Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 30 No. 2 (2023)

Constructing a Precise Method to Control Non-Linear Systems Employing Special Functions and Machine Learning

S.K. Sahani[1*], C. Yadav[2], K. Sahani[3*], and V.V. Singh[4]

[1]Department of Mathematics, Janakpur Campus, T.U., Janakpurdham, Nepal
[*2]Institute of Science and Technology, Patan Multiple Campus, T.U., Nepal
[3]Department of Civil Engineering, Kathmandu University, Dhulikhel, Nepal
[4]Professor Mathematics, Department Lingaya's Vidhyapeeth, Faridabad, India
Email:drsks53086@gmail.com[1*],chandrakantyadava@gmail.com[2],kameshwar.sahani@ku.edu.np[3*],singh_vijayvir@yahoo.com[4]

Article History:
Received: *08-05-2023*
Revised: *02-06-2023*
Accepted: *15-07-2023*

Abstract:

This paper is about finding the best way to control certain types of equations that describe heat and diffusion using methods that are not exact but close to the best solution. A new way to control things better is suggested using patterns we've seen before and computer learning. By employing the data gathered from the EEFs are computed using the Karhunen-Loève decompose in the PDE computer. The EEFs are used to convert the PDE system into a high-order ODE system. A smaller model (ROM) that may depict the primary behavior of the PDE subsystem is created using the SP approach. This facilitates understanding of the system. In order to reduce its size, a more basic model (ROM) that demonstrates the primary motions of the PDE system is created using the SP approach. Next, the optimal controller for the ROM is designed using the HJB technique. This ensures that the high-order ODE system is stable using the SP theory. By splitting the best way to control something into two sections, we can solve a math problem to figure out the first part and come up with a new equation for the second part. Third, a method is suggested for updating and controlling based on small improvements to solve a specific equation, and it has been proven to work effectively. Moreover, we use a technique called the NN approach to estimate the cost function. Finally, the simulation results demonstrate the effectiveness of the proposed optimum control approach when applied to a diffusion-reaction process using a unique spatially component.

Keywords: The HJB equation, KLD, NN, PDE systems, optimal control, and SP are all complicated mathematical concepts.

1. Introduction

Optimal control theory is a useful method for making controllers. There have been numerous significant discoveries regarding the optimal methods for regulating linear or nonlinear systems characterized by ordinary differential equations. These elements include the linear parabolic controller hypothesis, Bellman dynamic programming, and the application of the greatest assumption. [1], [2].

In real life, many industrial processes are spread out in different places, so how they work depends on where they are as well as when they are.

These systems are often explained using a group of complex math equations with specific boundary rules. - Optimal control methods are difficult to use for designing a controller in real-time with a regular computer due to the complex dimensions of PDE systems. In the last few decades, Rutkowski [3] has done important work in developing the best way to control PDE systems [3], Wang, Tung, [4], and Lions [5] looked at math stuff, and you can find more theoretical results in other places too. Meanwhile, people have studied how to control PDE systems effectively. Engineers have worked on ways to minimize performance measures using [6] LQ methods [7]. Meanwhile, people have studied how to control PDE systems from an engineering perspective. They have focused on methods that use LQ performance indices to minimize and improve control. Previous studies on how to make the best controller for PDE systems [8]–[10], can be divided into two types: first designing and then simplifying, and first simplifying and then designing approaches. In the first, optimal controller for PDE networks are designed using sophisticated mathematics. [8] These controllers are then simplified for use. Curtain, Zwart, Aksikas and their colleagues [9] Using algebraic operator Riccati formulae, I was able to solve the LQ ideal control issue for devices with a lot of states. Aksikas together with others. [10] I used a method called spectral factorization to find the feedback operator by solving a special type of mathematical equation. A novel approach to linear parabolic PDE system control was developed by Ray at the age of [11]. He used the original PDE model and found a way to control the system by figuring out a unique equation. Conversely, nevertheless, the reduce-then-design methods first turn the PDE system into a simpler ODE model and then use it for control design [12]–[19]. A way to control the amount of chemicals in a reactor was suggested in a book. [12] Also, two methods to control chemical reactions were created using different techniques. The investigation was undertaken by Xu and his peers. [13] They studied a type of math problem called a bilinear parabolic PDE system. A plan to make a not-perfect controller using a simple model. [14] Sadek and Bokhari used special math to estimate the state variables in a control problem with linear PDEs. [15] The Navier-Stokes equation was outfitted with a top-notch controller by 16-year-old Ravindra, who employed the Newton method to achieve optimal functionality. In Yadav et al. 's research, they used a method called approximate dynamic programming (ADP). [17] In the study by Padhi and others, make better microcontrollers by putting them together, [18], [19]. It should be emphasized that In the PDE problems covered in those works, the spatial differential operators are usually linear. [8]-[10], [12]-[19].

However, SDOs in actual industries are frequently nonlinear. For instance, in the process of rapid chemical vapour deposition, [11], [20], [21] Dispersion and react are nonlinear processes that rely on temperatures and conductivity of heat. [22], [23]. In this case, we want to create the best ways to control non-linear problems.

When the spatial operator is not straight, it's hard or impossible to figure out the basic functions using math. To get around this problem, some ways of using data to find patterns have been used. Examples of such methods are Karhunen-Loève decomposition (KLD) [14], [16]–[25] and singular-value decomposition. [26], [27]. By making use of EEFs, it is possible to generate a smaller ODE system that is then applied in the development of a controller.

However, only a few studies focused on finding the best way to control something, like Armaou and Christof ides [22]. Decreased gradient methods were used to solve the dynamical nonlinear code issue with equality limitations that resulted from the transformation of the optimum control problem. In recent years, the HJB method has been useful for designing the best way to control nonlinear systems. Some important progress has also been made for LPSs. nevertheless up until now, solving the HJB formulas with algebra or numbers has proven to be extremely difficult. Recently, researchers have used ADP to solve this problem in a specific way. [16]–[19], [28]–[36] This method has been used to solve this problem in a careful and quiet way. ADP was suggested in [28]. There are two primary categories: bidirectional robust computer and heuristic dynamic programming [29]–[33], [31], [34].

More algorithms and a scheme for all ADP kinds were developed by Prokhorov and Wunsch. Check out the special issue [36] for further details on the use of ADP to provide input regulation. A method for solving the HJB problem for continuously running systems was created by Saridis and Lee, but they didn't figure out how to find the cost function for each step of the process. Beard and others [38], [39] proposed a way to solve the HJB equation by solving a series of simpler equations using a method called successive approximation. This method uses Galerkin approximation to solve the equations. Abu-Khalaf and Clark [40] and [41] created near-optimal state feedback controllers for continuous nonlinear systems where control restrictions are present, drawing inspiration from the work of [37]–[39]. Cheng et al. [42] and [43] employed neural networks instead of a policy iteration to answer the evolving HJB equation. They also applied this approach to the situation where inputs are restricted. The authors have limited knowledge of research on the most effective approach to managing PDE problems employing HJB equations and a regressive SDO. In this paper, we create a simpler controller for systems with certain equations using a special framework. We base our work on certain functions and neural networks. We start by using the KLD to figure out the EEFs using snapshots. The PDE network is constructed using the EEFs as a high-dimensional, singularly perturbed mathematical representation of ODEs. The SP technique helps to create a simpler model that focuses on the most important parts of the PDE system. The optimal controls for the system are designed using the ROM. This guarantees the long-term stability of the high-order ODE system. To solve the HJB-like equation, we suggest a method that uses gradual improvements and neural networks. We also show that this method will always work. Finally, we did some tests on a computer to see how well our suggested way of controlling a chemical reaction works. We found that it worked well. To sum up, this study's key contributions can be categorized into three main areas.

Create a way to make a complex math method simpler for parabolic equations with a nonlinear part.

Suggest a better way to control non-linear PDE systems and come up with a new type of equation. The SP technique proves that the closed-loop high-order ODE system is stable in the long run.

Create a plan to update controls to solve a specific equation using a method that makes small changes and neural networks. Then show that this plan will work.

This is how the remainder of the paper is structured: The definition of elliptic PDE systems is given in **Section II.** In the third part, the KLD approach and the SP technique are used to generate a ROM. The best controller is made and explained in Section IV. Section V is about testing a model, and Section VI is the ending.

In short, this study has three main contributions.

1. Simplify a way to make parabolic PDE systems with a nonlinear SDO smaller using KLD and SP technique.
2. Suggest a good way to control systems that follow a certain mathematical rule, and come up with a new type of mathematical equation to help with this. The long-term stability of the closed-loop high-order ODE system is demonstrated by the SP approach.

3. Create a plan to update the control in order to solve a specific equation using a method where we keep improving our solution step by step and also use a neural network. Then show that this plan works and our solution keeps getting better.

Symbols: Rn×m stands for all real n × m matrices as R represents n-dimensional space, and R is actual amounts. Rn and d stand for the usual way of multiplying things together and measuring their length in Rn, in that order. If a matrix $\boldsymbol{M}$ is symmetrical, $\boldsymbol{M} > (<) 0$ means it is positive (negative) definite. The symbols Σ() and σ() represent the biggest and smallest singular values of a matrix.

The small letter T written above a matrix means to switch the rows and columns. $L^2([z, z], R^n)$ is a big space of vectors with a certain kind of math inside, and it's related to a type of function called square integrable. It's used for working with vectors in a certain way, and has a specific way of measuring their size and how they relate to each other.

$$\langle \boldsymbol{\omega}_1(\cdot), \boldsymbol{\omega}_2(\cdot) \rangle = \int_{z}^{\bar{z}} \langle \boldsymbol{\omega}_1(z), \boldsymbol{\omega}_2(z) \rangle_{R^n} \, dz$$

$$\|\boldsymbol{\omega}_1(\cdot)\|_2 = \langle \boldsymbol{\omega}_1(\cdot), \boldsymbol{\omega}_1(\cdot) \rangle^{\frac{1}{2}}$$

where $\boldsymbol{\omega}_1(\cdot)$ and $\boldsymbol{\omega}_2(\cdot)$ are any two elements of $L^2([\underline{z}, \bar{z}], R^n)$.

Explanation of Curved PDE Systems

In this paper, we study a group of curved PDE equations in one direction with a certain way of representing the state.

$\boldsymbol{y}t(z, t) = \boldsymbol{A}\,(\boldsymbol{y}(z, t)) + \boldsymbol{f}\,(y(z, t)) + \boldsymbol{B}(z)\boldsymbol{u}(t)$ (1)

Dependent on the limits or rules.

$$\begin{aligned} \boldsymbol{M}_1\boldsymbol{y}(\underline{z}, t) + \boldsymbol{N}_1\, \boldsymbol{y}_z(z, t)|_{z=\underline{z}} &= \boldsymbol{d}_1 \\ \boldsymbol{M}_2\boldsymbol{y}(\bar{z}, t) + \boldsymbol{N}_2\, \boldsymbol{y}_z(z, t)|_{z=\bar{z}} &= \boldsymbol{d}_2 \end{aligned} \qquad (2)$$

and the initial condition

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

The suffixes z and t are part derivatives with a relationship with z and t, while y(z, t) is the state function with components y1(z, t) to yn(z, t). What is written as y(z, t) is the state. The letters z and t represent partial derivatives.

The definition is related to the time coordinate, which is represented by the symbol [0,). The letter "u" of time "t" Rp is the input that is controlled. (y) is a curved SDO that involves some math stuff and follows a certain rule.

t [0,) is the time coordinate and represents the values greater than or equal to 0. The manipulated input is u(t) Rp. The equation (y) is a curved line with some rules that involves different types of math and meets a certain condition.

$$\|\boldsymbol{A}(\boldsymbol{y}_1) - \boldsymbol{A}(\boldsymbol{y}_2)\|_2 \le a_1 \|\boldsymbol{y}_1 - \boldsymbol{y}_2\|_2 + a_2 \|(\boldsymbol{y}_1 - \boldsymbol{y}_2)_z\|_2 + a_3 \|(\boldsymbol{y}_1 - \boldsymbol{y}_2)_{zz}\|_2 \qquad (4)$$

Where a1, a2, and a3 are real numbers that are positive. ***F*** (y) is a function that doesn't follow a straight line and meets the condition ***f*** (0) = 0. It is also continuous in a small neighborhood around each point. ***B*** (z) is a smooth matrix function that shows how control actions are spread out in different areas.

Where genuine negative integers a1, a2, and a3 are involved. ***f (y)*** is a globally a Lipschitz continuously nonlinear vector function that satisfies ***f*** (0) = 0. The adequate smooth matrix function of dimension suitable, The distribution of control operations in spatial domains is believed to be characterized by ***B(z)***.

The starting point is $\boldsymbol{y}_0(z)$, continuous vector d1 and d2 are constant matrix $\boldsymbol{M}_1$, $\boldsymbol{N}_1$, $\boldsymbol{M}_2$, and $\boldsymbol{N}_2$. In this research, it is assumed that the free- and closed-loop cases of the PDE system (1)–(3) are well-stated. Complex calculations will be needed for the well-posedness analysis of the PDE system (1)–(3), even for basic PDE networking with linear SDOs. [44]. We leave it for future research because the controller's architecture is the primary topic of this paper. The operator A has a specific domain, and the input operator is defined as $\boldsymbol{u}=\boldsymbol{B}(z)\boldsymbol{u}(t)$. Therefore, the nonlinear PDE system (1)-(3) can be restated in a simpler way.

$$\begin{cases} \boldsymbol{y}_t = \boldsymbol{A}(\boldsymbol{y}) + \boldsymbol{f}(\boldsymbol{y}) + \mathrm{B}\boldsymbol{u} \\ \boldsymbol{y} \in \mathrm{D}(\boldsymbol{A}) \\ \boldsymbol{y}(z, 0) = \boldsymbol{y}_0(z). \end{cases} \qquad (5)$$

III means the third in a sequence or the roman numeral for 3. Simplifying the Reduced-Order ODE System Formula

The nonlinear SDO of PDE structure (5) prevents the computation of the mathematical formulae for the eigenvalues and eigenfunctions of the PDE system. As a result, they prohibit the direct creation of finite-dimensional approximations of the PDE system using conventional basis symbol sets in combination with model reduction strategies like Galerkin's approach and symmetric collocation.

To circumvent this issue, we calculate a collection of EEFs (dominant spatial patterns) of the PDE system via the retrospective technique first, using KLD [24]. These EEFs will then be applied in conjunction with the SP method to obtain a ROM that precisely explains the prevailing dynamics of the nonlinear PDE system.

A. EEF Computation Using KLD

In many engineering domains, KLD is a well-liked statistical pattern analysis technique for identifying the dominating structure in a collection of a multidimensional process and deriving low-dimensional accurate descriptions. Given a set of data, KLD gives an estimate of how much each EEF contributes in relation to the group's total "electricity" (mean-square fluctuation). In addition, it generates a collection of orthogonal EEFs for the mathematical model that comprise the community. The image of the trimmed series of games, as has a lower mean-square error than the other basis of one dimensions is thus best served by the EEFs as an ideal basis. Stated differently, more energy is captured by the projection onto the first few EEFs than by any subsequent projection. Because of these characteristics, the EEFs are a suitable choice to take into account while reducing models. We now quickly go over the KLD process for composing EEFs using the quick snapshot technique within the context of divergent dynamic PDE circuits. Collect M sampling states of the PDE system (with u 0) (5), represented as **yi(z)**, so that the set is sufficiently big (referred to as an ensemble). ***yi***(*z*), $_i$ = 1,...,M, are referred to as "pictures" of the (5) solution. Given the assumption that the snapshots ***yi***(*z*) obtained from the PDE system (5) exhibit ergodic behavior [45], the following is an expression for the spatial association value:

$$\boldsymbol{R}(z,\xi) = \lim_{t\to\infty} \frac{1}{t}\int_0^t \boldsymbol{y}(z,\tau)\boldsymbol{y}^{\top}(\xi,\tau)d\tau$$

which is approximately calculated with

$$\boldsymbol{R}(z,\xi) = (1/M)\sum_{i=1}^{M} \boldsymbol{y}^{i}(z)\boldsymbol{y}^{T}(\xi). \qquad (6)$$

The Mercer theorem [46] states that ***R***(*z*, ξ) possesses a characteristic that

$$\boldsymbol{R}(z,\xi) = \sum_{i=1}^{\infty} \lambda_i \boldsymbol{\varphi}_i(z)\boldsymbol{\varphi}_i^{T}(\xi)$$

where the orthogonal eigenfunctions of ***R***(z,ξ) are represented by ***φi***(*z*) while the positive eigenvalues by ***λi***. Taking into account the subsequent integral

$$\int_{z}^{\bar{z}} \boldsymbol{R}(z,\xi)\boldsymbol{\varphi}_i(\xi)d\xi = \int_{z}^{\bar{z}} \sum_{j=1}^{\infty} \lambda_j \boldsymbol{\varphi}_j(z)\boldsymbol{\varphi}_i^{\top}(\xi)\boldsymbol{\varphi}_i(\xi)d\xi$$

$$= \lambda_i \boldsymbol{\varphi}(z) \qquad (7)$$

and employing (6), we've

$$\int_{z}^{\bar{z}} \boldsymbol{R}(z,\xi)\boldsymbol{\varphi}_i(\xi)d\xi = \int_{z}^{\bar{z}} \frac{1}{M}\sum_{i=1}^{M} \boldsymbol{y}_j(z)\boldsymbol{y}_j^{\top}(\xi)\boldsymbol{\varphi}_i(\xi)d\xi$$

$$= \sum_{j=1}^{M} \vartheta_j \boldsymbol{y}_j(z) \qquad (8)$$

$\vartheta_j = (1/M)\int_{z}^{\bar{z}} \boldsymbol{y}_j^{\top}(\xi)\boldsymbol{\varphi}_i(\xi)d\xi$. From (7) and (8), we get

$$\boldsymbol{\varphi}_i(z) = \sum_{j=1}^{M} \alpha_{ji}\boldsymbol{y}_j(z) \qquad (9)$$

Where, $\alpha ji = \lambda^{-}_{i} 1\vartheta_j$.

Observe that the EEFs {\i(z)} are merely linear mixtures of snapshots $\{y_i(z)\}$; hence, the computation of the coefficients is all that is needed to compute the EEFs. $[\boldsymbol{\alpha}_1 \cdots \boldsymbol{\alpha}_1 i < a_{Mi}] = \Delta T$. Reorganizing (7) and putting (6) and (9) into (7) produce

$\boldsymbol{Y}(z)(\boldsymbol{C}\boldsymbol{\alpha}_i) = \boldsymbol{Y}(z)(\lambda_i\boldsymbol{\alpha}_i)$ (10)

where $\boldsymbol{C} \triangleq (c_{kj})_{M\times M} \in \mathrm{R}^{M\times M}$, $c_{kj} = (1/M)\int_{\underline{z}}^{\bar{z}} \boldsymbol{y}^{k}(\xi)\boldsymbol{y}^{j}(\xi)d\xi$,

and $\boldsymbol{Y}(z) \triangleq [\boldsymbol{y}^1(z)\cdots\boldsymbol{y}^M(z)] \in \mathrm{R}^{M\times M}$.

The conventional method of evaluation deconstruction can be used to calculate all Eigenvectors are $\{\boldsymbol{\alpha} i\}$ and related eigenvalues $\{\lambda_i\}$ for the subsequent eigenvalue issues include $\boldsymbol{C}\boldsymbol{\alpha}_i = \boldsymbol{\lambda}_i\alpha i$, assumes linear independence of the set $\{\boldsymbol{yi}(z)\}$. After standardising the eigenvector αi to meet αi, αi RM = $1/(M\lambda i)$, EEFs can be globally orthogonal. Next, using (9), all EEFs φi(z) are obtained directly.

Remark 1: It is important to note because in the PDE system (5), i.e., $(\boldsymbol{\varphi i}(z)) = \boldsymbol{\lambda i\varphi i}(z)$, The Eigen functions and eigenvalues of the operator $(\boldsymbol{y})$ are not the EEFs and their corresponding eigenvalues. The amplitude of an EEF, which stands for the "architecture" that symbolizes the cohesion of the outfit, which consisted is the matching force that is recorded.

B. Using SP Methodology to Derive Reduced-Order ODE Modeling

The primary characteristic of parabolic PDE systems, which are very dissipative, is the ability to divide the SDO's eigen spectrum into a stable fast complement and a finite-dimensional slow one. This suggests that finite-dimensional slow systems can adequately explain the prevailing dynamic behavior of such systems [20], [21], [48]. Nevertheless, it is frequently impossible to obtain the nonlinear SDO's eigen spectrum. The division of the PDE system using EEFs is essentially not distinctive from using other conventional basis functions sets, such as the legendary equations and sine and cosine functions. if the collection of pictures is sufficiently vast and includes sufficient data on the PDE system's worldwide fluctuations [21]. By omitting the fast modes outright, Galerkin's technique was used in [17]–[19], [21], and [22] to develop an EEF-based finite-dimensional ODE model. In this part, we apply the SP technique—basically a multitime-scale approach—to derive a ROM based on EEFs. For an estimate of your the PDE system, we first employ a high-order ODE model based on EEFs (5). To further minimise its size, an SP model of a linked fast/slow ODE systems is built, and a fast moments scale is added to achieve a ROM. The eigenvalues of matrix $\boldsymbol{C}$ are represented by $\{\boldsymbol{\lambda_i}\}$, and the equivalent EEFs in KLD are represented by $\{\boldsymbol{\varphi_i}(z)\}$, as demonstrated in subsection III-A. When $\{\boldsymbol{\lambda_i}\}$ and $\{\boldsymbol{\varphi_i}(z)\}$ are sorted according to $\{\boldsymbol{\lambda_i}\}$ in a decreasing order, that is, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda M$, the PDE equation (5)'s resolution $\boldsymbol{y}(z, t)$ can be roughly expressed as

$$\hat{\boldsymbol{y}}(z,t) \triangleq \sum_{i=1}^{M} x_i(t)\boldsymbol{\varphi}_i(z). \qquad (11)$$

An M-dimensional ODE system is obtained by substituting (11) for y(z,t) in equation (5) and applying EEFs to both sides for the expression to accomplish the center composition.

$$\dot{\boldsymbol{x}} = \tilde{\boldsymbol{A}}(\boldsymbol{x}) + \tilde{\boldsymbol{f}}(\boldsymbol{x}) + \tilde{\boldsymbol{B}}\boldsymbol{u}$$
$$\boldsymbol{x}(0) = \boldsymbol{x}_0 \qquad (12)$$

where $\boldsymbol{x} \triangleq [x_1 \cdots x_M]^{\mathsf{T}}$, $\tilde{\boldsymbol{A}}(\boldsymbol{x}) \triangleq \langle \boldsymbol{A}(\hat{\boldsymbol{y}}), \Phi(z)\rangle$, $\tilde{\boldsymbol{f}}(\boldsymbol{x}) \triangleq \langle \boldsymbol{f}(\hat{\boldsymbol{y}}), \Phi(z)\rangle$, $\tilde{\boldsymbol{B}} \triangleq \langle \boldsymbol{B}(z), \Phi(z)\rangle$, and $\boldsymbol{x}_0 \triangleq \langle \hat{\boldsymbol{y}}_0(z), \Phi(z)\rangle$, with $\boldsymbol{\Phi}(z) \triangleq [\boldsymbol{\varphi}_1(z) \cdots \boldsymbol{\varphi}_M(z)]^{\mathsf{T}}$. From [20, Proposition 1],

The disparity among the PDE equation (5)'s resolution $\boldsymbol{y}(z, t)$ furthermore it is demonstrated that the solution $\boldsymbol{y}\hat{}(z, t)$ [derived by (11) and (12)] satisfies $\rho y(z, t) - \hat{y}(z, t)\rho 2 = \mu(M)$, where $\mu(M)$) is a small optimistic actual number that depends on M. additionally $\lim_{M\to\infty} \mu(M) = 0$. This can be shown from [20, Claim 1]. Owing to the ODE technique's (12) large dimension, a greater dimension lowering is necessary. At this equilibrium point, or origin, of interest, a linearization is performed for (12). After that, (12) is the same recast as

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \tilde{\boldsymbol{B}}\boldsymbol{u} + \tilde{g}(\boldsymbol{x})$$
$$\boldsymbol{x}(0) = \boldsymbol{x}_0 \qquad (13)$$

where $\boldsymbol{A} \triangleq \partial[\tilde{\boldsymbol{A}}(\boldsymbol{x}) + \tilde{\boldsymbol{f}}(\boldsymbol{x})]/\partial \boldsymbol{x}|_{\boldsymbol{x}=0}$ and $\tilde{g}(\boldsymbol{x}) \triangleq \tilde{\boldsymbol{A}}(\boldsymbol{x}) + \tilde{\boldsymbol{f}}(\boldsymbol{x}) - \boldsymbol{A}\boldsymbol{x}$.

In this instance, we recast (13) as a connected ODE systems with slow and fast subsystems and matching measures of N and M N.

$$\begin{cases} \dot{\boldsymbol{x}}_s = \boldsymbol{A}_s\boldsymbol{x}_s + \boldsymbol{A}_{sf}\boldsymbol{x}_f + \boldsymbol{B}_s\boldsymbol{u} + \tilde{g}_s(\boldsymbol{x}_s, \boldsymbol{x}_f) \\ \dot{\boldsymbol{x}}_f = \boldsymbol{A}_{fs}\boldsymbol{x}_s + \boldsymbol{A}_f\boldsymbol{x}_f + \boldsymbol{B}_f\boldsymbol{u} + \tilde{g}_f(\boldsymbol{x}_s, \boldsymbol{x}_f) \\ \boldsymbol{x}_s(0) = \boldsymbol{x}_{s0} \\ \boldsymbol{x}_f(0) = \boldsymbol{x}_{f0} \end{cases} \qquad (14)$$

where $\boldsymbol{x}_s \triangleq [x_1 \cdots x_N]^{\mathsf{T}} \in \mathrm{R}^N$, $\boldsymbol{x}_f \triangleq [x_{N+1} \cdots x_M]^{\mathsf{T}} \in \mathrm{R}^{M-N}$, $\tilde{\boldsymbol{g}}_s \triangleq [\tilde{g}_1 \cdots \tilde{g}_N]^{\mathsf{T}} \in \mathrm{R}^N$, $\tilde{\boldsymbol{g}}_f \triangleq [\tilde{g}_{N+1} \cdots \tilde{g}_M]^{\mathsf{T}} \in \mathrm{R}^{M-N}$, $[\boldsymbol{x}_s^{\mathsf{T}}\ \boldsymbol{x}_f^{\mathsf{T}}] = \boldsymbol{x}^{\mathsf{T}}$, $[\tilde{\boldsymbol{g}}_s^{\mathsf{T}}\ \tilde{\boldsymbol{g}}_f^{\mathsf{T}}] = \tilde{\boldsymbol{g}}^{\mathsf{T}}$, $[\boldsymbol{x}_{s0}^{\mathsf{T}}\ \boldsymbol{x}_{f0}^{\mathsf{T}}] = \boldsymbol{x}_0^{\mathsf{T}}$, and $\boldsymbol{A}_s$, $\boldsymbol{A}_{sf}$, $\boldsymbol{A}_{fs}$, $\boldsymbol{A}_f$, $\boldsymbol{B}_s$, and $\boldsymbol{B}_f$ are block matrices of

Suitable proportions that fulfil $\begin{matrix} \boldsymbol{A}_s & \boldsymbol{A}_{sf} \\ \boldsymbol{A}_{fs} & \boldsymbol{A}_f \end{matrix} = \boldsymbol{A}$ and

$\begin{matrix} \boldsymbol{B}_s \\ \boldsymbol{B}_f \end{matrix} = \tilde{\boldsymbol{B}}$. The xs subsystem's depth, or N, needs to be selected so that it fulfils

$$\sum_{i=1}^{N} \lambda_i / \sum_{j=1}^{M} \lambda_j \geq 1 - \zeta \qquad (15)$$

Regarding a modest positive real number ζ. Assume that σ($\boldsymbol{A}_s$) = {1, 2,..., N } symbolises As's ganz range, while $\sigma(\boldsymbol{A}_f)$ = 1, 2,..., $_{M-N}$ represents the eigen spectrum of $\boldsymbol{A}_f$. Then, our theories regarding the distinct attribute of (14) are stated in our presumption that follows.

Addressing

{˜1} ≥ Re{2} ≥ ··· ≥ Re{N } and which{1} ≥ which{2} ≥ ··· ≥ Re{ˆM −N }, where \j is the imaginary part. This is the first postulate.

A tiny positive value, ε = |Re{\1}|/|Re{\1}| < 1, is present. The order-of-magnitude nomenclature in this case is O(ε) [47]; which means that ς(ε) = O(ε) if the real numbers k1 and k2 exist with respect to ς(ε)<k1 ε, ε<k2).

Regarding a modest positive real number ζ. Assume that σ($\boldsymbol{A}_s$) = {1, 2,..., N } represents the eigen spectrum associated with As, while σ($\boldsymbol{Af}$) = 1, 2,..., M −N represents the eigen rainbow of $\boldsymbol{Af}$. Then, our theories regarding the distinct attribute of (14) are stated in the presumption that follows.

Assumption 1:

1) $\mathrm{Re}\{\tilde{\lambda}_1\} \geq \mathrm{Re}\{\tilde{\lambda}_2\} \geq \cdots \geq \mathrm{Re}\{\tilde{\lambda}_N\}$ and $\mathrm{Re}\{\hat{\lambda}_1\} \geq \mathrm{Re}\{\hat{\lambda}_2\} \geq \cdots \geq \mathrm{Re}\{\hat{\lambda}_{M-N}\}$, where $\mathrm{Re}\{\tilde{\lambda}_j\}$ denotes the real part of $\tilde{\lambda}_j$.
2) $\mathrm{Re}\{\hat{\lambda}_1\} < 0$ and $|\mathrm{Re}\{\tilde{\lambda}_N\}|/|\mathrm{Re}\{\hat{\lambda}_1\}| = O(\varepsilon)$, where

A tiny positive value, ε = |Re{\1}|/|Re{\1}| < 1, is present. The order-of-magnitude designation in this case is O(ε) [47]; namely, ς(ε) = O(ε) if positive real numbers k1 and k_2 exist with respect to ς(ε)<k_1 ε, ε<k_2).

Remark 2: The parabolic PDE systems with nonlinear SDOs have the fast/slow dissolution feature. [20], [21], and basic SDOs [48]. This characteristic is stated using Assumption 1, which is shown here. For hyperbolic PDE frameworks, the criteria in Assumption 1 are frequently met by appropriately selecting constants *N* and *M*.

A typical SP counterpart of the xf module may be obtained by multiplying both sides by ε (14) that is equal to what is presented in Proposition 1.

$$\begin{cases} \dot{\boldsymbol{x}}_s = \boldsymbol{A}_s\boldsymbol{x}_s + \boldsymbol{B}_s\boldsymbol{u} + \boldsymbol{g}_s(\boldsymbol{x}_s, \boldsymbol{x}_f) \\ \varepsilon\dot{\boldsymbol{x}}_f = \boldsymbol{A}_{f\varepsilon}\boldsymbol{x}_f + \varepsilon\boldsymbol{B}_f\boldsymbol{u} + \varepsilon\boldsymbol{g}_f(\boldsymbol{x}_s, \boldsymbol{x}_f) \\ \boldsymbol{x}_s(0) = \boldsymbol{x}_{s0} \\ \boldsymbol{x}_f(0) = \boldsymbol{x}_{f0} \end{cases} \tag{16}$$

where $\boldsymbol{A}_{f\varepsilon} \triangleq \boldsymbol{\varepsilon A}_f$, $\boldsymbol{g}_s(\boldsymbol{x}_s, \boldsymbol{x}_f) \triangleq \boldsymbol{A}_{sf}\boldsymbol{x}_f + \tilde{\boldsymbol{g}}\ (\boldsymbol{x}_s, \boldsymbol{x}_f)$, and $\boldsymbol{g}_f(\boldsymbol{x}_s, \boldsymbol{x}_f) \triangleq \boldsymbol{A}_{fs}\boldsymbol{x}_s + \tilde{\boldsymbol{g}}_f(\boldsymbol{x}_s, \boldsymbol{x}_f)$.

Introducing a fast time scale $\tau = t/\varepsilon$ and setting $\varepsilon = 0$, then, the $\boldsymbol{x}_f$ subsystem in (16) is

$$\frac{\partial \boldsymbol{x}_f}{\partial \tau} = \boldsymbol{A}_{f\varepsilon}\boldsymbol{x}_f. \tag{17}$$

System (17) is globally exponentially stable based on the concept of ε and the fact that Re 1 < 0. Consequently, $\boldsymbol{x_f} = 0$. The ROM is obtained as (16).

$$\begin{aligned} \dot{\boldsymbol{x}}_s &= \boldsymbol{A}_s\boldsymbol{x}_s + \boldsymbol{B}_s\boldsymbol{u} + \boldsymbol{g}_s(\boldsymbol{x}_s) \\ \boldsymbol{x}_s(0) &= \boldsymbol{x}_{s0} \end{aligned} \qquad (18)$$

In which case gs (xs) = gs (xs, 0).

Remark 3: The approach used to get the ROM here differs from the approach described in [20]. The linked fast/slow ODE structure (14) is derived in [20] via the coordinate change, where $\boldsymbol{A}_s$ and $\boldsymbol{A}_f$ are orthogonal matrices and the terms $\boldsymbol{A}_{sf}\,xf$ and $\boldsymbol{A}_{fs}\boldsymbol{x}_s$ are miss-identified.

Iv. Synthesis of Approximate Optimal Controller

This section will create an estimated optimum controller using NN for the PDE system (5) using the HJB theoretical structure, based around the ROM (18).

A. HJB Approach to Optimal Controller Synthesis

Now, let's examine a generic effectiveness functionality.

$$V\left(\boldsymbol{x}_{s0}\right) = \int_0^{\infty} Q(\boldsymbol{x}_s) + \boldsymbol{u}^{\top}\boldsymbol{R}\boldsymbol{u}\ \ dt \qquad (19)$$

In this case, $Q\ (\boldsymbol{x}_s)$ is a positive definite function; that is, if xs = 0, then Q ($\boldsymbol{x}_s$) > 0; Q ($\boldsymbol{x}_s$) = 0 only in that case. R is a subset of Rp×p. When ∀xs = 0, $Q\ (\boldsymbol{x}_s) > 0$; $Q\ (\boldsymbol{x}_s) = 0$ only when $\boldsymbol{x}_s = 0$. Here, $Q\ (\boldsymbol{x}_s)$ is an optimistic continuous function. A positive definite matrix is denoted by R ∈ Rp×p. When $\boldsymbol{x}_s = 0$, V ($\boldsymbol{x}_s$) > 0; that is, V($\boldsymbol{x}_s$) = 0 only in the event $\boldsymbol{x}_s = 0$. For xs = Ω − Rn, V ($\boldsymbol{x}_s$) − C1(Ω) is a positive definite function. We present the concept of acceptable control as follows.

Criterion 1[37], [38] (acceptable Control): Given system (18), with $\boldsymbol{x}_s$ ©, a control variable u($\boldsymbol{x}_s$): Ω · R^p is determined to be acceptable with regard to (19) on Ω, represented by u($\boldsymbol{x}_s$) (Ω), provided that the following requirements are met: First, u is continual on Ω; second, u(0) = 0; third, u($\boldsymbol{x}_s$) stabilises the system (18); and fourth, V ($\boldsymbol{x}_s$0) <, xs0 `.

Our goal is to minimise the performance functional (19) in order to identify an acceptable control $\boldsymbol{u}*(\boldsymbol{x}_s)$ (Ω). It is widely acknowledged that the problems that follow may be correspondingly solved by solving the optimum control issue regarding the system in question. HJB

$$\inf_{\boldsymbol{u} \in \mathrm{U}(\Omega)} H(\boldsymbol{u}) = 0 \qquad (20)$$

with the Bernoulli component denoted by $H(\boldsymbol{u})$.

$$H\ (\boldsymbol{u}) =^{\Delta}\ V*\ \ \mathrm{T}\ (\boldsymbol{A}\ \ \boldsymbol{x}\ \ \ + \boldsymbol{B}\ \ \boldsymbol{u} + \boldsymbol{g}\ \ ) + Q + \boldsymbol{u}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}$$

in which the partial derivative exists about x is indicated by the exponent $\boldsymbol{x}_s$, meaning that V ∗ =∆ ∂V ∗/∂**x**, and V ∗ is the magnitude of the variable. The initial-order required condition of optimality, which is satisfied by the best possible control u∗, is ∂H(u)/∂u u=u∗ = 0. As a result, we have Since u∗, the best command, violates ∂H(**u**)/∂**u** u=u∗ = 0, the situation is

$$\boldsymbol{u}^* = -\frac{1}{2}\boldsymbol{R}^{-1}\boldsymbol{B}_s^{\mathsf{T}} V^*_{\boldsymbol{x}_s}. \qquad (21)$$

Conversely, the related expense variable V for an uncontrolled parameter u (Ω) satisfies the following GHJB equation [37]–[39]:

$$(V_{\boldsymbol{x}_s})^{\mathsf{T}}(\boldsymbol{A}_s\boldsymbol{x}_s + \boldsymbol{B}_s\boldsymbol{u} + \boldsymbol{g}_s) + Q + \boldsymbol{u}^{\mathsf{T}}\boldsymbol{R}\boldsymbol{u} = 0. \qquad (22)$$

Substituting $\boldsymbol{u}^*$ into (22) yields

$$V^{*\mathsf{T}}_{\boldsymbol{x}_s}(\boldsymbol{A}_s\boldsymbol{x}_s + \boldsymbol{B}_s\boldsymbol{u}^* + \boldsymbol{g}_s) + Q + \boldsymbol{u}^{*\mathsf{T}}\boldsymbol{R}\boldsymbol{u}^* = 0. \qquad (23)$$

(18) Selecting V ∗ as the potential Lyapunov product for the system as a whole (18), we are left with

$$\dot{V}^* = V^{*\mathsf{T}}_{\boldsymbol{x}_s}(\boldsymbol{A}_s\boldsymbol{x}_s + \boldsymbol{B}_s\boldsymbol{u}^* + \boldsymbol{g}_s) = -Q - \boldsymbol{u}^{*\mathsf{T}}\boldsymbol{R}\boldsymbol{u}^* < 0. \qquad (24)$$

This implies that the closed-loop system of (18) with optimal control is differently stable.

$\boldsymbol{u}*$. The HJB equation (20) may be rewritten as follows from (21) and (23).

$$V^{*\mathsf{T}}_{\boldsymbol{x}_s}(\boldsymbol{A}_s\boldsymbol{x}_s + \boldsymbol{g}_s) + Q(\boldsymbol{x}_s) - (1/4)\, V^{*\mathsf{T}}_{\boldsymbol{x}_s}\boldsymbol{B}_s\boldsymbol{R}^{-1}\boldsymbol{B}_s^{\mathsf{T}} V^*_{\boldsymbol{x}_s} = 0. \qquad (25)$$

Thus, the HJB equation (25) for $V*$ may be solved by reducing the issue of constructing an optimum control (21) to that of solving it.

The two components of the dynamics of the system (18) are the negative term $\boldsymbol{g}_s$ and the linear portion $\boldsymbol{A}_s\boldsymbol{x}_s$, as we can see. In this research, we suggest an optimum microcontroller synthesizing approach employing the following modal feedback control rule to fully use the quadratic part:

$$\boldsymbol{u}(\boldsymbol{x}_s) = \boldsymbol{u}_1(\boldsymbol{x}_s) + \boldsymbol{u}_2(\boldsymbol{x}_s). \qquad (26)$$

Let us examine a few different versions of the monetary value V ($\boldsymbol{x}_s$) and the state penalty function $Q(\boldsymbol{x}_s)$:

$$Q(\boldsymbol{x}_s) = \boldsymbol{x}_s^{\mathsf{T}}\boldsymbol{Q}_1\boldsymbol{x}_s + \tilde{Q}(\boldsymbol{x}_s) \qquad (27)$$

$$V(\boldsymbol{x}_s) = \boldsymbol{x}_s^{\mathsf{T}}\boldsymbol{P}\boldsymbol{x}_s + \tilde{V}(\boldsymbol{x}_s) \qquad (28)$$

where $\boldsymbol{Q}_1 > 0 \in \mathrm{R}^{n\times n}$, $\tilde{V}(\boldsymbol{x}_s) \in C^1(\Omega)$, and $\boldsymbol{P} > 0 \in \mathrm{R}^{n\times n}$ is the solution to the following algebraic Riccati equation (ARE):

$$\boldsymbol{A}_s^{\mathsf{T}}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A}_s + \boldsymbol{Q}_1 - \boldsymbol{P}\boldsymbol{B}_s\boldsymbol{R}^{-1}\boldsymbol{B}_s^{\mathsf{T}}\boldsymbol{P} = 0. \qquad (29)$$

The optimum controller may be expressed similar as (28) + (21), where

$$\boldsymbol{u}^* = \boldsymbol{u}_1^* + \boldsymbol{u}_2^* \qquad (30)$$

$$\boldsymbol{u}_1^* \triangleq -\boldsymbol{R}^{-1}\boldsymbol{B}_s^{\top}\boldsymbol{P}\boldsymbol{x}_s \tag{31}$$

$$\boldsymbol{u}_2^* \triangleq -\frac{1}{2}\boldsymbol{R}^{-1}\boldsymbol{B}_s^{\top}\tilde{V}^*_{\boldsymbol{x}_s} \tag{32}$$

$\boldsymbol{P}$ and $\hat{V}^*$ have yet to be ascertained. It follows from (22) that a novel kind of GHJB-like formula is provided by employing (26)–(29).

$$(\tilde{V}_{\boldsymbol{x}_s})^{\top}(\bar{\boldsymbol{A}}_s + \boldsymbol{B}_s\boldsymbol{u}_2) + \bar{\boldsymbol{Q}} + \boldsymbol{u}_2^{\top}\boldsymbol{R}\boldsymbol{u}_2 = 0 \tag{33}$$

where $\bar{\boldsymbol{A}}_s \triangleq \boldsymbol{A}_s\boldsymbol{x}_s + \boldsymbol{B}_s\boldsymbol{R}^{-1}\boldsymbol{B}_s^{\top}\boldsymbol{P}\boldsymbol{x}_s + \boldsymbol{g}_s$ and $\bar{\boldsymbol{Q}} \triangleq \tilde{\boldsymbol{Q}} + 2\boldsymbol{x}_s^{\top}\boldsymbol{P}\boldsymbol{g}_s$. Replacing $\boldsymbol{u}_2$ in (33) with $\boldsymbol{u}_2^*$ that is given in (32) yields a new type of HJB-like equation

$$\left(\tilde{V}^*_{\boldsymbol{x}_s}\right)^{\top}\bar{\boldsymbol{A}}_s + \bar{\boldsymbol{Q}} - (1/4)\left(\tilde{V}^*_{\boldsymbol{x}_s}\right)^{\top}\boldsymbol{B}_s\boldsymbol{R}^{-1}\boldsymbol{B}_s^{\top}\tilde{V}^*_{\boldsymbol{x}_s} = 0. \tag{34}$$

[From (24)] it is evident that the synthesized ideal system for control (30) [which is equivalent to (21)] can both maximize velocity across the ROM and ensure the ROM's maximal closed-loop applications stability (18). Additionally, (12)'s closed-loop architecture is highly stable, as demonstrated by **Theorem 1.**

In the case of the nonlinear parabolic PDE system (5), the first assumption is false is true, is the subject of Theorem 1. Next, if xs0 \1, xf0 \2, and ε (*0, ε∗*), thus, in an identical manner, the optimal operator (30) [or (21) may guarantee that the closed-loop system of (12) is asymmetrically stable. This is because there are positive real values $\delta 1$, $\delta 2$, and $\varepsilon*$.

Note that the best controller can make sure the system stays stable over time. This is because the solution obtained through a specific equation will keep getting closer to a specific value.- Additionally, [20, Proposition 1] shows that the PDE scheme (5)'s resolution $y(z, t)$ will approach the solution $\hat{\boldsymbol{y}}(z, t)$ as M grows. Actually, because of the curved shape of the SDO, we can accurately describe the main way parabolic PDE systems behave using just a few patterns. Several studies have shown that a very small M can give good results, meaning that $\hat{\boldsymbol{y}}(z, t)$ and $\boldsymbol{y}(z, t)$ behave similarly. The top control system for ROM (18) can achieve results that are almost as good as the original PDE system (5).

V. Conclusion

Within the context of HJB theories, an approximation optimum control system has been synthesized in this research for a group of parabolic PDE solutions that are nonlinear. The prevailing dynamics of the PDE system may be accurately characterised by the ROM that is created using the data-based KLD method and SP technique, because the SDO of the PDE component is nonlinear. The high-order ODE system's closed-loop systems terminal stability is ensured, and an estimated optimum controller is offered by two sections to fully use the linear portion of the ROM. The linear portion of the ROM, which is acquired by solving and ARE, is the target of the first section of the synthesized controller. Conversely, the latter portion is which NN is used in the cost function estimation process. Ultimately, the case the investigation's stochastic diffusion–reaction processes is used to demonstrate the effectiveness of the mechanism of control that was developed, according to the simulation findings.

Refrences

[1] F. L. Lewis and V. L. Syrmos, Optimal Control. New York: Wiley, 1995.

[2] D. P. Bertsekas, Dynamic Programming and Optimal Control, 3rd ed. Nashua, NH: Athena Scientific, 2005.

[3] A. G. Butkovskiy, Distributed Control Systems. New York: American Elsevier, 1969.

[4] P. K. C. Wang and F. Tung, "Optimum control of distributed parameter systems," J. Basic Eng., vol. 86, no. 1, pp. 67–79, Mar. 1964.

[5] J. L. Lions, Optimal Control of Systems Governed by Partial Differential Equations. New York: Springer-Verlag, 1971.

[6] H. O. Fattorini, Infinite Dimensional Optimization Theory and Optimal Control. Cambridge, U.K.: Cambridge Univ. Press, 1999.

[7] X. Li and J. Yong, Optimal Control Theory for Infinite Dimensional Systems. Boston, MA: Birkhäuser, 1995.

[8] R. F. Curtain and H. J. Zwart, An Introduction to Infinite-Dimensional Linear Systems Theory. New York: Springer-Verlag, 1995.

[9] I. Aksikas, A. Fuxman, J. F. Forbes, and J. J. Winkin, "LQ control design of a class of hyperbolic PDE systems: Application to fixed-bed reactor," Automatica, vol. 45, no. 6, pp. 1542–1548, Jun. 2009.

[10] I. Aksikas, J. J. Winkin, and D. Dochain, "Optimal LQ-feedback regula- tion of a nonisothermal plug flow reactor model by spectral factorization," IEEE Trans. Autom. Control, vol. 52, no. 7, pp. 1179–1193, Jul. 2007.

[11] W. H. Ray, Advanced Process Control. New York: McGraw-Hill, 1981.

[12] M. Li and P. D. Christofides, "An input/output approach to the optimal transition control of a class of distributed chemical reactors," Chem. Eng. Sci., vol. 62, no. 11, pp. 2979–2988, 2007.

[13] M. Li and P. D. Christofides, "Optimal control of diffusion–convection– reaction processes using reduced-order models," Comput. Chem. Eng., vol. 32, no. 9, pp. 2123–2135, Sep. 2008.

[14] C. Xu, Y. Ou, and E. Schuster, "Sequential linear quadratic control of bilinear parabolic PDEs based on POD model reduction," Automatica, vol. 47, no. 2, pp. 418–426, Feb. 2011.

[15] I. S. Sadek and M. A. Bokhari, "Optimal control of a parabolic distrib- uted parameter system via orthogonal polynomials," Optim. Control Appl. Methods, vol. 19, no. 3, pp. 205–213, May/Jun. 1998.

[16] S. S. Ravindran, "A reduced-order approach for optimal control of fluids using proper orthogonal decomposition," Int. J. Numer. Method Fluid, vol. 34, no. 5, pp. 425–448, Nov. 2000.

[17] V. Yadav, R. Padhi, and S. N. Balakrishnan, "Robust/optimal temperature profile control of a high-speed aerospace vehicle using neural networks," IEEE Trans. Neural Netw., vol. 18, no. 4, pp. 1115–1128, Jul. 2007.

[18] R. Padhi, S. N. Balakrishnan, and T. Randolph, "Adaptive-critic based optimal neuro control synthesis for distributed parameter systems," Automatica, vol. 37, no. 8, pp. 1223–1234, 2001.

[19] R. Padhi and S. N. Balakrishnan, "Proper orthogonal decomposition based optimal neurocontrol synthesis of a chemical reactor process using approximate dynamic programming," J. Neural Netw., vol. 16, no. 5/6, pp. 719–728, Jun. 2003.

[20] J. Baker and P. D. Christofides, “Finite-dimensional approximation and control of nonlinear parabolic PDE systems,” Int. J. Control, vol. 73, no. 5, pp. 439–456, 2000.

[21] J. Baker and P. D. Christofides, “Output feedback control of parabolic PDE systems with nonlinear spatial differential operators,” Ind. Eng. Chem. Res., vol. 38, no. 11, pp. 4372–4380, Nov. 1999.

[22] A. Armaou and P. D. Christofides, “Dynamic optimization of dissipative PDE systems using nonlinear order reduction,” Chem. Eng. Sci., vol. 57, no. 24, pp. 5083–5114, Dec. 2002.

[23] H.-X. Li and C. K. Qi, Spatio-Temporal Modeling of Nonlinear Dis- tributed Parameter Systems: A Time/Space Separation Based Approach. New York: Springer-Verlag, 2011.

[24] P. Holmes, J. L. Lumley, and G. Berkooz, Turbulence, Coherent Struc- tures, Dynamical Systems and Symmetry. Cambridge, U.K.: Cambridge Univ. Press, 1996.

[25] L. G. Bleris and M. V. Kothare, “Low-order empirical modeling of dis- tributed parameter systems using temporal and spatial eigenfunctions,” Comput. Chem. Eng., vol. 29, no. 4, pp. 817–827, Mar. 2005.

[26] D. H. Gay and W. H. Ray, “Identification and control of distributed parameter systems by means of the singular value decomposition,” Chem. Eng. Sci., vol. 50, no. 10, pp. 1519–1539, May 1995.

[27] D. G. Zheng, “System identification and model-based control for a class of distributed parameter systems,” Ph.D. dissertation, Texas Tech Univ., Lubbock, TX, 2003.

[28] B. Widrow, N. K. Gupta, and S. Maitra, “Punish/reward: Learning with a critic in adaptive threshold systems,” IEEE Trans. Syst., Man, Cybern., vol. SMC-3, no. 5, pp. 455–465, Sep. 1973.

[29] J. Seiffertt, S. Sanyal, and D. C. Wunsch, “Hamilton–Jacobi–Bellman equations and approximate dynamic programming on time scales,” IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 38, no. 4, pp. 918–923, Aug. 2008.

[30] H. Zhang, Q. Wei, and 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 Trans. Syst., Man, Cybern. B, Cybern., vol. 38, no. 4, pp. 937–942, Aug. 2008.

[31] A. Al-Tamimi, M. Abu-Khalaf, and F. L. Lewis, “Adaptive critic designs for discrete-time zero-sum games with application to H control,” IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 37, no. 1, pp. 240–247, Feb. 2007.

[32] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, “Discrete-time nonlin- ear HJB solution using approximate dynamic programming convergence proof,” IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 38, no. 4, pp. 943–949, Aug. 2008.

[33] F. L. Lewis and K. G. Vamvoudakis, “Reinforcement learning for partially observable dynamic processes: Adaptive dynamic programming using measured output data,” IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 41, no. 1, pp. 14–25, Feb. 2011.

[34] H. Zhang and Y. Luo, “Neural-network-based near-optimal control for a class of discrete-time affine nonlinear systems with control constraints,” IEEE Trans. Neural Netw., vol. 20, no. 9, pp. 1490–1503, Sep. 2009.