
ELSEVIER
Volume 22 • Issue 7 • August 2012
ISSN 0959-1524
Journal of
Process Control
IFAC
A Journal of IFAC
The International Federation
of Automatic Control
www.elsevier.com/locate/jprocont

Journal of Process Control 22 (2012) 1161–1170

Contents lists available at SciVerse ScienceDirect

Journal of Process Control

journal homepage: www.elsevier.com/locate/jprocont

Online policy iteration algorithm for optimal control of linear hyperbolic PDE systems

Biao Luo, Huai-Ning Wu *

Science and Technology on Aircraft Control Laboratory, School of Automation Science and Electrical Engineering, Beihang University (Beijing University of Aeronautics and Astronautics), Beijing 100191, PR China

ARTICLE INFO

Article history:
Received 22 November 2011
Received in revised form 19 April 2012
Accepted 26 April 2012
Available online 1 June 2012

Keywords:
Hyperbolic PDE systems
Distributed optimal control
Policy iteration
Space-dependent Riccati differential equation
Convergence

ABSTRACT

In this paper, the linear quadratic (LQ) optimal control problem is considered for a class of linear distributed parameter systems described by first-order hyperbolic partial differential equations (PDEs). Reinforcement learning (RL) technique is introduced for adaptive optimal control design from the design-then-reduce (DTR) framework. Initially, a policy iteration (PI) algorithm is proposed, which learns the solution of the space-dependent Riccati differential equation (SDRDE) online without requiring the internal system dynamics of the PDE system. To prove its convergence, the PI algorithm is shown to be equivalent to an iterative procedure of a sequence of space-dependent Lyapunov differential equations (SDLDEs). Then, the convergence is established by showing that the solutions of SDLDEs are a monotone non-increasing sequence that converges to the solution of the SDRDE. For implementation purpose, an online least-square method is developed for the approximation of the solutions of the SDLDEs. Finally, the proposed design method is applied to the distributed control of a steam-jacketed tubular heat exchanger to illustrate its effectiveness.

1. Introduction

In practice, a significant number of industrial processes are inherently distributed in space so that their behaviors depend on spatial position as well as time, such as heat conduction, fluid flow, and chemical reactor processes [1,2]. The mathematical models of these processes usually take the form of partial differential equations (PDEs) and are often derived from the fundamental balances of momentum, energy, and material. The well-known classification of PDE systems could be hyperbolic, parabolic, elliptic, etc. [1], according to the properties of the spatial differential operator. Due to their infinite-dimensional nature, it is very difficult directly using control design methods of lumped parameters systems (LPSs) for these PDE systems.

Optimal control theory is an important tool for the controller synthesis. The main objective of the optimal control policy is to regulate a dynamic system by minimizing a given cost criterion. Over the past decades, the optimal control theory of PDE systems has been well studied from mathematical point of view [3–5]. Meanwhile, the optimal control problem of PDE systems has also been well studied from engineering point of view, where a considerable attention has been paid to methods that are based on the minimization of linear quadratic (LQ) cost criteria. Existing works on the optimal controller design of PDE systems can be classified into two types: *reduce-then-design* (RTD) [6–10] and *design-then-reduce* (DTR) [1,4,5,11–21]. The former initially discretizes the PDE system into an approximate finite-dimensional ordinary differential equation (ODE) system, which is then used for optimal control design purposes. This approach is mostly used for parabolic PDE systems, because their dominant dynamic behaviors are usually characterized by a finite (typically small) number of degrees of freedom. In contrast to parabolic PDE systems, hyperbolic PDE systems involve a spatial differential operator whose eigenvalues cluster along vertical or nearly vertical asymptotes in the complex plane and thus cannot be accurately represented by a finite number of modes. This feature prohibits the application of modal decomposition techniques to the hyperbolic PDE system to derive an approximate ODE model and suggests addressing the control problem on the basis of the infinite-dimensional model itself. Following this suggestion, some control approaches have been developed for hyperbolic PDE systems in the past decades, including the LQ optimal control method via spectral factorization [13] and operator Riccati equation (ORE) [4,5,14], boundary control method [15–17], the sliding mode control method [18] and model predictive control method [19] on the basis of equivalent ODE realizations obtained by the method of characteristics, the nonlinear control method [20] through a combination of PDE theory and geometric control techniques, and the fuzzy control method based on directly the *T–S* fuzzy PDE model [21].

* Corresponding author.
E-mail address: whn@buaa.edu.cn (H.-N. Wu).

0959-1524/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jprocont.2012.04.011

However, most of the above approaches are model-based that require the full knowledge of the mathematical system models, which in most real cases are either unavailable or costly to obtain, due to the high-complexity of the industrial processes. The most prominent feature of such cases is the presence of vast volume of data accompanied by the lack of an effective process physical model that can support control. Moreover, the modeling and identification are also very difficult for PDE systems [22,23], even for ODE systems [24]. Then, these model-based methods encounter limits on the achievable performance, induced by conservatism introduced in the process modeling step. To overcome this limitation and to improve the performance beyond model-based control design, it is significant to develop control approaches for the cases that accurate modeling is impossible. This is a promising but challenging area for future control theory researches. Reinforcement learning (RL) [25–28] is a kind of machine learning method, which refers to an actor or agent that interacts with its environment and aims to learn the optimal actions, or control policies, by observing their responses from the environment. It affords a methodology for learning the feedback control actions online in real time based on system performance without necessarily knowing the system dynamics. This also overcomes computational complexity, such as the *curse of dimensionality* [29] associated with dynamic programming. In recent years, RL techniques have been used to the optimal control design of ODE systems [28,30–34]. For linear continuous ODE systems, policy iteration (PI) was employed to find the optimal control law online in [31,34]. This algorithm is theoretically equal to the well known iterative method developed in [35], where the algebraic Riccati equation (ARE) was approximated by a sequence of linear Lyapunov equations (LEs). The optimal control for nonlinear ODE systems with the PI algorithm was also studied in [32,34,36]. In [36], an offline PI method was developed for nonlinear systems with control constraints, while an online PI algorithm was given in [32]. Later, a so-called synchronous PI method [33] and a generalized PI version [37] were also suggested for the optimal control problem for nonlinear ODE systems. In addition, some important approximate dynamic programming (ADP) methods were proposed to solve the optimal tracking control problem of discrete-time nonlinear affine systems [38] and general nonlinear systems [39]. Nevertheless, to the best of our knowledge, the RL-based adaptive optimal control problem of linear hyperbolic PDE systems from the DTR framework has not been addressed yet, which motivates the present study.

In this paper, we try to synthesize an adaptive distributed optimal control law for a class of linear hyperbolic PDE systems by using the RL techniques and the DTR approach. To avoid directly solving the model-based space-dependent Riccati differential equation (SDRDE) for the LQ optimal control problem of linear hyperbolic PDE systems [14], a PI algorithm is proposed to learn the solution of the SDRDE online without requiring the knowledge of the internal system dynamics. The convergence of the PI algorithm is established by showing that the SDRDE can be successively approximated by a sequence of space-dependent Lyapunov differential equations (SDLDEs). For implementation purpose, we develop a least-square method to online estimate the solutions of the SDLDEs. Finally, the simulation study on the distributed control of a steam-jacketed tubular heat exchanger is given to show the effectiveness of the proposed design method.

The reminder of the paper is arranged as follows. The problem description and some preliminary results are given in Section 2. The PI algorithm is proposed and related issues are analyzed in Section 3. A least-square based implementation method for adaptive distributed optimal control is developed in Section 4. Simulation studies on a steam-jacketed tubular heat exchanger are provided in Section 5. Finally, a brief conclusion is drawn in Section 6.

Notations: $\mathbb{R}$, $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ are the set of real numbers, the n-dimensional Euclidean space and the set of all real $n\times m$ matrices, respectively. $\|\cdot\|$ and $\langle\cdot,\cdot\rangle_{\mathbb{R}^n}$ denote the Euclidean norm and inner product for vectors, respectively. Identity matrix, of appropriate dimensions, is denoted by $\boldsymbol{I}$. The superscript 'T' is used for the transpose of a vector or a matrix. For a symmetric matrix $\boldsymbol{M}$, $\boldsymbol{M}>(\geq,<,\leq)0$ means that it is positive definite (positive semi-definite, negative definite, negative semi-definite, respectively). The space-varying symmetric matrix function $\boldsymbol{M}(z), z\in[\underline{z},\bar{z}]$ is positive definite (positive semi-definite, negative definite, negative semi-definite, respectively), if $\boldsymbol{M}(z)>(\geq,<,\leq)0$ for each $z\in[\underline{z},\bar{z}]$. The symbol $*$ is used as an ellipsis in matrix expressions that are induced by symmetry, e.g., $[\boldsymbol{P}(z)\boldsymbol{A}(z)+*]\triangleq\boldsymbol{P}(z)\boldsymbol{A}(z)+\boldsymbol{A}^T(z)\boldsymbol{P}(z)$, $z\in[\underline{z},\bar{z}]$.

$\mathcal{H}^n\triangleq\mathcal{L}_2([\underline{z},\bar{z}];\mathbb{R}^n)$ is an infinite-dimensional Hilbert space of n-dimensional square integrable vector functions $\boldsymbol{\omega}(z)\in\mathcal{H}^n$, $z\in[\underline{z},\bar{z}]\subset\mathbb{R}$ equipped with the inner product and norm $\langle\boldsymbol{\omega}_1,\boldsymbol{\omega}_2\rangle=\int_{\underline{z}}^{\bar{z}}\langle\boldsymbol{\omega}_1(z),\boldsymbol{\omega}_2(z)\rangle_{\mathbb{R}^n}\,dz$ and $\|\boldsymbol{\omega}_1\|_2=\langle\boldsymbol{\omega}_1,\boldsymbol{\omega}_1\rangle^{1/2}$, where $\boldsymbol{\omega}_1$ and $\boldsymbol{\omega}_2$ are any two elements of $\mathcal{H}^n$.

2. Preliminaries and problem description

We consider the following linear first-order hyperbolic PDE systems in one spatial dimension with a state-space description of the form:

$$\frac{\partial\boldsymbol{y}(z,t)}{\partial t}=\boldsymbol{A}_1\frac{\partial\boldsymbol{y}(z,t)}{\partial z}+\boldsymbol{A}_2(z)\boldsymbol{y}(z,t)+\boldsymbol{B}(z)\boldsymbol{u}(z,t)\tag{1}$$

subject to the boundary condition

$$\boldsymbol{y}(\underline{z},t)=0\tag{2}$$

and the initial condition

$$\boldsymbol{y}(z,0)=\boldsymbol{y}_0(z)\tag{3}$$

where $z\in[\underline{z},\bar{z}]\subset\mathbb{R}$ and $t\in[0,\infty)$ denote spatial position and time, respectively, $\boldsymbol{y}(z,t)=[\,y_1(z,t)\ \cdots\ y_n(z,t)\,]^T\in\mathbb{R}^n$ is the state, $\boldsymbol{y}_0(z)\in\mathbb{R}^n$ is the initial state, and $\boldsymbol{u}(z,t)=[\,u_1(z,t)\ \cdots\ u_m(z,t)\,]^T\in\mathbb{R}^m$ is the manipulated control input. $\boldsymbol{A}_1$ is a real known diagonal $n\times n$ matrix, $\boldsymbol{A}_2(z)$ and $\boldsymbol{B}(z)$ are real continuous space-varying matrix functions of appropriate dimensions. The PDE system (1)–(3) is referred to as an *unforced* PDE system when $\boldsymbol{u}(z,t)\equiv 0$, and the corresponding system dynamics, i.e., $\boldsymbol{A}_1(\partial\boldsymbol{y}(z,t)/\partial z)+\boldsymbol{A}_2(z)\boldsymbol{y}(z,t)$, is referred to as *internal system dynamics* that is *unknown* in this paper.

Notice that the matrix $\boldsymbol{A}_1$ in (1) is assumed to be diagonal. This is representative to some extent and is widely used in the literature [13,14,40,41]. Typical examples satisfying this assumption include plug flow reactors, fixed-bed reactors, and steam-jacketed tubular heat exchangers where the elements of $\boldsymbol{A}_1$ are the fluid velocities [1,42,43].

Remark 1. Due to the high-complexity of real industrial processes, the accurate modeling and identification of which is impossible or costly to conduct, and the most prominent feature is the presence of vast volume of data accompanied by the lack of an effective process physical model that can support control. Thus, the statement that the internal system dynamics is unknown or not completely known is true for the vast majority of processes of practical interest. The development of control approaches for such a case is significant, but also challenging.

We consider the following linear distributed state feedback control law:

$$\boldsymbol{u}(z,t)=\boldsymbol{K}(z)\boldsymbol{y}(z,t)\tag{4}$$

where $\boldsymbol{K}(z)$ is the control gain to be determined, which is a real continuous space-varying $m\times n$ matrix function defined on the interval $[\underline{z},\bar{z}]$.

The substitution of the control law (4) into (1) results in the following closed-loop PDE system:

$$\frac{\partial \boldsymbol{y}(z,t)}{\partial t}=\boldsymbol{A}_1\frac{\partial \boldsymbol{y}(z,t)}{\partial z}+\bar{\boldsymbol{A}}_2(z)\boldsymbol{y}(z,t) \tag{5}$$

where

$$\bar{\boldsymbol{A}}_2(z)\triangleq \boldsymbol{A}_2(z)+\boldsymbol{B}(z)\boldsymbol{K}(z). \tag{6}$$

For convenience, we denote $\boldsymbol{y}(\cdot,t)\triangleq\boldsymbol{y}(z,t)$, $z\in[\underline{z},\bar{z}]$, and $\boldsymbol{u}(\cdot,t)\triangleq\boldsymbol{u}(z,t)$, $z\in[\underline{z},\bar{z}]$ and denote $\boldsymbol{M}(\cdot)\triangleq\boldsymbol{M}(z)$, $z\in[\underline{z},\bar{z}]$ and $\dot{\boldsymbol{M}}(\cdot)\triangleq d\boldsymbol{M}(z)/dz$, $z\in[\underline{z},\bar{z}]$ for some space-varying matrix function $\boldsymbol{M}(z)$, $z\in[\underline{z},\bar{z}]$. Define the following infinite-horizon LQ cost functional:

$$V_{\boldsymbol{u}}(\boldsymbol{y}(\cdot,t))\triangleq\int_t^{+\infty}(\langle\boldsymbol{y}(\cdot,\tau),\boldsymbol{Q}(\cdot)\boldsymbol{y}(\cdot,\tau)\rangle+\langle\boldsymbol{u}(\cdot,\tau),\boldsymbol{R}(\cdot)\boldsymbol{u}(\cdot,\tau)\rangle)\,d\tau \tag{7}$$

where $\boldsymbol{Q}(z)>0$ and $\boldsymbol{R}(z)>0$.

The optimal control problem of PDE system (1)–(3) here is to find a control law of the form (4) such that the LQ cost functional $V_{\boldsymbol{u}}(\boldsymbol{y}_0(\cdot))$ is minimized, i.e.,

$$\boldsymbol{u}(z,t)=\boldsymbol{u}^*(z,t)\triangleq\arg\min_{\boldsymbol{u}}V_{\boldsymbol{u}}(\boldsymbol{y}_0(\cdot)) \tag{8}$$

We now give the following preliminary definition and result.

Definition 1. The unforced PDE system (1)–(3) is said to be *exponentially stable*, if there exist real constants $\rho,\sigma>0$ such that the following expression holds:

$$\left\|\boldsymbol{y}(\cdot,t)\right\|_2^2\le\sigma e^{-\rho t}\left\|\boldsymbol{y}_0(\cdot)\right\|_2^2,\ \forall\ \ t\ge0.$$

Lemma 1. [14]. *Consider the system (1)–(3) with the state feedback control law (4) and cost functional $V_{\boldsymbol{u}}(\boldsymbol{y}_0(\cdot))$. Suppose that $\boldsymbol{A}_1<0$. Let $\boldsymbol{P}^*(z)>0$ be a diagonal real continuous space-varying $n\times n$ matrix function defined on interval $[\underline{z},\bar{z})$, and $\boldsymbol{P}^*(\bar{z})=0$. If $\boldsymbol{P}^*(z)$ satisfies the SDRDE*

$$\boldsymbol{A}_1\frac{d\boldsymbol{P}^*(z)}{dz}=[\boldsymbol{P}^*(z)\boldsymbol{A}_2(z)+*]+\boldsymbol{Q}(z)-\boldsymbol{P}^*(z)\boldsymbol{B}(z)\boldsymbol{R}^{-1}(z)\boldsymbol{B}^T(z)\boldsymbol{P}^*(z),$$
$$\boldsymbol{P}^*(\bar{z})=0,\ z\in[\underline{z},\bar{z}] \tag{9}$$

then $\boldsymbol{P}^*(z)$ is the unique solution of the SDRDE (9) and the solution of the optimal control problem (8) is

$$\boldsymbol{u}^*(z,t)=-\boldsymbol{R}^{-1}(z)\boldsymbol{B}^T(z)\boldsymbol{P}^*(z)\boldsymbol{y}(z,t) \tag{10}$$

and the minimum cost is

$$V_{\boldsymbol{u}^*}(\boldsymbol{y}_0(\cdot))=\langle\boldsymbol{y}_0(\cdot),\boldsymbol{P}^*(\cdot)\boldsymbol{y}_0(\cdot)\rangle. \tag{11}$$

Remark 2. It is mentioned in [14] that the solution $\boldsymbol{P}^*(z)$ of SDRDE (9) is diagonal. The existence of such a $\boldsymbol{P}^*(z)$ can provide an optimal control law (10) for the PDE system (1)–(3). The reason for the need of a diagonal matrix function $\boldsymbol{P}^*(z)$ results from the condition $\boldsymbol{P}^*(z)\boldsymbol{A}_1=\boldsymbol{A}_1^T\boldsymbol{P}^*(z)$ required for the derivation of the SDRDE (9) [14]. This may lead to conservatism of the control design. However, for the weakly hyperbolic PDE system (i.e., $\boldsymbol{A}_1=a_1\boldsymbol{I}$, where a_1 is a constant) [43], the restriction of $\boldsymbol{P}^*(z)$ can be removed.

It is observed from Lemma 1 that the distributed optimal control problem of the PDE system (1)–(3) hinges on the solution of the SDRDE (9). A direct solution approach was presented in [14], but it is a model-based method which relies on the full knowledge of the PDE system model. Thus, it can not be used for the case that internal dynamics of the PDE system is unknown. Furthermore, the model-based methods encounter limits on the achievable performance, induced by conservatism introduced in the process modeling step. To overcome this limitation and improve the performance beyond model-based control design, we try to develop an adaptive distributed optimal control approach when the PDE system model is not completely known.

3. Adaptive distributed optimal control design

In this section, we propose a PI algorithm for the adaptive optimal control design without requiring the knowledge of internal dynamics of the PDE system (1)–(3). The PI algorithm can learn the solution of the SDRDE by measuring system states online. Learning control policy from online measured data is striking because it requires no prior knowledge of a process model (i.e., the internal system dynamics), moreover, the estimations and assumptions introduced in the process modeling step are omitted.

We first give procedure of the PI algorithm as follows:

Algorithm 1. *Step 1:* Give an initial diagonal matrix function $\boldsymbol{P}^{(0)}(z)>0$, $z\in[\underline{z},\bar{z})$, $\boldsymbol{P}^{(0)}(\bar{z})=0$, such that $\boldsymbol{u}^{(0)}(z,t)=-\boldsymbol{R}^{-1}(z)\boldsymbol{B}^T(z)\boldsymbol{P}^{(0)}(z)\boldsymbol{y}(z,t)$ is an exponentially stabilizing control law. Let $i=0$.

Step 2: Solve the following equation for $\boldsymbol{P}^{(i+1)}(z)$

$$\langle\boldsymbol{y}(\cdot,t),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle=\langle\boldsymbol{y}(\cdot,t+\Delta t),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t+\Delta t)\rangle+\int_t^{t+\Delta t}\Xi^{(i)}(\tau)\,d\tau \tag{12}$$

where $\Xi^{(i)}(\tau)\triangleq\langle\boldsymbol{y}(\cdot,\tau),\boldsymbol{Q}(\cdot)\boldsymbol{y}(\cdot,\tau)\rangle+\langle\boldsymbol{u}^{(i)}(\cdot,\tau),\boldsymbol{R}(\cdot)\boldsymbol{u}^{(i)}(\cdot,\tau)\rangle$, $\boldsymbol{P}^{(i+1)}(z)>0$ is a diagonal real continuous space-varying $n\times n$ matrix function defined on interval $[\underline{z},\bar{z})$, and $\boldsymbol{P}^{(i+1)}(\bar{z})=0$.

Step 3: Update $\boldsymbol{u}^{(i+1)}(z,t)$ by

$$\boldsymbol{u}^{(i+1)}(z,t)=\boldsymbol{K}^{(i+1)}(z)\boldsymbol{y}(z,t) \tag{13}$$

with

$$\boldsymbol{K}^{(i+1)}(z)=-\boldsymbol{R}^{-1}(z)\boldsymbol{B}^T(z)\boldsymbol{P}^{(i+1)}(z). \tag{14}$$

Step 4: Set $i=i+1$. If $\bar{\sigma}\left(\int_{\underline{z}}^{\bar{z}}(\boldsymbol{P}^{(i)}(z)-\boldsymbol{P}^{(i-1)}(z))\,dz\right)\le\zeta$ (ζ is a small positive real number, and $\bar{\sigma}(\cdot)$ denotes the maximum singular value of a matrix), stop iteration, else, go to Step 2 and continue.

Remark 3. The PI algorithm involves two basic operations, policy evaluation (Step 2) and policy improvement (Step 3). Policy evaluation is used for computing the cost for a control policy, and policy improvement is used for finding a better control policy with the corresponding cost. The two operations perform alternately to approach the optimal control policy and minimum (optimal) cost. Notice that it does not require the knowledge of the internal dynamics of the PDE system (1)–(3), whose information is embedded in the online measurement of the system states $\boldsymbol{y}(z,t)$ and $\boldsymbol{y}(z,t+\Delta t)$, and evaluation of the cost $\int_t^{t+\Delta t}\Xi^{(i)}(\tau)\,d\tau$. That is to say, the lack of knowledge about the internal system dynamics does not have any impact on the PI algorithm to learn the solution of the SDRDE. Thus, the resulting adaptive optimal control policy will not be affected by any errors between the dynamics of a model of the system and the dynamics of the real process.

Remark 4. It is noted that the information of $\boldsymbol{B}(z)$ of PDE system (1)–(3) is needed in the PI algorithm. This is reasonable for real industrial processes, because the spatial positions for distributing actuators are usually given beforehand.

Next, we establish the convergence of the PI algorithm. Let us first consider a diagonal real continuous space-varying $n\times n$ matrix function $\boldsymbol{P}(z)>0$ defined on interval $[\underline{z},\bar{z})$, and $\boldsymbol{P}(\bar{z})=0$. Differentiating $\langle\boldsymbol{y}(\cdot,t),\boldsymbol{P}(\cdot)\boldsymbol{y}(\cdot,t)\rangle$ with respect to time t along the state trajectories of the closed-loop system (5), we get

$$\frac{d}{dt}\langle \boldsymbol{y}(\cdot,t),\boldsymbol{P}(\cdot)\boldsymbol{y}(\cdot,t)\rangle=\left\langle \frac{\partial \boldsymbol{y}(\cdot,t)}{\partial t},\boldsymbol{P}(\cdot)\boldsymbol{y}(\cdot,t)\right\rangle$$
$$+\left\langle \boldsymbol{y}(\cdot,t),\boldsymbol{P}(\cdot)\frac{\partial \boldsymbol{y}(\cdot,t)}{\partial t}\right\rangle=\int_{\underline{z}}^{\bar{z}}\left(\boldsymbol{A}_1\frac{\partial \boldsymbol{y}(z,t)}{\partial z}+\bar{\boldsymbol{A}}_2(z)\boldsymbol{y}(z,t)\right)^T$$
$$\times\boldsymbol{P}(z)\boldsymbol{y}(z,t)\,dz+\int_{\underline{z}}^{\bar{z}}\boldsymbol{y}^T(z,t)\boldsymbol{P}(z)\left(\boldsymbol{A}_1\frac{\partial \boldsymbol{y}(z,t)}{\partial z}+\bar{\boldsymbol{A}}_2(z)\boldsymbol{y}(z,t)\right)dz$$
$$=\int_{\underline{z}}^{\bar{z}}\left(\frac{\partial \boldsymbol{y}(z,t)}{\partial z}\right)^T\boldsymbol{A}_1^T\boldsymbol{P}(z)\boldsymbol{y}(z,t)\,dz+\int_{\underline{z}}^{\bar{z}}\boldsymbol{y}^T(z,t)\boldsymbol{P}(z)\boldsymbol{A}_1\frac{\partial \boldsymbol{y}(z,t)}{\partial z}dz$$
$$+\langle\bar{\boldsymbol{A}}_2(\cdot)\boldsymbol{y}(\cdot,t),\boldsymbol{P}(\cdot)\boldsymbol{y}(\cdot,t)\rangle+\langle\boldsymbol{y}(\cdot,t),\boldsymbol{P}(\cdot)\bar{\boldsymbol{A}}_2(\cdot)\boldsymbol{y}(\cdot,t)\rangle \qquad (15)$$

Applying integration by parts, we find that

$$\int_{\underline{z}}^{\bar{z}}\boldsymbol{y}^T(z,t)\boldsymbol{P}(z)\boldsymbol{A}_1\frac{\partial \boldsymbol{y}(z,t)}{\partial t}dz=\boldsymbol{y}^T(z,t)\boldsymbol{P}(z)\boldsymbol{A}_1\boldsymbol{y}(z,t)\Big|_{z=\underline{z}}^{z=\bar{z}}$$
$$-\int_{\underline{z}}^{\bar{z}}\frac{\partial \boldsymbol{y}^T(z,t)}{\partial z}\boldsymbol{P}(z)\boldsymbol{A}_1\boldsymbol{y}(z,t)dz-\int_{\underline{z}}^{\bar{z}}\boldsymbol{y}^T(z,t)\frac{d\boldsymbol{P}(z)}{dz}\boldsymbol{A}_1\boldsymbol{y}(z,t)dz. \qquad (16)$$

Considering the boundary condition (2) and $\boldsymbol{P}(\bar{z})=0$, we have

$$\boldsymbol{y}^T(z,t)\boldsymbol{P}(z)\boldsymbol{A}_1\boldsymbol{y}(z,t)\Big|_{z=\underline{z}}^{z=\bar{z}}=0. \qquad (17)$$

Since both $\boldsymbol{P}(z)$ and $\boldsymbol{A}_1$ are diagonal matrices, the equality $\boldsymbol{P}(z)\boldsymbol{A}_1=\boldsymbol{A}_1^T\boldsymbol{P}(z)$ holds clearly. Then, it follows from (15) to (17) that

$$\frac{d}{dt}\langle \boldsymbol{y}(\cdot,t),\boldsymbol{P}(\cdot)\boldsymbol{y}(\cdot,t)\rangle=-\int_{\underline{z}}^{\bar{z}}\boldsymbol{y}^T(z,t)\boldsymbol{A}_1\frac{d\boldsymbol{P}(z)}{dz}\boldsymbol{y}(z,t)dz$$
$$+\langle\bar{\boldsymbol{A}}_2(\cdot)\boldsymbol{y}(\cdot,t),\boldsymbol{P}(\cdot)\boldsymbol{y}(\cdot,t)\rangle+\langle\boldsymbol{y}(\cdot,t),\boldsymbol{P}(\cdot)\bar{\boldsymbol{A}}_2(\cdot)\boldsymbol{y}(\cdot,t)\rangle$$
$$=-\langle\boldsymbol{y}(\cdot,t),\boldsymbol{A}_1\dot{P}(\cdot)\boldsymbol{y}(\cdot,t)\rangle+\langle\boldsymbol{y}(\cdot,t),\bar{\boldsymbol{A}}_2^T(\cdot)\boldsymbol{P}(\cdot)\boldsymbol{y}(\cdot,t)\rangle$$
$$+\langle\boldsymbol{y}(\cdot,t),\boldsymbol{P}(\cdot)\bar{\boldsymbol{A}}_2(\cdot)\boldsymbol{y}(\cdot,t)\rangle=\langle\boldsymbol{y}(\cdot,t),(-\boldsymbol{A}_1\dot{\boldsymbol{P}}(\cdot)+[\boldsymbol{P}(\cdot)\bar{\boldsymbol{A}}_2(\cdot)$$
$$+*])\boldsymbol{y}(\cdot,t)\rangle. \qquad (18)$$

Therefore, we have the following result:

Theorem 1. *Let $\boldsymbol{P}^{(i+1)}(z)>0$ be a diagonal real continuous space-varying $n\times n$ matrix function defined on interval $[\underline{z},\bar{z})$, and $\boldsymbol{P}^{(i+1)}(\bar{z})=0$, $i=0,1,2,\ldots$. Then, $\boldsymbol{P}^{(i+1)}(z)$ is a solution of Eq. (12) together with (13), if and only if $\boldsymbol{P}^{(i+1)}(z)$ is a solution of the SDLDE*

$$\boldsymbol{A}_1\frac{d\boldsymbol{P}^{(i+1)}(z)}{dz}=[\boldsymbol{P}^{(i+1)}(z)\bar{\boldsymbol{A}}_2^{(i)}(z)+*]+\boldsymbol{Q}(z)+(\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i)}(z) \qquad (19)$$

where

$$\bar{\boldsymbol{A}}_2^{(i)}(z)=\boldsymbol{A}_2(z)+\boldsymbol{B}(z)\boldsymbol{K}^{(i)}(z) \qquad (20)$$

and $\boldsymbol{K}^{(i)}(z)$ is obtained by (14).

Proof. Applying the control law $\boldsymbol{u}^{(i)}(z,t)=\boldsymbol{K}^{(i)}(z)\boldsymbol{y}(z,t)$, $i=0,1,2,\ldots$ to the PDE system (1)–(3), the state of the closed-loop PDE system satisfies

$$\frac{\partial \boldsymbol{y}(z,t)}{\partial t}=\boldsymbol{A}_1\frac{\partial \boldsymbol{y}(z,t)}{\partial z}+\boldsymbol{A}_2(z)\boldsymbol{y}(z,t)+\boldsymbol{B}(z)\boldsymbol{u}^{(i)}(z,t)$$
$$=\boldsymbol{A}_1\frac{\partial y(z,t)}{\partial z}+\bar{\boldsymbol{A}}_2^{(i)}(z)\boldsymbol{y}(z,t). \qquad (21)$$

From (18) and (21), we have

$$\frac{d}{dt}\langle \boldsymbol{y}(\cdot,t),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle=\langle\boldsymbol{y}(\cdot,t),(-\boldsymbol{A}_1\dot{\boldsymbol{P}}^{(i+1)}(\cdot)$$
$$+[\boldsymbol{P}^{(i+1)}(\cdot)\bar{\boldsymbol{A}}_2^{(i)}(\cdot)+*])\boldsymbol{y}(\cdot,t)\rangle.$$

Integrating the above equation from t to $t+\Delta t$ yields

$$\langle\boldsymbol{y}(\cdot,t+\Delta t),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t+\Delta t)\rangle-\langle\boldsymbol{y}(\cdot,t),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle$$
$$=\int_t^{t+\Delta t}\langle\boldsymbol{y}(\cdot,\tau),(-\boldsymbol{A}_1\dot{\boldsymbol{P}}^{(i+1)}(\cdot)+[\boldsymbol{P}^{(i+1)}(\cdot)\bar{\boldsymbol{A}}_2^{(i)}(\cdot)+*])\boldsymbol{y}(\cdot,\tau)\rangle\,d\tau. \qquad (22)$$

On the one hand, if $\boldsymbol{P}^{(i+1)}(z)$ is a solution of Eq. (19), we have

$$-\boldsymbol{A}_1\frac{d\boldsymbol{P}^{(i+1)}(z)}{dz}+[\boldsymbol{P}^{(i+1)}(z)\bar{\boldsymbol{A}}_2^{(i)}(z)+*]$$
$$=-(\boldsymbol{Q}(z)+(\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i)}(z)). \qquad (23)$$

The substitution of (23) into (22) and considering (13) yield

$$\langle\boldsymbol{y}(\cdot,t+\Delta t),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t+\Delta t)\rangle-\langle\boldsymbol{y}(\cdot,t),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle$$
$$=-\int_t^{t+\Delta t}\langle\boldsymbol{y}(\cdot,\tau),(\boldsymbol{Q}(\cdot)+(\boldsymbol{K}^{(i)}(\cdot))^T\boldsymbol{R}(\cdot)\boldsymbol{K}^{(i)}(\cdot))\boldsymbol{y}(\cdot,\tau)\rangle\,d\tau$$
$$=-\int_t^{t+\Delta t}\Xi^{(i)}(\tau)\,d\tau.$$

This implies that $\boldsymbol{P}^{(i+1)}(z)$ is also a solution of Eq. (12).

On the other hand, we prove that $\boldsymbol{P}^{(i+1)}(z)$ is the unique solution of (12) by contradiction. According to (12), we have

$$\frac{d}{dt}\langle\boldsymbol{y}(\cdot,t),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle=\lim_{\Delta t\to 0}\frac{1}{\Delta t}(\langle\boldsymbol{y}(\cdot,t+\Delta t),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t+\Delta t)\rangle$$
$$-\langle\boldsymbol{y}(\cdot,t),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle)=-\lim_{\Delta t\to 0}\frac{1}{\Delta t}\int_t^{t+\Delta t}\Xi^{(i)}(\tau)\,d\tau$$
$$=-\lim_{\Delta t\to 0}\frac{1}{\Delta t}\left(\int_0^{t+\Delta t}\Xi^{(i)}(\tau)\,d\tau-\int_0^{t}\Xi^{(i)}(\tau)\,d\tau\right)$$
$$=-\frac{d}{dt}\int_0^t\Xi^{(i)}(\tau)\,d\tau=-\Xi^{(i)}(t) \qquad (24)$$

Assume that $\tilde{\boldsymbol{P}}^{(i+1)}(z)$ is another solution of (12). Then similar to (24), we have

$$\frac{d}{dt}\langle\boldsymbol{y}(\cdot,t),\tilde{\boldsymbol{P}}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle=-\Xi^{(i)}(t). \qquad (25)$$

Subtracting (25) from (24), we get $(d/dt)\langle\boldsymbol{y}(\cdot,t),(\boldsymbol{P}^{(i+1)}(\cdot)-\tilde{\boldsymbol{P}}^{(i+1)}(\cdot))\boldsymbol{y}(\cdot,t)\rangle=0$ for $\forall\boldsymbol{y}(\cdot,t)$. Then, $\langle\boldsymbol{y}(\cdot,t),(\boldsymbol{P}^{(i+1)}(\cdot)-\tilde{\boldsymbol{P}}^{(i+1)}(\cdot))\boldsymbol{y}(\cdot,t)\rangle=c$, where c is a real constant, $c=0$ when $\boldsymbol{y}(\cdot,t)=0$. Thus, $\langle\boldsymbol{y}(\cdot,t),(\boldsymbol{P}^{(i+1)}(\cdot)-\tilde{\boldsymbol{P}}^{(i+1)}(\cdot))\boldsymbol{y}(\cdot,t)\rangle=0$, for $\forall\boldsymbol{y}(\cdot,t)$. This means that $\tilde{\boldsymbol{P}}^{(i+1)}(z)=\boldsymbol{P}^{(i+1)}(z)$ for each $z\in[\underline{z},\bar{z}]$. The proof is completed. ■

Remark 5. Theorem 1 indicates that, given an initial exponentially stabilizing control policy $\boldsymbol{u}^{(0)}(z,t)=\boldsymbol{K}^{(0)}(z)\boldsymbol{y}(z,t)$, the iteration (12) together with (13) is equivalent to the iteration (19) together with (14). Therefore, the convergence of PI algorithm can be established by proving that the iteration (19) together with (14) is convergent.

Namely, we prove that the solution of SDLDE (19) converges to the solution of SDRDE (9) when i goes to infinity.

Lemma 2. Let $\boldsymbol{u}^{(0)}(z,t)=\boldsymbol{K}^{(0)}(z)\boldsymbol{y}(z,t)$ be an exponentially stabilizing control law. If (13) together with (12) is used to update control policy, then, the new control policies are also exponentially stabilizing, i.e., the closed-loop PDE system (5) with $\boldsymbol{u}^{(i)}(z,t)=\boldsymbol{K}^{(i)}(z)\boldsymbol{y}(z,t)$, $i=1,2,\ldots$, is exponentially stable.

Proof. With the control law $\boldsymbol{u}^{(i)}(z,t)=\boldsymbol{K}^{(i)}(z)\boldsymbol{y}(z,t)$, $i=1,2,\ldots$, the state of the closed-loop PDE system is the solution of Eq. (21). We take

$$W(\boldsymbol{y}(\cdot,t))=\langle \boldsymbol{y}(\cdot,t),\boldsymbol{P}^{(i)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle$$

as a Lyapunov function candidate. According to (18), differentiating $W(\boldsymbol{y}(\cdot,t))$ with respect to time t along the solution of (21) yields

$$\frac{dW(\boldsymbol{y}(\cdot,t))}{dt}=\frac{d}{dt}\langle \boldsymbol{y}(\cdot,t),\boldsymbol{P}^{(i)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle=\langle \boldsymbol{y}(\cdot,t),(-\boldsymbol{A}_1\dot{\boldsymbol{P}}^{(i)}(\cdot)$$
$$+[\boldsymbol{P}^{(i)}(\cdot)\bar{\boldsymbol{A}}_2^{(i)}(\cdot)+*])\boldsymbol{y}(\cdot,t)\rangle \tag{26}$$

From (20), we have

$$\bar{\boldsymbol{A}}_2^{(i)}(z)=\boldsymbol{A}_2(z)+\boldsymbol{B}(z)\boldsymbol{K}^{(i)}(z)=\boldsymbol{A}_2(z)+\boldsymbol{B}(z)\boldsymbol{K}^{(i-1)}(z)+\boldsymbol{B}(z)(\boldsymbol{K}^{(i)}(z)$$
$$-\boldsymbol{K}^{(i-1)}(z))=\bar{\boldsymbol{A}}_2^{(i-1)}(z)+\boldsymbol{B}(z)(\boldsymbol{K}^{(i)}(z)-\boldsymbol{K}^{(i-1)}(z))$$

then,

$$-\boldsymbol{A}_1\frac{d\boldsymbol{P}^{(i)}(z)}{dz}+[\boldsymbol{P}^{(i)}(z)\bar{\boldsymbol{A}}_2^{(i)}(z)+*]=-\boldsymbol{A}_1\frac{d\boldsymbol{P}^{(i)}(z)}{dz}$$
$$+[\boldsymbol{P}^{(i)}(z)\bar{\boldsymbol{A}}_2^{(i-1)}(z)+*]+\left[\boldsymbol{P}^{(i)}(z)\boldsymbol{B}(z)(\boldsymbol{K}^{(i)}(z)-\boldsymbol{K}^{(i-1)}(z))+*\right]$$
$$=-\boldsymbol{A}_1\frac{d\boldsymbol{P}^{(i)}(z)}{dz}+[\boldsymbol{P}^{(i)}(z)\bar{\boldsymbol{A}}_2^{(i-1)}(z)+*]-(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)$$
$$\times(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z))-(\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i)}(z)$$
$$+(\boldsymbol{K}^{(i-1)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i-1)}(z). \tag{27}$$

Theorem 1 shows that the Eq. (12) together with (13) is equivalent to Eq. (19) together with (14). Then, it follows from (19) and (27) that

$$-\boldsymbol{A}_1\frac{d\boldsymbol{P}^{(i)}(z)}{dz}+\left[\boldsymbol{P}^{(i)}(z)\bar{\boldsymbol{A}}_2^{(i)}(z)+*\right]=-\left(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z)\right)^T\boldsymbol{R}(z)$$
$$\left(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z)\right)-\boldsymbol{Q}(z)-\left(\boldsymbol{K}^{(i)}(z)\right)^T\boldsymbol{R}(z)\boldsymbol{K}^{(i)}(z). \tag{28}$$

From (26) and (28), we have

$$\frac{dW(\boldsymbol{y}(\cdot,t))}{dt}$$
$$=-\left\langle \boldsymbol{y}(\cdot,t),\left(\boldsymbol{K}^{(i-1)}(\cdot)-\boldsymbol{K}^{(i)}(\cdot)\right)^T\boldsymbol{R}(\cdot)\left(\boldsymbol{K}^{(i-1)}(\cdot)-\boldsymbol{K}^{(i)}(\cdot)\right)\boldsymbol{y}(\cdot,t)\right\rangle$$
$$-\left\langle \boldsymbol{y}(\cdot,t),\left(\boldsymbol{Q}(\cdot)+\left(\boldsymbol{K}^{(i)}(\cdot)\right)^T\boldsymbol{R}(\cdot)\boldsymbol{K}^{(i)}(\cdot)\right)\boldsymbol{y}(\cdot,t)\right\rangle,$$

thus,

$$\frac{dW(\boldsymbol{y}(\cdot,t))}{dt}+\left\langle \boldsymbol{y}(\cdot,t),\boldsymbol{Q}(\cdot)\boldsymbol{y}(\cdot,t)\right\rangle<0 \text{ for } \forall \boldsymbol{y}(\cdot,t)\neq 0 \tag{29}$$

Considering $\boldsymbol{Q}(z)>0$ and $\boldsymbol{P}^{(i)}(z)\geq 0$, $z\in[\underline{z},\bar{z}]$, hence, there exist real constants $\alpha,\beta>0$ such that

$$W(\boldsymbol{y}(\cdot,t))\leq\alpha\left\langle \boldsymbol{y}(\cdot,t),\boldsymbol{y}(\cdot,t)\right\rangle \tag{30}$$

$$\left\langle \boldsymbol{y}(\cdot,t),\boldsymbol{Q}(\cdot)\boldsymbol{y}(\cdot,t)\right\rangle\geq\beta\left\langle \boldsymbol{y}(\cdot,t),\boldsymbol{y}(\cdot,t)\right\rangle \tag{31}$$

where $\alpha\triangleq\max_{j\in\{1,2,\cdots,n\}}\left\{\max_{z\in[\underline{z},\bar{z}]}\{\lambda_j(\boldsymbol{P}^{(i)}(z))\}\right\}$ and $\beta\triangleq\min_{j\in\{1,2,\cdots,n\}}\left\{\min_{z\in[\underline{z},\bar{z}]}\{\lambda_j(\boldsymbol{Q}(z))\}\right\}$, $\lambda_j(\cdot)$ represents the jth eigenvalue of a matrix. It follows from (30) and (31) that

$$\left\langle \boldsymbol{y}(\cdot,t),\boldsymbol{Q}(\cdot)\boldsymbol{y}(\cdot,t)\right\rangle\geq\beta\left\langle \boldsymbol{y}(\cdot,t),\boldsymbol{y}(\cdot,t)\right\rangle\geq\alpha^{-1}\beta W(\boldsymbol{y}(\cdot,t)). \tag{32}$$

From (29) and (32), we can get

$$\frac{dW(\boldsymbol{y}(\cdot,t))}{dt}+\alpha^{-1}\beta W(\boldsymbol{y}(\cdot,t))<0 \text{ for } \forall \boldsymbol{y}(\cdot,t)\neq 0. \tag{33}$$

Thus, by using Lemma 1 in [21], we can conclude that the closed-loop PDE system (5) is exponentially stable.■

Remark 6. Lemma 3 reveals that, given an initial exponentially stabilizing control law, all control policies (in the PI algorithm) obtained using iteration (12) and (13) are exponentially stabilizing.

Theorem 2. *Let $\boldsymbol{P}^*(z)$ be the solution of the SDRDE (9), $\boldsymbol{P}^{(0)}(z)>0$ be a diagonal real continuous space-varying $n\times n$ matrix function defined on interval $[\underline{z},\bar{z})$ and $\boldsymbol{P}^{(0)}(\bar{z})=0$, such that $\boldsymbol{u}^{(0)}(z,t)=-\boldsymbol{R}^{-1}(z)\boldsymbol{B}^T(z)\boldsymbol{P}^{(0)}(z)\boldsymbol{y}(z,t)$ is an exponentially stabilizing control law. If $\boldsymbol{P}^{(i+1)}(z)$, $i=0,1,2,\ldots$ are the solutions of SDLDE (19), then,*

(1) $\boldsymbol{P}^*(z)\leq\boldsymbol{P}^{(i+1)}(z)\leq\boldsymbol{P}^{(i)}(z)$, $i=1,2,\cdots$, for $\forall z\in[\underline{z},\bar{z}]$;
(2) $\boldsymbol{P}^{(i)}(z)$ converges uniformly to $\boldsymbol{P}^*(z)$ when $i\to\infty$.

Proof. (1) For $i=1,2,\ldots$, it follows from (20) that

$$\bar{\boldsymbol{A}}_2^{(i-1)}(z)=\boldsymbol{A}_2(z)+\boldsymbol{B}(z)\boldsymbol{K}^{(i-1)}(z)=\boldsymbol{A}_2(z)+\boldsymbol{B}(z)\boldsymbol{K}^{(i)}(z)$$
$$+\boldsymbol{B}(z)(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z))=\bar{\boldsymbol{A}}_2^{(i)}(z)+\boldsymbol{B}(z)(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z)). \tag{34}$$

From (19), we have

$$\boldsymbol{A}_1\frac{d\boldsymbol{P}^{(i)}(z)}{dz}=[\boldsymbol{P}^{(i)}(z)\bar{\boldsymbol{A}}_2^{(i-1)}(z)+*]+\boldsymbol{Q}(z)+(\boldsymbol{K}^{(i-1)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i-1)}(z). \tag{35}$$

Substituting (34) into (35) yields

$$\boldsymbol{A}_1\frac{d\boldsymbol{P}^{(i)}(z)}{dz}=[\boldsymbol{P}^{(i)}(z)(\bar{\boldsymbol{A}}_2^{(i)}(z)+\boldsymbol{B}(z)(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z)))+*]+\boldsymbol{Q}(z)$$
$$+(\boldsymbol{K}^{(i-1)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i-1)}(z)=[\boldsymbol{P}^{(i)}(z)\bar{\boldsymbol{A}}_2^{(i)}(z)+*]+\boldsymbol{Q}(z)$$
$$+(\boldsymbol{K}^{(i-1)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i-1)}(z)+[\boldsymbol{P}^{(i)}(z)\boldsymbol{B}(z)(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z))+*]. \tag{36}$$

Let $\Delta\boldsymbol{P}^{(i)}(z)\triangleq\boldsymbol{P}^{(i)}(z)-\boldsymbol{P}^{(i+1)}(z)$, where $\boldsymbol{P}^{(i+1)}(z)$ is the solution of (19). Subtracting (19) from (36) yields

$$\boldsymbol{A}_1\frac{d\Delta\boldsymbol{P}^{(i)}(z)}{dz}=[\Delta\boldsymbol{P}^{(i)}(z)\bar{\boldsymbol{A}}_2^{(i)}(z)+*]+(\boldsymbol{K}^{(i-1)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i-1)}(z)$$
$$-(\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i)}(z)+[\boldsymbol{P}^{(i)}(z)\boldsymbol{B}(z)(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z))+*]. \tag{37}$$

Adding and subtracting the term

$$[(\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z))+*]=(\boldsymbol{K}^{(i-1)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i)}$$
$$+(\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i-1)}(z)-2(\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i)}(z)$$

from the right side of (37) and rearranging terms yield

$$\begin{aligned}
\boldsymbol{A}_1\frac{d\Delta\boldsymbol{P}^{(i)}(z)}{dz} &= [\Delta\boldsymbol{P}^{(i)}(z)\bar{\boldsymbol{A}}_2^{(i)}(z)+*]\\
&+[\boldsymbol{P}^{(i)}(z)\boldsymbol{B}(z)(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z))+*]+(\boldsymbol{K}^{(i-1)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i-1)}(z)\\
&-(\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i)}(z)+[(\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z))+*]\\
&-(\boldsymbol{K}^{(i-1)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i)}-(\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i-1)}(z)\\
&+2(\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)\boldsymbol{K}^{(i)}(z)=[\Delta\boldsymbol{P}^{(i)}(z)\bar{\boldsymbol{A}}_2^{(i)}(z)+*]\\
&+(\boldsymbol{K}^{(i-1)}-\boldsymbol{K}^{(i)})^T\boldsymbol{R}(z)(\boldsymbol{K}^{(i-1)}-\boldsymbol{K}^{(i)})\\
&+[(\boldsymbol{R}^{-1}(z)\boldsymbol{B}^T(z)\boldsymbol{P}^{(i)}(z)+\boldsymbol{K}^{(i)}(z))^T\boldsymbol{R}(z)(\boldsymbol{K}^{(i-1)}(z)-\boldsymbol{K}^{(i)}(z))+*]
\end{aligned} \tag{38}$$

Using (14), we obtain

$$\boldsymbol{A}_1\frac{d\Delta\boldsymbol{P}^{(i)}(z)}{dz} = [\Delta\boldsymbol{P}^{(i)}(z)\bar{\boldsymbol{A}}_2^{(i)}(z) +*]+(\boldsymbol{K}^{(i-1)}-\boldsymbol{K}^{(i)})^T\boldsymbol{R}(z)(\boldsymbol{K}^{(i-1)}-\boldsymbol{K}^{(i)}),$$

i.e.,

$$\begin{aligned}
&-\boldsymbol{A}_1\frac{d\Delta\boldsymbol{P}^{(i)}(z)}{dz}+[\Delta\boldsymbol{P}^{(i)}(z)\bar{\boldsymbol{A}}_2^{(i)}(z)+*]\\
&=-(\boldsymbol{K}^{(i-1)}-\boldsymbol{K}^{(i)})^T\boldsymbol{R}(z)(\boldsymbol{K}^{(i-1)}-\boldsymbol{K}^{(i)}).
\end{aligned} \tag{39}$$

With the control law $\boldsymbol{u}^{(i)}(z,t)=\boldsymbol{K}^{(i)}(z)\boldsymbol{y}(z,t)$ and using (18), we can obtain that the state of the resulting closed-loop PDE system satisfies

$$\begin{aligned}
&\frac{d}{dt}\langle\boldsymbol{y}(\cdot,t),\Delta\boldsymbol{P}^{(i)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle=\langle\boldsymbol{y}(\cdot,t),(-\boldsymbol{A}_1\Delta\dot{\boldsymbol{P}}^{(i)}(\cdot)\\
&+[\Delta\boldsymbol{P}^{(i)}(\cdot)\bar{\boldsymbol{A}}_2^{(i)}(\cdot)+*])\boldsymbol{y}(\cdot,t)\rangle.
\end{aligned} \tag{40}$$

From (39) and (40), we have

$$\begin{aligned}
&\frac{d}{dt}\langle\boldsymbol{y}(\cdot,t),\Delta\boldsymbol{P}^{(i)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle\\
&=-\langle\boldsymbol{y}(\cdot,t),(\boldsymbol{K}^{(i-1)}-\boldsymbol{K}^{(i)})^T\boldsymbol{R}(\cdot)(\boldsymbol{K}^{(i-1)}-\boldsymbol{K}^{(i)})\boldsymbol{y}(\cdot,t)\rangle\le 0
\end{aligned}$$

which means that $\langle\boldsymbol{y}(\cdot,t),\Delta\boldsymbol{P}^{(i)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle$ is a non-increasing function with respect to time t. Since $\boldsymbol{u}^{(i)}(z,t)=\boldsymbol{K}^{(i)}(z)\boldsymbol{y}(z,t)$ is an exponentially stabilizing control policy according to Lemma 2, then we have $\lim_{t\to\infty}\langle\boldsymbol{y}(\cdot,t),\Delta\boldsymbol{P}^{(i)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle=0$. It further implies that $\langle\boldsymbol{y}(\cdot,t),\Delta\boldsymbol{P}^{(i)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle\ge 0$, $t\ge 0$ for $\forall\boldsymbol{y}(\cdot,t)$. Thus, we get $\Delta\boldsymbol{P}^{(i)}(z)\ge 0$ for $\forall z\in[\underline{z},\bar{z}]$, i.e., $\boldsymbol{P}^{(i+1)}(z)\le\boldsymbol{P}^{(i)}(z)$, $i=1,2,\ldots$, $\forall z\in[\underline{z},\bar{z}]$.

Now, we prove $\boldsymbol{P}^{(i)}(z)\ge\boldsymbol{P}^*(z)$, $\forall z\in[\underline{z},\bar{z}]$, $i=1,2,\ldots$. With arbitrary state $\boldsymbol{y}(\cdot,t)$ and control law $\boldsymbol{u}^{(i)}(z,t)=\boldsymbol{K}^{(i)}(z)\boldsymbol{y}(z,t)$, $i=0,1,2,\ldots$, the cost functional (7) can be written as

$$\begin{aligned}
V_{u^{(i)}}(\boldsymbol{y}(\cdot,t))&=\int_t^{+\infty}\Xi^{(i)}(\tau)\,d\tau\\
&=\int_t^{+\infty}\langle\boldsymbol{y}(\cdot,\tau),(\boldsymbol{Q}(\cdot)+(\boldsymbol{K}^{(i)}(\cdot))^T\boldsymbol{R}(\cdot)\boldsymbol{K}^{(i)}(\cdot))\boldsymbol{y}(\cdot,\tau)\rangle\,d\tau.
\end{aligned} \tag{41}$$

According to (18), (19) and (41), we have

$$\begin{aligned}
V_{u^{(i)}}(\boldsymbol{y}(\cdot,t))&=-\int_t^{+\infty}\langle\boldsymbol{y}(\cdot,\tau),(-\boldsymbol{A}_1\dot{\boldsymbol{P}}^{(i+1)}(\cdot)\\
&+[\boldsymbol{P}^{(i+1)}(\cdot)\bar{\boldsymbol{A}}_2^{(i)}(\cdot)+*])\boldsymbol{y}(\cdot,\tau)\rangle\,d\tau\\
&=-\int_t^{+\infty}\frac{d}{d\tau}\langle\boldsymbol{y}(\cdot,\tau),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,\tau)\rangle\,d\tau\\
&=-\langle\boldsymbol{y}(\cdot,\tau),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,\tau)\rangle\Big|_{\tau=t}^{\tau=\infty}.
\end{aligned}$$

Because $\boldsymbol{u}^{(i)}(z,t)=\boldsymbol{K}^{(i)}(z)\boldsymbol{y}(z,t)$ is an exponentially stabilizing control policy, we obtain

$$V_{u^{(i)}}(\boldsymbol{y}(\cdot,t))=\langle\boldsymbol{y}(\cdot,t),\boldsymbol{P}^{(i+1)}(\cdot)\boldsymbol{y}(\cdot,t)\rangle. \tag{42}$$

Since $\boldsymbol{P}^*(z)$ is the solution of the SDRDE (8), the associated cost $V_{u^*}(\boldsymbol{y}(\cdot,t))=\langle\boldsymbol{y}(\cdot,t),\boldsymbol{P}^*(\cdot)\boldsymbol{y}(\cdot,t)\rangle$ is the minimum. Thus, we have $V_{u^{(i)}}(\boldsymbol{y}(\cdot,t))\ge V_{u^*}(\boldsymbol{y}(\cdot,t))$ for $\forall\boldsymbol{y}(\cdot,t)$, which means that $\boldsymbol{P}^{(i)}(z)\ge\boldsymbol{P}^*(z)$, $\forall z\in[\underline{z},\bar{z}]$, $i=1,2,\ldots$ This completes the proof of the first part of Theorem 2.

(2) To show that the sequence $\boldsymbol{P}^{(i)}(z)$ has a limit as $i\to\infty$, we choose an arbitrary $\boldsymbol{e}\in\boldsymbol{E}$ ($\boldsymbol{E}\subset\mathscr{H}^n$ is a set of orthogonal basis). Then, from the first part of Theorem 2, $\langle\boldsymbol{e},\boldsymbol{P}^{(i)}(z)\boldsymbol{e}\rangle$ is non-increasing as $i\to\infty$ and is uniformly bounded below by $\langle\boldsymbol{e},\boldsymbol{P}^*(z)\boldsymbol{e}\rangle$. Hence, $\lim_{i\to\infty}\langle\boldsymbol{e},\boldsymbol{P}^{(i)}(z)\boldsymbol{e}\rangle$ exists for all $\boldsymbol{e}\in\boldsymbol{E}$, since a bounded monotone sequence always has a limit, denoted as $\lim_{i\to\infty}\langle\boldsymbol{e},\boldsymbol{P}^{(i)}(z)\boldsymbol{e}\rangle\triangleq\langle\boldsymbol{e},\boldsymbol{P}^{(\infty)}(z)\boldsymbol{e}\rangle$. This means that the limit of $\boldsymbol{P}^{(i)}(z)$ exists for $\forall z\in[\underline{z},\bar{z}]$, and $\lim_{i\to\infty}\boldsymbol{P}^{(i)}(z)=\boldsymbol{P}^{(\infty)}(z)$.

Now, we will show that $\boldsymbol{P}^{(\infty)}(z)$ satisfies the SDRDE (9), i.e., $\boldsymbol{P}^{(\infty)}(z)=\boldsymbol{P}^*(z)$. It follows from (14) and (19) that $\boldsymbol{P}^{(i+1)}(z)$ satisfies the equation

$$\begin{aligned}
\boldsymbol{A}_1\frac{\partial\boldsymbol{P}^{(i+1)}(z)}{\partial z}&=[\boldsymbol{P}^{(i+1)}(z)\bar{\boldsymbol{A}}_2^{(i)}(z)+*]+\boldsymbol{Q}(z)\\
&+\boldsymbol{P}^{(i)}(z)\boldsymbol{B}(z)\boldsymbol{R}(z)\boldsymbol{B}^T(z)\boldsymbol{P}^{(i)}(z).
\end{aligned} \tag{43}$$

Integrating both sides of Eq. (43) from z to $\bar{z}$ and considering $\boldsymbol{P}^{(i+1)}(\bar{z})=0$, we get

$$\begin{aligned}
\boldsymbol{A}_1\boldsymbol{P}^{(i+1)}(z)&=-\int_z^{\bar{z}}\{[\boldsymbol{P}^{(i+1)}(\varsigma)\bar{\boldsymbol{A}}_2^{(i)}(\varsigma)+*]\\
&+\boldsymbol{Q}(\varsigma)+\boldsymbol{P}^{(i)}(\varsigma)\boldsymbol{B}(\varsigma)\boldsymbol{R}(\varsigma)\boldsymbol{B}^T(\varsigma)\boldsymbol{P}^{(i)}(\varsigma)\}\,\boldsymbol{d}\varsigma.
\end{aligned}$$

Taking the limit of both sides of the above expression and using the bounded convergence theorem [44], yield

$$\begin{aligned}
\boldsymbol{A}_1\boldsymbol{P}^{(\infty)}(z)&=-\int_z^{\bar{z}}\{[\boldsymbol{P}^{(\infty)}(\varsigma)\bar{\boldsymbol{A}}_2^{(\infty)}(\varsigma)+*]\\
&+\boldsymbol{Q}(\varsigma)+\boldsymbol{P}^{(\infty)}(\varsigma)\boldsymbol{B}(\varsigma)\boldsymbol{R}(\varsigma)\boldsymbol{B}^T(\varsigma)\boldsymbol{P}^{(\infty)}(\varsigma)\}\,\boldsymbol{d}\varsigma.
\end{aligned}$$

This implies that $\boldsymbol{P}^{(\infty)}(z)$ is continuous, then, differentiating yields

$$\begin{aligned}
\boldsymbol{A}_1\frac{\partial\boldsymbol{P}^{(\infty)}(z)}{\partial z}&=[\boldsymbol{P}^{(\infty)}(z)\bar{\boldsymbol{A}}_2^{(\infty)}(z)+*]\\
&+\boldsymbol{Q}(z)+\boldsymbol{P}^{(\infty)}(z)\boldsymbol{B}(z)\boldsymbol{R}(z)\boldsymbol{B}^T(z)\boldsymbol{P}^{(\infty)}(z).
\end{aligned}$$

This means that $\boldsymbol{P}^{(\infty)}(z)$ is a solution of the SDRDE (9). By uniqueness of the solution, $\boldsymbol{P}^{(\infty)}(z)=\boldsymbol{P}^*(z)$. The proof is complete. ■

Remark 7. Observe that the SDLDE plays a critical role for convergence analysis of the PI algorithm in this paper. (1) Theorem 1 shows that the PI algorithm is theoretically equivalent to an iterative procedure of a sequence of SDLDEs. (2) The solution of SDLDE gives a compact representation (42) of the cost of an arbitrary exponentially stabilizing control policy. (3) Theorem 2 implies that the solution of SDLDE allows us to improve the cost of the original control. By using the PI algorithm to iteratively solve a sequence of SDLDEs online, we can obtain the approximate solution of the SDRDE without the knowledge of the internal system dynamics. In fact, the SDLDE can be viewed as a generalization of the SDRDE for an arbitrary exponentially stabilizing control policy. The SDLDE is a linear differential equation, while the SDRDE is nonlinear. Thus, The SDLDE is theoretically easier to solve than the SDRDE.

Remark 8. It is worth mentioning that the proposed PI algorithm (i.e. Algorithm 1) may not be directly applied to linear parabolic PDE systems. The reason is that the optimal control problem of linear parabolic PDE systems requires solving another SDRDE which is completely different than (9). Thus, a new PI algorithm for linear parabolic PDE systems is needed, which is left for future investigation. Moreover, we notice that the proposed PI algorithm can be used for nonlinear hyperbolic PDE systems in the vicinity of the operating point. However, it cannot predict the "nonlocal" behavior far from the operating point and certainly not the "global" behavior throughout the state space. To overcome this difficulty, a new PI algorithm should be developed for nonlinear hyperbolic PDE systems. We also leave this issue for future study.

4. Implementation of adaptive distributed optimal control

Notice that in each iterative step of the PI algorithm, we need to solve Eq. (12). To this end, we derive a least-square approximation method on a set $\mathcal{D} \subset \mathcal{H}^n$ such that $\boldsymbol{y}(\cdot, t) \in \mathcal{D}$, to estimate the solution of Eq. (12). Denoting $\boldsymbol{P}^{(i)}(z) = diag\{p_1^{(i)}(z), \ldots, p_n^{(i)}(z)\}$, $i = 1, 2, \ldots$, then (12) can be rewritten as

$$\sum_{j=1}^{n} \int_{\underline{z}}^{\bar{z}} p_j^{(i+1)}(z)(y_j^2(z,t) - y_j^2(z,t+\Delta t))\,dz = \int_t^{t+\Delta t} \Xi^{(i)}(\tau)\,d\tau, \quad i = 0, 1, 2, \ldots \tag{44}$$

Let $\boldsymbol{\psi}_j(z) = [\,\varphi_{1,j}(z) \quad \cdots \quad \varphi_{N_j,j}(z)\,]^T$, $j = 1, 2, \ldots, n$ be basis function vectors for approximating $p_j^{(i)}(z)$, where N_j is the number of basis functions in $\boldsymbol{\psi}_j(z)$. Then, $p_j^{(i)}(z)$ is approximated by

$$\hat{p}_j^{(i)}(z) = (\boldsymbol{w}_j^{(i)})^T \boldsymbol{\psi}_j(z) = \boldsymbol{\psi}_j^T(z)\boldsymbol{w}_j^{(i)}, \tag{45}$$

where $\boldsymbol{w}_j^{(i)} = [\,w_{j,1}^{(i)} \quad \cdots \quad w_{j,N_j}^{(i)}\,]^T$ is the weight vector. Thus, the left side of Eq. (44) can be written as

$$\begin{aligned}
&\sum_{j=1}^{n} \int_{\underline{z}}^{\bar{z}} \hat{p}_j^{(i+1)}(z)(y_j^2(z,t) - y_j^2(z,t+\Delta t))\,dz \\
&= \sum_{j=1}^{n} \int_{\underline{z}}^{\bar{z}} (\boldsymbol{w}_j^{(i+1)})^T \boldsymbol{\psi}_j(z)(y_j^2(z,t) - y_j^2(z,t+\Delta t))\,dz \\
&= \sum_{j=1}^{n} (\boldsymbol{w}_j^{(i+1)})^T \int_{\underline{z}}^{\bar{z}} \boldsymbol{\psi}_j(z)(y_j^2(z,t) - y_j^2(z,t+\Delta t))\,dz \\
&= (\boldsymbol{w}^{(i+1)})^T \boldsymbol{\theta}(t, t+\Delta t)
\end{aligned}$$

where

$$\boldsymbol{w}^{(i+1)} = \left[(\boldsymbol{w}_1^{(i+1)})^T \quad \cdots \quad (\boldsymbol{w}_n^{(i+1)})^T \right]^T \text{ and}$$

$$\boldsymbol{\theta}(t, t+\Delta t) = \begin{bmatrix} \int_{\underline{z}}^{\bar{z}} \boldsymbol{\psi}_1(z)(y_1^2(z,t) - y_1^2(z,t+\Delta t))\,dz \\ \vdots \\ \int_{\underline{z}}^{\bar{z}} \boldsymbol{\psi}_n(z)(y_n^2(z,t) - y_n^2(z,t+\Delta t))\,dz \end{bmatrix}.$$

Then, we rewrite (44) as

$$(\boldsymbol{w}^{(i+1)})^T \boldsymbol{\theta}(t, t+\Delta t) = \int_t^{t+\Delta t} \hat{\Xi}^{(i)}(\tau)\,d\tau, \quad i = 0, 1, 2, \ldots \tag{46}$$

where $\hat{\Xi}^{(i)}(\tau) \triangleq \langle \boldsymbol{y}(\cdot,\tau), \boldsymbol{Q}(\cdot)\boldsymbol{y}(\cdot,\tau)\rangle + \langle \hat{\boldsymbol{u}}^{(i)}(\cdot,\tau), \boldsymbol{R}(\cdot)\hat{\boldsymbol{u}}^{(i)}(\cdot,\tau)\rangle$.

Accordingly, the control law (13) can be approximately updated by

$$\hat{\boldsymbol{u}}^{(i+1)}(z,t) = -\boldsymbol{R}^{-1}(z)\boldsymbol{B}^T(z)\hat{\boldsymbol{P}}^{(i+1)}(z)\boldsymbol{y}(z,t) \tag{47}$$

where $\hat{\boldsymbol{P}}^{(i+1)}(z) = diag\{\hat{p}_1^{(i+1)}(z), \ldots, \hat{p}_n^{(i+1)}(z)\}$.

Remark 9. Observe that the iteration from (12) to (13) in the PI algorithm is converted to the weight iteration from (46) to (47). The evaluation of the right side of Eq. (46) requires the online measured system states. After the weight vector $\boldsymbol{w}^{(i+1)}$ is computed via (46), the control policy is approximately updated by (47) accordingly.

To compute spatial integrals $\boldsymbol{\theta}(t, t+\Delta t)$, $\langle \boldsymbol{y}(\cdot,\tau), \boldsymbol{Q}(\cdot)\boldsymbol{y}(\cdot,\tau)\rangle$ and $\langle \hat{\boldsymbol{u}}^{(i)}(\cdot,\tau), \boldsymbol{R}(\cdot)\hat{\boldsymbol{u}}^{(i)}(\cdot,\tau)\rangle$ in (46), we discretize the spatial domain $[\underline{z}, \bar{z}]$ into space instances $\{z_k, z_0 = \underline{z}, z_{N_z} = \bar{z}, k = 0, 1, 2, \ldots, N_z\}$ of the same distance d, where $d \triangleq z_{k+1} - z_k = (\bar{z} - \underline{z})/N_z$. Then, the jth component of $\boldsymbol{\theta}(t, t+\Delta t)$ can be evaluated with

$$\begin{aligned}
&\int_{\underline{z}}^{\bar{z}} \boldsymbol{\psi}_j(z)(y_j^2(z,t) - y_j^2(z,t+\Delta t))\,dz \\
&= \sum_{k=0}^{N_z-1} \boldsymbol{\psi}_j(z_k)(y_j^2(z_k,t) - y_j^2(z_k,t+\Delta t))\,d, \quad j = 1, 2, \ldots, n.
\end{aligned}$$

Similarly, $\langle \boldsymbol{y}(\cdot,\tau), \boldsymbol{Q}(\cdot)\boldsymbol{y}(\cdot,\tau)\rangle$ and $\langle \hat{\boldsymbol{u}}^{(i)}(\cdot,\tau), \boldsymbol{R}(\cdot)\hat{\boldsymbol{u}}^{(i)}(\cdot,\tau)\rangle$ can also be evaluated.

Note that $\boldsymbol{w}^{(i+1)}$ has N unknown elements, where $N = \sum_{j=1}^{n} N_j$. Thus, in order to solve for $\boldsymbol{w}^{(i+1)}$, at least N equations are required. Here, we construct $\bar{N}(\bar{N} \geq N)$ equations. In each time interval $[t, t+\Delta t]$, we collect $\bar{N}$ sample state sets (each set contains N_z state data) along state trajectories, and construct the least-square solution of the weights as follows:

$$\boldsymbol{w}^{(i+1)} = (\boldsymbol{\Theta}\boldsymbol{\Theta}^T)^{-1}\boldsymbol{\Theta}\boldsymbol{\xi}^{(i)} \tag{48}$$

where

$$\boldsymbol{\Theta} = [\,\boldsymbol{\theta}(t, t+\delta t) \quad \cdots \quad \boldsymbol{\theta}(t+(\bar{N}-1)\delta t, t+\bar{N}\delta t)\,]$$

$$\boldsymbol{\xi}^{(i)} = [\,\xi_1^{(i)} \quad \cdots \quad \xi_{\bar{N}}^{(i)}\,]^T$$

with $\delta t = \Delta t/\bar{N}$ and $\xi_k^{(i)} = \int_{t+(k-1)\delta t}^{t+k\delta t} \hat{\Xi}^{(i)}(\tau)\,d\tau$, $k = 1, \ldots, \bar{N}$.

It is worth mentioning that the least-square method (48) should satisfy the persistence of excitation (PE) condition, which can be obtained by injecting probing noises into states.

Based on the above least-square method for estimating weights, the specific implementation of the adaptive distributed optimal control for the PDE system (1)–(3) can be represented as follows:

Algorithm 2. *Step 1:* Select basis function vectors $\boldsymbol{\psi}_j(z)$, $j = 1, 2, \ldots, n$. Give an initial weight vector $\boldsymbol{w}^{(0)}$ such that $\hat{\boldsymbol{u}}^{(0)}(z, t) = -\boldsymbol{R}^{-1}(z)\boldsymbol{B}^T(z)\hat{\boldsymbol{P}}^{(0)}(z)\boldsymbol{y}(z, t)$ is an exponentially stabilizing control law, where each element of $\hat{\boldsymbol{P}}^{(0)}(z)$ is computed via (45). Let $i = 0$.

Step 2: With control policy $\hat{\boldsymbol{u}}^{(i)}$, collect $\bar{N}$ sample data sets along state trajectories of the closed-loop PDE system, and evaluate the associated matrices $\boldsymbol{\Theta}$ and $\boldsymbol{\xi}^{(i)}$ during time interval $[i \cdot \Delta t, (i+1) \cdot \Delta t]$. Compute $\boldsymbol{w}^{(i+1)}$ via (48) at time instant $t = (i+1) \cdot \Delta t$.

Step 3: Update the control policy with (47) at time instant $t = (i+1)\Delta t$.

Step 4: Set $i = i + 1$. If $\left\| \boldsymbol{w}^{(i)} - \boldsymbol{w}^{(i-1)} \right\| \le \varepsilon$ (ε is a small positive real number), stop iteration and the weights remains invariable, else, go to Step 2 and continue.

Remark 10. Note that in [45], a computational efficient model predictive control (MPC) approach was proposed to solve the optimal control problem of hyperbolic PDE systems, and good results were achieved. However, there are two main differences between the developed PI algorithm and the MPC in [45]. Firstly, the MPC needs appropriate analytical model [45–47] and computes open-loop policies offline (as mentioned in [45]) which are then used for real time control purpose, while the PI algorithm does not require the internal system dynamics and computes optimal control policy online by conducting closed-loop simulations. Secondly, the MPC optimization problems require extra solvers (such as interior-point method in [47]), while the proposed PI algorithm is a direct optimal control method based on dynamic programming [26] in which the optimization is merged into the online learning process. Moreover, it has been shown in [47] via a comparative study for discrete-time ODE systems that RL method may certainly be competitive with MPC even in contexts where a good deterministic system model is available.

5. Simulation studies

To illustrate the effectiveness of the developed adaptive distributed optimal control approach, we conduct simulation studies on a steam-jacket tubular heat exchanger [1]. The dynamic model of which has the form:

$$\frac{\partial T}{\partial t} = -\nu \frac{\partial T}{\partial z} - \frac{hA}{\rho C_p}(T - T_w) \tag{49}$$

subject to the boundary condition

$T(0, t) = T_f, t \in [0, +\infty)$

and initial condition

$T(z, 0) = T_0(z), z \in [0, L]$.

In the above model, T denotes the temperature in the tubular heat exchanger, T_w, T_f and T_0 denote steam-jacket temperature, heat exchanger inlet constant temperature and initial temperature, respectively. In addition, t, z and L denote the independent time and space variables, and the length of the exchanger, respectively.

By taking change of variables as

$$y \triangleq T - T_f, \quad u \triangleq T_w - T_f, \quad y_0 \triangleq T_0 - T_f \quad \text{and } a(z) = \frac{hA}{\rho C_p}, \tag{50}$$

the equivalent representation of the model (49) is obtained as follows:

$$\frac{\partial y}{\partial t} = -\nu \frac{\partial y}{\partial z} - a(z)y + a(z)u \tag{51}$$

subject to the boundary condition

$y(0, t) = 0, t \in [0, +\infty)$

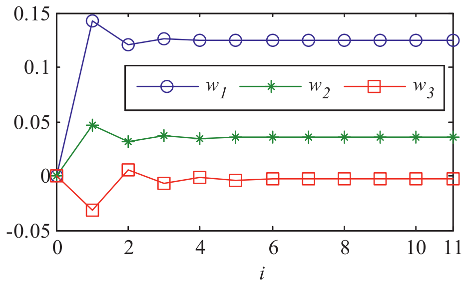

Fig. 1. The weights w_1,w_2 and w_3 in each iteration.

and initial condition

$y(z, 0) = y_0(z), z \in [0, L]$.

The control objective is to determine the adjustment in the steam-jacket temperature T_w (through a steam inlet valve), such that the exchanger temperature T approaches the desired temperature profile $T_d(z)$. Here, we are interested in the constant profile T_d. Since $T_d(0) = T_f$, then $T_d(z) = T_f$ for all $z \in [0,L]$. Thus, the desired profile for system (51) is $y_d \triangleq T_d - T_f = 0$. According to (9), the distributed optimal control of system (51) is to solve the following SDRDE for $p^*(z)$:

$$\begin{aligned} -\nu \frac{\partial p^*(z)}{\partial z} &= -a(z)p^*(z) - p^*(z)a(z) + q(z) \\ &-p^*(z)a(z)r^{-1}(z)a(z)p^*(z), \; p^*(z) > 0, \; z \in [0, L], \; p^*(L) = 0. \end{aligned} \tag{52}$$

Now, we apply the developed adaptive distributed optimal control method (i.e., Algorithm 2) to online learn the solution of SDRDE (52). The system parameters are assumed to be: $\nu = 1$, $a(z) = 8 - 9\exp(-2z/L)$, $L = 1$ and $T_f = 340$ (then, $T_d = 340$). Let $y_0(z) = 0.5T_f \sin(4\pi z/L)$ (i.e., the initial temperature profile is assumed to be $T_0(z) = T_f + 0.5T_f \sin(4\pi z/L)$). Choose $\boldsymbol{Q}(z)$ and $\boldsymbol{R}(z)$ in cost functional (7) as $\boldsymbol{Q}(z) = q(z) = 1$ and $\boldsymbol{R}(z) = r(z) = 1$, $z \in [\underline{z}, \bar{z}]$. We select 5 (i.e., $N = N_1 = 5$) basis functions as $\varphi_k(z) = \sin(k\pi z/L)$, $k = 1, \ldots, 5$ for approximating $p^*(z)$. Let w_k be the weights and set initial values as $w_k^{(0)} = 0$, $k = 1, \ldots, 5$. In Algorithm 2, select the value of stop criterion $\varepsilon = 10^{-5}$, the sampling step size in space $d = 0.02$ (i.e., $N_z = 50$) and in time $\delta t = 0.02(s)$. In each iterative step, after collecting 10 (i.e., $\bar{N} = 10$) system state sets, the least-square method (48) is used to update weights, that is, the weights is updated every $0.2(s)$ (i.e., $\Delta t = 0.2(s)$). In order to satisfy the PE condition, we reset the system state as initial state after each update. Figs. 1 and 2 show the weights in each iterative step, where we observe that the weight vector converges to $\begin{bmatrix} 0.1247 & 0.0353 & -0.0030 & -0.0129 & 0.0300 \end{bmatrix}^T$ with accuracy ε at $i = 11$ iteration (i.e., at time instant $t = 2.2(s)$). That is, the $\hat{p}(z)$ is convergent at time $t = 2.2(s)$, then, the PI algorithm is terminated and the $\hat{p}(z)$ remains invariable from this instant on.

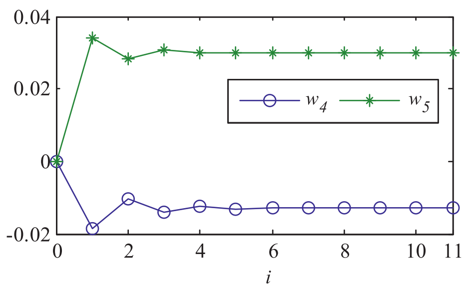

Fig. 2. The weights w_4 and w_5 in each iteration.

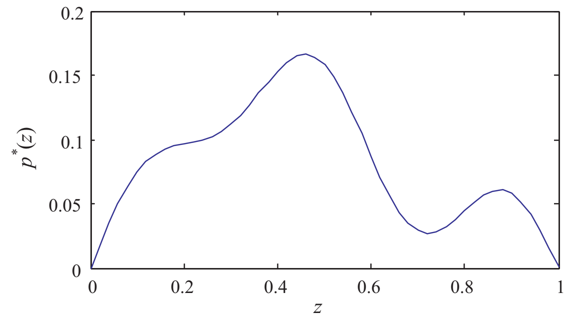

Fig. 3. The profile of $p^*(z)$.

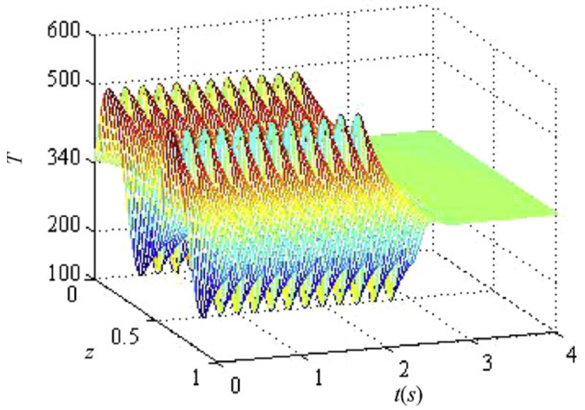

Fig. 4. The evolution of exchanger's temperature profile.

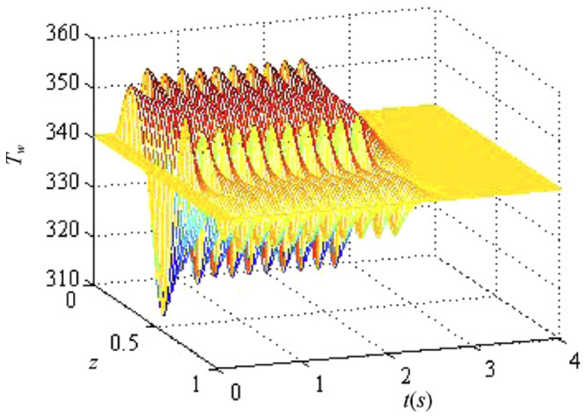

Fig. 5. The evolution of jacket's temperature profile.

It can be seen from the figures that the PI algorithm converges very fast. Fig. 3 gives the final profile of $\hat{p}(z)$ (denoted as $p^*(z)$). Figs. 4 and 5 show the actual exchanger's and jacket's temperature profile, respectively.

6. Conclusions

This paper has addressed the adaptive optimal control problem of linear hyperbolic PDE systems from the DTR framework, where the internal system dynamics is unknown. The thought of RL technique is introduced to solve this problem. An adaptive distributed optimal control method based on PI and least-square function approximation is developed. The PI algorithm was proposed to learn the solution of SDRDE by measuring system state online. The convergence of the algorithm is established by showing that its equivalent sequence of SDLDEs is convergent. For implementation purpose, the PI algorithm is realized by using a least-square method to approximate the solutions of the SDLDEs. Finally, by implementing the developed control method on a steam-jacketed tubular heat exchanger, the achieved simulation results illustrate its effectiveness.

Acknowledgment

This work was supported in part by the National Basic Research Program of China (973 Program) (2012CB720003); in part by the National Natural Science Foundation of China under Grants 61074057, 61121003, and 91016004. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] W.H. Ray, Advanced Process Control, McGraw-Hill, New York, 1981.
[2] P.D. Christofides, Nonlinear and Robust Control of PDE Systems: Methods and Applications to Transport-Reaction Processes, Birkhäuser, Boston, MA, 2001.
[3] J.L. Lions, Optimal Control of Systems Governed by Partial Differential Equations, Springer Verlag, New York, 1971.
[4] R.F. Curtain, H.J. Zwart, An Introduction to Infinite-dimensional Linear Systems Theory, Springer Verlag, New York, 1995.
[5] A. Bensoussan, G. Da Prato, M.C. Delfour, S.K. Mitter, Representation and Control of Infinite Dimensional Systems, Second ed., Birkhäuser, Boston, 2007.
[6] B. Luo, H.-N. Wu, Approximate optimal control design for nonlinear one-dimensional parabolic PDE systems using empirical eigenfunctions and neural network, IEEE Transactions on Systems, Man, and Cybernetics. Part B, in press.
[7] M. Li, P.D. Christofides, Optimal control of diffusion–convection–reaction processes using reduced-order models, Computers and Chemical Engineering 32 (9) (2008) 2123–2135.
[8] C. Xu, Y. Ou, E. Schuster, Sequential linear quadratic control of bilinear parabolic PDEs based on POD model reduction, Automatica 47 (2) (2011) 418–426.
[9] V. Yadav, R. Padhi, S.N. Balakrishnan, Robust/optimal temperature profile control of a high-speed aerospace vehicle using neural networks, IEEE Transactions Neural Network 18 (4) (2007) 1115–1128.
[10] R. Padhi, S.N. Balakrishnan, T. Randolph, Adaptive-critic based optimal neuro control synthesis for distributed parameter systems, Automatica 37 (8) (2001) 1223–1234.
[11] M.J. Balas, Finite dimensional control of distributed parameter systems by Galerkin approximation of infinite dimensional controller, Journal of Mathematical Analysis and Applications 114 (1) (1986) 17–36.
[12] I.Y. Smets, D. Dochain, J.F. Van Impe, Optimal temperature control of a steady-state exothermic plug-flow reactor, AIChE Journal 48 (2) (2002) 279–286.
[13] I. Aksikas, J.J. Winkin, D. Dochain, Optimal LQ-feedback regulation of a nonisothermal plug flow reactor model by spectral factorization, IEEE Transactions on Automatic Control 52 (7) (2007) 1179–1193.
[14] I. Aksikas, A. Fuxman, J.F. Forbes, J.J. Winkin, LQ control design of a class of hyperbolic PDE systems: application to fixed-bed reactor, Automatica 45 (6) (2009) 1542–1548.
[15] M. Krstic, A. Smyshlyaev, Backstepping boundary control for first-order hyperbolic PDEs and application to systems with actuator and sensor delays, Systems Control Letters 57 (9) (2008) 750–758.
[16] J.M. Coron, B. D'Andrea-Novel, G. Bastin, A strict Lyapunov function for boundary control of hyperbolic systems of conservation laws, IEEE Transactions on Automatic Control 52 (1) (2007) 2–11.
[17] A. Diagne, G. Bastin, J.M. Coron, Lyapunov exponential stability of 1D linear hyperbolic systems of balance laws, Automatica 48 (1) (2011) 109–114.
[18] H. Sira-Ramirez, Distributed sliding mode control in systems described by quasilinear partial differential equations, Systems Control Letters 13 (2) (1989) 177–181.
[19] H. Shang, J.F. Forbes, M. Guay, Model predictive control for quasilinear hyperbolic distributed parameter systems, Industrial and Engineering Chemistry Research 43 (9) (2004) 2140–2149.
[20] P.D. Christofides, P. Daoutidis, Feedback control of hyperbolic PDE systems, AIChE Journal 42 (11) (1996) 3063–3086.
[21] J.-W. Wang, H.-N. Wu, H.-X. Li, Distributed fuzzy control design of nonlinear hyperbolic PDE systems with application to nonisothermal plug-flow reactor, IEEE Transactions on Fuzzy Systems 19 (3) (2011) 514–526.
[22] H.-X. Li, C.-K. Qi, Spatio-Temporal Modeling of Nonlinear Distributed Parameter Systems: A Time/Space Separation Based Approach, Springer Verlag, New York, 2011.
[23] T. Okita, Y. Kobayashi, S. Tanaka, An identification of hyperbolic distributed parameter systems using transient response, Electrical Engineering in Japan 101 (3) (1981) 131–138.
[24] H. Zhang, Y. Quan, Modeling, identification, and control of a class of nonlinear systems, IEEE Transactions on Fuzzy Systems 9 (2) (2001) 349–354.

[25] L.P. Kaelbling, M.L. Littman, A.W. Moore, Reinforcement learning: a survey, Journal of Artificial Intelligence Research 4 (1996) 237–285.
[26] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, MIT Press, Cambridge, Mass, 1998.
[27] J. Morimoto, K. Doya, Robust reinforcement learning, Neural Computation 17 (2) (2005) 335–359.
[28] F.L. Lewis, D. Vrabie, Reinforcement learning and adaptive dynamic programming for feedback control, IEEE Circuits and Systems Magazine 9 (3) (2009) 32–50.
[29] W.B. Powell, Approximate Dynamic Programming: Solving the Curses of Dimensionality, Wiley-Interscience, Hoboken, N.J, 2007.
[30] J. Si, Y. Wang, Online learning control by association and reinforcement, IEEE Transactions Neural Network 12 (2) (2001) 264–276.
[31] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, F.L. Lewis, Adaptive optimal control for continuous-time linear systems based on policy iteration, Automatica 45 (2) (2009) 477–484.
[32] D. Vrabie, F.L. Lewis, Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems, Neural Networks 22 (3) (2009) 237–246.
[33] K.G. Vamvoudakis, F.L. Lewis, Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem, Automatica 46 (5) (2010) 878–888.
[34] J.J. Murray, C.J. Cox, G.G. Lendaris, R. Saeks, Adaptive dynamic programming, IEEE Transactions on Systems, Man, and Cybernetics. Part C 32 (2) (2002) 140–153.
[35] D.L. Kleinman, On an iterative technique for Riccati equation computations, IEEE Transactions on Automatic Control 13 (1) (1968) 114–115.
[36] M. Abu-Khalaf, F.L. Lewis, Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach, Automatica 41 (5) (2005) 779–791.
[37] K. Vamvoudakis, D. Vrabie, F. Lewis, Online policy iteration based algorithms to solve the continuous-time infinite horizon optimal control problem, in: IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning, Nashville, TN, 2009, pp. 36–41.
[38] H. Zhang, Q. Wei, Y. Luo, A novel infinite-time optimal tracking control scheme for a class of discrete-time nonlinear systems via the greedy HDP iteration algorithm, IEEE Transactions on Systems, Man, and Cybernetics. Part B 38 (4) (2008) 937–942.
[39] H. Zhang, L. Cui, X. Zhang, Y. Luo, Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method, IEEE Transactions Neural Network 22 (2011) 2226–2236 (12, part: 2).
[40] I. Karafyllis, P. Daoutidis, Control of hot spots in plug flow reactors, Computers and Chemical Engineering 26 (7–8) (2002) 1087–1094.
[41] J.J. Winkin, D. Dochain, P. Ligarius, Dynamical analysis of distributed parameter tubular reactors, Automatica 36 (3) (2000) 349–361.
[42] H.K. Rhee, R. Aris, N.R. Amundson, First-Order Partial Differential Equations: Vols. I and II, Prentice Hall, New Jersey, 1986.
[43] J. Smoller, Shock Waves and Reaction–Diffusion Equations, Springer-Verlag, Berlin, 1983.
[44] R.F. Curtain, A.J. Pritchard, Functional Analysis in Modern Applied Mathematics, Academic Press, London, New York, 1977.
[45] S. Dubljevic, P. Mhaskar, N.H. El-Farra, P.D. Christofides, Predictive control of transport–reaction processes, Computers and Chemical Engineering 29 (11–12) (2005) 2335–2345.
[46] E.F. Camacho, C. Bordons, Model Predictive Control, Springer-Verlag, London, 1999.
[47] D. Ernst, M. Glavic, F. Capitanescu, L. Wehenkel, Reinforcement learning versus model predictive control: a comparison on a power system problem, IEEE Transactions on Systems, Man, and Cybernetics. Part B 39 (2) (2009) 517–529.