow \infty} P_k = P^*$, and $\lim_{k \rightarrow \infty} L_k = P \Delta A^*$,

where K_{k+1} , P_k , and L_k are obtained from Algorithm 3, for $k = 0, 1, 2, \dots$, P^* is the optimal solution of (4) and $K^* = R^{-1} B^T P^*$.

Proof Under Assumption 3, the iterations in Algorithm 3 is equivalent to the ones in (7) and (8). Then, by Lemma 1, the results hold. Thus, the proof is complete.

In practical implementation, one can introduce a predefined threshold $\epsilon > 0$ to check if

$$\|P_{k+1} - P_k\| < \epsilon \quad (23)$$

and to determine if exploration noise e and online learning are still needed. Thus, the exploration/exploitation trade-off can be balanced. Indeed, a larger ϵ may lead to shorter exploration time and therefore will allow the system to implement the noise-free control policy sooner. On the other hand, using a smaller $\epsilon > 0$ allows the learning system to better improve the control policy but longer learning time may be needed to achieve desired convergence.

A truck-trailer parking problem

Problem description

The truck-trailer system considered in this paper is with an *on-axle* hitch, which lies on the real axle of the truck [1]. A typical truck of this type are referred to as a terminal tractor, also known as a yard truck. It is an off-highway semi-tractor intended to move semi-trailers, within a cargo yard, warehouse facility, or intermodal facility.

One typical use case of a terminal tractor is shown in Fig. 1, where a trailer needs to be parked into a parallel parking spot. Due to the space limitation, there are obstacles on each side and the width of the aisle available for making maneuvers is usually no more than 20 m. There can be other sporadic obstacles in the aisle, such as over-length trailers and other temporarily parked tractors.

Fig. 1 A truck-trailer parallel parking problem. The trailer (red) is required to be parked into a parallel spot, as indicated by the two lines. There are 20 m longitudinal clearance for the truck (blue) to make use of. The gray areas represent obstacles

A truck-trailer model

We consider a truck-trailer system as shown in Fig. 2, and can be presented in the following form: [19]

$$\dot{x} = v \cos(\theta) \quad (24)$$

$$\dot{y} = v \sin(\theta) \quad (25)$$

$$\dot{\theta} = \frac{v}{D} \tan(\delta) \quad (26)$$

$$\dot{\gamma} = -\frac{v}{L} \sin(\gamma) - \frac{v}{D} \tan(\delta), \quad (27)$$

where x and y are the coordinate of the reference point (rear wheel axle) of the truck, or the location of the truck kingpin; v is the longitudinal velocity measured at the reference point; δ is the steering angle; θ is the orientation/heading of the truck; γ is the relative angle between the truck and the trailer, note that $\gamma + \theta$ gives the orientation of the trailer; D is the wheelbase of the truck and L is the wheelbase of the trailer.

Without loss of generality, our control objective is to drive the truck and trailer to the origin $(x, y, \theta, \gamma) = 0$. Indeed, if the target position is not at the origin of the current coordinate system, we can always create a new coordinate at the target with the desired truck heading being the new x -axis and establish the transformation between the two coordinates.

Longitudinal and lateral dynamics and control

The truck-trailer dynamics (24)–(27) is highly nonlinear and under-actuated. To simplify the problem, we separate the consideration of the longitudinal and lateral controllers.

Fig. 2 A truck-trailer system with an on-axis hitch. The location of the truck-trailer is defined at the center of the rear axle of the truck

We consider the longitudinal controller takes the form of

$$v = -v_0 \operatorname{sgn}(x), \quad (28)$$

where $v_0 > 0$ is a constant, and sgn is the sign function. In addition, as soon as the truck-trailer hits the obstacle, which is inflated with safety margin, the speed is immediately set to zero.

As for lateral dynamics, it can be observed that if the steering control policy only depends on the four state variables, then the overall path geometry of the truck-trailer system is independent of the longitudinal velocity, as long as it is non-zero and does not change signs. Furthermore, the linearized system of (24)–(27) at the origin is uncontrollable. However, since the movement along the x -axis can mostly be taken care of by the longitudinal control, the lateral control policy only needs to focus on the lateral error dynamics (25)–(27), of which the linearized system at the origin is controllable. Indeed, the lateral dynamics for forward movement becomes

$$\dot{y} = \sin(\theta) \quad (29)$$

$$\dot{\theta} = \frac{u}{D} \quad (30)$$

$$\dot{\gamma} = -\frac{1}{L} \sin(\gamma) - \frac{u}{D} \quad (31)$$

and one for backward movement is

$$\dot{y} = -\sin(\theta) \quad (32)$$

$$\dot{\theta} = -\frac{u}{D} \quad (33)$$

$$\dot{\gamma} = \frac{1}{L} \sin(\gamma) + \frac{u}{D}. \quad (34)$$

It is easily understandable that driving a truck-trailer backward is much more difficult than driving it forward. This can be seen by comparing the γ -subsystem in (31) and (34). Indeed, with $u \equiv 0$, the linearized γ -subsystem has a negative eigenvalue $-\frac{1}{L}$ for forward maneuvers, but a positive eigenvalue $\frac{1}{L}$ for backward maneuvers. In other words, if a truck is moving forward along a straight line, the hitch angle γ will converge to 0. However, if it is move backwards along a straight line, γ will quickly diverge.

Steering control

Based on the analysis made in “Longitudinal and lateral dynamics and control”, we design the forward and backward steering control policies can be designed differently.

First, in the presence of the uncertain parameter D of which the range is known, it is practical to design a linear control policy

$$u = -K_f \begin{bmatrix} y \\ \theta \end{bmatrix} \quad (35)$$

that locally stabilizes the subsystem comprised of (29) and (30). Due to the stable nature of the γ -subsystem, the overall system (29) together with the control policy (35) is locally asymptotically stable. Of course, one can make the control policy depend on γ , but the performance improvement is less likely to be significant in the parallel parking problem.

Second, for backward movement, the steering control policy to be designed is a linear feedback controller

$$u = -K_b \begin{bmatrix} y \\ \theta \\ \gamma \end{bmatrix} \quad (36)$$

that stabilizes (32)–(34), and at the same time minimize a given cost function (3). This problem could not be solved directly using conventional LQR approach, due to the uncertain parameters D and L . Thus, we will apply the ADP approached developed in this paper to find the control policy for backward steering.

Control strategy involving ADP

A strategy inspired from human drivers

A standard parking strategy we observed from human drivers involves at least two intermediate spots, as shown in Fig. 3.

Fig. 3 Intermediate targets to achieve the parallel parking maneuver. To begin with, a human driver would first aim at a target left to the goal, and make the trailer pointing approximately the spot on the right side of the target spot. Once the truck-trailer is getting sufficiently close to the final target after backing, the intermediate spot is then chosen to be the one at the center front

An experienced human driver would first drive towards intermediate target 1, at which the end of the trailer would point towards the spot on the right side of the desired spot. Then the driver would drive backwards to reach the final target. If the accuracy at the end of the backward movement is still large, the forward movement is repeated. Nevertheless, if the reference point of the truck has small lateral error, the next action to take would be a forward movement towards the intermediate target 2, followed by a backward movement to reach the final target. This back-and-forth adjustment can be repeated until desired accuracy is met.

Here, we follow the general strategy as a human driver would take, but design feedback controllers to automate the steering and speed control. In addition, the ADP-based methodology is incorporated to the steering control for backward movements.

High-level control strategy involving ADP

We assume all four state variables are instantaneously measurable. Indeed, the location of the truck and its orientation can be accurately measured by real-time kinematic (RTK) GPS. The hitch angle can be measured by a physical encoder, or by the means of computer vision (see [10], for example).

Next, to solve the parallel parking problem, we propose a high-level control strategy as shown in Fig. 4. From starting position, the truck-trailer will first drive forward towards target 1, until the longitudinal criterion is reached or a safety margin between the truck and the obstacle is met. Then the truck starts to back-up the trailer towards the dock door. When backup is complete, the parking error is evaluated to decide if a pull-up adjustment is needed, and which target should be aimed for.

Fig. 4 A high-level human-like control strategy to achieve the desired parallel parking

The steering control policy is always fixed when the truck is moving forward. As for backward driving, the steering control policy will be updated via the proposed ADP method whenever a backing maneuver is finished.

Change of coordinates for non-origin targets

The proposed control design methodology is based on the assumption that the truck-trailer always need to be stabilized at the origin. Therefore, when we are making the truck-trailer to reach a target that is not at the origin of the current coordinated system, we need to compute the error signals in a new coordinate originated at the target, such that a desired control input can be correctly computed. Indeed, we can simply perform the following coordinate transformation:

$$\bar{x} = (x - x') \cos(\theta') + (y - y') \sin(\theta') \quad (37)$$

$$\bar{y} = -(x - x') \sin(\theta') + (y - y') \cos(\theta') \quad (38)$$

$$\bar{\theta} = \theta - \theta' \quad (39)$$

$$\bar{\gamma} = \gamma \quad (40)$$

and feedback control gains are to be applied to these converted error signals.

Note that the steering angle and the control input u remain the same, under different coordinates.

Numerical simulation

Simulation setup

The simulation is programmed and conducted in MATLAB R2020a. All the ordinary differential equations are solved using forward Euler method with a fixed time step of $0.2s$. Simulation code are fully accessible in the GitHub repository <https://github.com/yu-jiang/padp>.

The initial condition of the truck-trailer is set to $(x, y, \theta, \gamma) = (0, -5, 0, 0)$, and the final parking goal is at the origin. The intermediate target 1 is set to $(x, y, \theta, \gamma) = (20, 2, 0.3, 0)$, and target 2 at $(x, y, \theta, \gamma) = (20, 0, 0, 0)$. Criterion to decide forward adjustment depends on the final lateral accuracy y , and is defined as follows:

1. $y \geq 0.3$, drive to target 1.
2. $y \leq 0.3$ and $|y| > 0.05$, drive to target 2.
3. $|y| \leq 0.05$, task complete.

The truck wheelbase is set to 3.0m and the trailer wheelbase is set to 11.0m. Note that both these values are blind to the controller. The Q and R weight matrices are set to be $Q = I_3$ and $R = 100$.

We assume the initial steering control policies, or the feedback gains, are computed for a truck-trailer system with $D = 3.0$ m and $L = 6.0$ m. For backward steering, and the weight matrices were $Q = I_3$ and $R = 1000$. Hence, the gain is computed as

$$K_b = \begin{bmatrix} -0.0316 & 0.8164 & 2.2538 \end{bmatrix}. \quad (41)$$

As for forward steering, we only focus on the truck dynamics, i.e., (29) and (30) with $\gamma \equiv 0$, since the trailer subsystem is local stable by itself when moving forward. Then we set $Q = I_2$ and $R = 1000$ and solve for the first two elements of the control gain, and let the third element be zero. Thus, the forward controller gain becomes

$$K_f = \begin{bmatrix} 0.0316 & 0.4367 & 0 \end{bmatrix}. \quad (42)$$

We simulated the “no-learning” as well as the “learning with ADP” scenarios, and they both start with the same control policies. The “no-learning” scenario always stuck to the same control policies, while the “learning with ADP” scenario kept updating the backward steering control policy as soon as a backward maneuver is complete. In both scenarios, we set five as the maximum number trials allowed.

Fig. 5 Simulation results and comparison between using a fixed steering control policy and the proposed ADP-based control strategy

Simulation results

The simulation results are shown in Figs. 5 and 6. An animation of the full-simulation can be found at https://youtu.be/CFPBQ_DP4Nc.

Each forward and backward movement combination in this simulation is referred to as a *trial*. In the first trial, one can see that in both cases (i.e., no learning, and with ADP-based learning), the performance is the same. This is because the ADP-based control strategy uses the same initial control policy as the no-learning case. However, starting from the second trial, the ADP-base control strategy starts to perform online learning and gradually modify the control policy towards the optimal solution. After four trials, the ADP-based control strategy has parked the trailer into the spot with insignificant lateral error. Due to the control strategy, the intermediate point switched to the front of the spot, and the last trial resulted in desired parking accuracy. On the other hand, without online learning and always under the initial control policy, the truck-trailer system did not make any notable progress after five trials, compared with the initial condition before trial one.

Finally, after five iterations, the learned feedback control gain matrix for backward steering is

$$K_b^{(5)} = \begin{bmatrix} -0.1000 & 2.9863 & 4.3194 \end{bmatrix}. \quad (43)$$

For validation purpose, we computed the ideal feedback gain K_b^* using the precise system matrices, and obtain

$$K_b^* = \begin{bmatrix} -0.1000 & 2.9843 & 4.3163 \end{bmatrix}. \quad (44)$$

Fig. 6 Comparison between using a fixed steering control policy and using the proposed ADP-based control strategy

Similarly, the estimated cost matrix after five iterations $P^{(5)}$ and the ideal optimal cost P^* are

$$P^{(5)} = 10^{-4} \begin{bmatrix} 0.0030 & -0.0445 & -0.0475 \\ -0.0445 & 0.8058 & 0.8954 \\ -0.0475 & 0.8954 & 1.0250 \end{bmatrix} \quad (45)$$

and

$$P^* = 10^{-4} \begin{bmatrix} 0.0030 & -0.0445 & -0.0475 \\ -0.0445 & 0.8051 & 0.8947 \\ -0.0475 & 0.8947 & 1.0241 \end{bmatrix}. \quad (46)$$

The approximation error is expected to be further reduced as the iterative process continues.

Conclusions and future work

In conclusion, this paper has presented a novel and practical ADP approach to handle nonlinear systems with parametric uncertainties. The proposed methodology makes use of the online data directly measured from the nonlinear plant and can learn the linear optimal controller with respect to the linearized system at the origin. Then, the methodology has been integrated into a control strategy to achieve precise truck-trailer parking, giving parametric uncertainties from the environment.

There are several related topics deserve further investigations in the future. First, it is interesting to figure out how to extend the proposed ADP method into more general truck-trailer maneuvers, such as alley dock backing in which curves are to be tracked. Second, in this paper we only incorporated ADP into the backward steering control, it will be very useful to see if ADP-like ideas can be developed to dynamically change the intermediate points and actively avoid obstacles. Finally, conducting real-world experiments with truck and trailers can further demonstrate the effectiveness of the proposed method.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the

permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Altafini C (2002) Following a path of varying curvature as an output regulation problem. *IEEE Trans Autom Control* 47(9):1551–1556
- Altafini C, Speranzon A, Wahlberg B (2001) A feedback control scheme for reversing a truck and trailer vehicle. *IEEE Trans Robot Autom* 17(6):915–922
- Balakrishnan SN, Ding J, Lewis FL (2008) Issues on stability of ADP feedback controllers for dynamical systems. *IEEE Trans Syst Man Cybern Part B: Cybern* 38(4):913–917
- Bellman R (1957) *Dynamic programming*. Princeton University Press, Princeton
- Bertsekas DP (2007) *Dynamic programming and optimal control*, 4th edn. Athena Scientific Belmont, Belmont
- Bertsekas DP, Tsitsiklis JN (1996) *Neuro-dynamic programming*. Athena Scientific, Nashua
- Bian T, Jiang ZP (2016) Value iteration and adaptive dynamic programming for data-driven adaptive optimal control design. *Automatica* 71:348–360
- Bian T, Jiang ZP (2019) Continuous-time robust dynamic programming. *SIAM J Control Optim* 57(6):4150–4174
- Chen C, Modares H, Xie K, Lewis FL, Wan Y, Xie S (2019) Reinforcement learning-based adaptive optimal exponential tracking control of linear systems with unknown dynamics. *IEEE Trans Autom Control* 64(11):4423–4438
- Hafner M, Pilutti T (2017) Control for automated trailer backup. In: SAE Technical Paper, 2017-01-0040 <https://doi.org/10.4271/2017-01-0040>
- Halgamuge SK, Runkler TA, Glesner M (1994) A hierarchical hybrid fuzzy controller for real-time reverse driving support of vehicles with long trailers. In: *Proceedings of 1994 IEEE 3rd international fuzzy systems conference*, vol 2, pp 1207–1210
- Hornik K, Stinchcombe M, White H (1989) Multilayer feedforward networks are universal approximators. *Neural Netw* 2(5):359–366
- Jiang Y, Fan JL, Chai TY, Lewis FL, Li JN (2018) Tracking control for linear discrete-time networked control systems with unknown dynamics and dropout. *IEEE Trans Neural Netw Learn Syst* 29(10):4607–4620
- Jiang Y, Jiang ZP (2015) Global adaptive dynamic programming for continuous-time nonlinear systems. *IEEE Trans Autom Control* 60(11):2917–2929
- Jiang Y, Jiang ZP (2017) *Robust adaptive dynamic programming*. Wiley, New York
- Jiang ZP, Bian T, Gao W (2020) Learning-based control: a tutorial and some recent results. *Found Trends® Syst Control* 8(3):176–284. <https://doi.org/10.1561/26000000023>
- Jiang ZP, Jiang Y (2013) Robust adaptive dynamic programming for linear and nonlinear systems: an overview. *Eur J Control* 19(5):417–425
- Kleinman D (1968) On an iterative technique for Riccati equation computations. *IEEE Trans Autom Control* 13(1):114–115
- Leng Z, Minor M (2010) A simple tractor-trailer backing control law for path following. In: *2010 IEEE/RSJ international conference on intelligent robots and systems*, Taipei, Taiwan, pp 5538–5542
- Lewis FL, Liu D (eds) (2013) *Reinforcement learning and approximate dynamic programming for feedback control*. Wiley, Hoboken
- Lewis FL, Vrabie D (2009) Reinforcement learning and adaptive dynamic programming for feedback control. *IEEE Circuits Syst Mag* 9(3):32–50
- Lewis FL, Vrabie D, Syrmos VL (2012) *Optimal control*, 3rd edn. Wiley, New York
- Nguyen D, Widrow B (1989) The truck backer-upper: an example of self-learning in neural networks. In: *International 1989 joint conference on neural networks*, vol 2, pp 357–363
- Odekunle A, Gao W, Davari M, Jiang ZP (2020) Reinforcement learning and non-zero-sum game output regulation for multi-player linear uncertain systems. *Automatica* 112:108672
- Park GH, Pao YH (1998) Training neural-net controllers with the help of trajectories generated with fuzzy rules (demonstrated with the truck backup task). *Neurocomputing* 18(1–3):91–105
- Park J, Sandberg IW (1991) Universal approximation using radial-basis-function networks. *Neural Comput* 3(2):246–257
- Powell WB (2007) *Approximate dynamic programming: solving the curses of dimensionality*. Wiley, New York
- Prajna S, Papachristodoulou A, Wu F (2004) Nonlinear control synthesis by sum of squares optimization: a Lyapunov-based approach. In: *Proceedings of the Asian control conference*, pp 157–165
- Ritzen P, Roebroek E, Van De Wouw N, Jiang ZP, Nijmeijer H (2016) Trailer steering control of a tractor-trailer robot. *IEEE Trans Control Syst Technol* 24(4):1240–1252. <https://doi.org/10.1109/TCST.2015.2499699>
- Ritzen P, Roebroek E, van de Wouw N, Jiang ZP, Nijmeijer H (2015) Trailer steering control of a tractor-trailer robot. *IEEE Trans Control Syst Technol* 24(4):1240–1252
- Si J, Barto AG, Powell WB, Wunsch DC et al (eds) (2004) *Handbook of learning and approximate dynamic programming*. Wiley Inc, Hoboken
- Song R, Wei Q, Zhang H, Lewis FL (2019) Discrete-time non-zero-sum games with completely unknown dynamics. *IEEE Trans Cybern* <https://doi.org/10.1109/TCYB.2019.2957406>
- Sutton RS, Barto AG (1998) *Reinforcement learning: an introduction*. Cambridge University Press, Cambridge
- Sutton RS, Barto AG (2018) *Reinforcement learning: an introduction*. MIT Press, Cambridge
- Tanaka K, Sano M (1994) A robust stabilization problem of fuzzy control systems and its application to backing up control of a truck-trailer. *IEEE Trans Fuzzy Syst* 2(2):119–134
- Vamvoudakis KG, Lewis FL (2010) Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem. *Automatica* 46(5):878–888
- Vamvoudakis KG, Lewis FL (2011) Multi-player non-zero-sum games: online adaptive learning solution of coupled Hamilton–Jacobi equations. *Automatica* 47(8):1556–1569
- Vrabie D, Vamvoudakis KG, Lewis FL (2013) *Optimal adaptive control and differential games by reinforcement learning principles*. The Institution of Engineering and Technology, London
- Wang FY, Zhang H, Liu D (2009) Adaptive dynamic programming: an introduction. *IEEE Comput Intell Mag* 4(2):39–47
- Wei Q, Song R, Yan P (2015) Data-driven zero-sum neuro-optimal control for a class of continuous-time unknown nonlinear systems with disturbance using ADP. *IEEE Trans Neural Netw Learn Syst* 27(2):444–458
- Werbos P (1974) Beyond regression: new tools for prediction and analysis in the behavioral sciences. Ph.D. thesis, Harvard University Comm Appl Math
- Werbos P (1977) Advanced forecasting methods for global crisis warning and models of intelligence. *Gen Syst Yearb* 22:25–38
- Werbos P (2013) Reinforcement learning and approximate dynamic programming (RLADP)—foundations, common misconceptions and the challenges ahead. In: Lewis FL, Liu D (eds) *Reinforcement learning and approximate dynamic programming for feedback control*. Wiley, Hoboken, pp 3–30
- Yang Y, Vamvoudakis KG, Modares H, Yin Y, Wunsch DC (2020) Hamiltonian-driven hybrid adaptive dynamic programming. *IEEE*

Trans Syst Man Cybern: Syst <https://doi.org/10.1109/TSMC.2019.2962103>.

45. Zhang H, Cui L, Zhang X, Luo Y (2011) Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method. IEEE Trans Neural Netw 22(12):2226–2236

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.