mathematics

MDPI

Article

Neural Networks Simulation of Distributed SEIR System †

Tibor Kmet [1,*], Maria Kmetova [2] and Ladislav Végh [1]

[1] Department of Informatics, J. Selye University, Bratislavská Cesta 3322, 945 01 Komárno, Slovakia; veghl@ujs.sk
[2] Department of Mathematics, Faculty of Natural Sciences and Informatics, Constantine the Philosopher University, Tr. A. Hlinku 1, 949 74 Nitra, Slovakia; mkmetova@ukf.sk
* Correspondence: kmett@ujs.sk
† This paper is an extension of our conference paper at the AlCoB 2016 Algorithms for Computational Biology conference held on 21–22 June 2016 in Trujillo, Spain.

Abstract: In this paper, a neural network-based optimal control synthesis is presented for distributed optimal control problems. We deal with solutions of systems controlled by parabolic differential equations with control and state constraints and discrete time delays. The given optimal control problem is transformed into a discrete nonlinear problem and then implemented into a feed-forward adaptive critic neural network. We propose a new algorithm to reach optimal control and an optimal trajectory using a feed-forward neural network. We present a concrete application of this simulation method on the SEIR (Susceptible—Exposed—Infectious—Recovered) optimal control problem of a distributed system for disease control. The results show that the adaptive-critic-based neural network approach is suitable for the solution of optimal distributed control problems with delay in state and control variables subject to control-state constraints and simulates the spread of disease in the SEIR system.

Keywords: feed-forward neural network; distributed control problem with discrete delays; state and control constraints; optimal control synthesis; complex systems

MSC: 68T07

check for updates

Citation: Kmet, T.; Kmetova, M.; Végh, L. Neural Networks Simulation of Distributed SEIR System. *Mathematics* **2023**, *11*, 2113. https://doi.org/10.3390/math11092113

Academic Editors: Juan Francisco Sánchez-Pérez, Gonzalo García Ros, Manuel Conesa and Xinsong Yang

Received: 2 March 2023
Revised: 26 April 2023
Accepted: 27 April 2023
Published: 29 April 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In recent decades, the application of nonlinear dynamic systems has proven to be useful in many fields, for example, computer networks, robotics, transportation systems, epidemiology, and biology. These nonlinear dynamic systems were extended with control variables to control the dynamic of the systems. The design of controllers for nonlinear dynamic systems has also drawn intensive attention [1]. Many results have been obtained for the control of dynamic systems using adaptive control and neural networks (see, e.g., [2–4]).

According to Marino [5] (p. 55), "Adaptive controls for nonlinear systems are defined as nonlinear dynamic (state or output feedback) compensators capable of guaranteeing asymptotic tracking of an output reference signal for any unknown parameter vector and any initial condition for the closed loop system when time varying disturbances are not present." Dynamic programming has been a useful technique to control nonlinear dynamic systems [6], where we want to attain the minimum or maximum of a given objective function, optimization criterion, or performance index representing the mathematical expression of a phenomenon.

The aim of our article, which is related to works [7–10], is to apply neural network methods in the study and simulation of nonlinear complex adaptive epidemiological systems, specifically a SEIR system. We consider an optimal distributed control problem for systems governed by parabolic differential equations with control and state constraints and a discrete time delay for the SEIR model, which contains the intensity of vaccination. There are some numerical methods based on necessary conditions of optimality [11,12], where the state and

co-state equations are solved forward and backward in time. In the last decade, efficient techniques have been developed [13–15] to construct learning systems for the approximate solution of optimal control problems. Characterized by self-learning and adaptability, artificial neural networks are also a serviceable tool to implement controlled learning [16–18]. "Multilayer feedforward networks with as few as one hidden layer are indeed capable of universal approximation in a very precise and satisfactory sense" [19] (p. 260). Therefore, feedforward networks are used in adaptive dynamic programming (ADP) algorithms or adaptive critic design (ACD), which was proposed by Werbos [17,18] to deal with optimal control problems forward in time.

The finite element approximation approach has been thoroughly studied in [20–22] and [12] for parabolic and elliptic optimal control problems. This approach plays an important role in the numerical treatment of optimal control problems. The characteristic of this solution is that the optimal control problem is translated into a finite-dimensional nonlinear programming problem (NLP-problem) by discretization. These methods, basically, apply nonlinear programming techniques to the resulting finite-dimensional optimization problem, as one can see in [23,24] and [12]. Thereafter, neural networks are used as a universal approximation to solve finite-dimensional optimization problems forward in time with ACD [18,25–27].

In this paper, we use an adaptive critic neural network architecture based on [7] for two optimal distributed control problems. In Section 2, we introduce an optimal control problem for a parabolic differential equation where state and control variables are delayed in time and also a control problem with control and state constraints. In Section 3, we transform the initial continuous problem into a nonlinear programming problem using discretization. Section 4 presents a new algorithm using adaptive critic neural network design, where the adaptive neural network approximates optimal control and critic network co-state variables. Section 5 presents a short description of the SEIR model and investigates the optimal vaccine coverage threshold needed for disease control and eradication as the optimal control problem subject to control and state constraints. For the numerical experiments described in Section 5, Matlab 2021b software was used.

2. The Optimal Control Problem

We consider the nonlinear control problem governed by parabolic equations with delays in state and control variables for mixed control-state constraints. We denote $x(p,t) \in R^n$ and $u(p,t) \in R^m$ the state and control variable, respectively, in a given space-time domain $Q = [a,b] \times \left[t_0, t_f\right]$, where the final time t_f is fixed. The optimal control problem is to minimize

$$\mathcal{J}(u) = \int_a^b g(x(p,t_f))dp + \int_a^b \int_{t_0}^{t_f} f_0(x(p,t), x(p,t-\tau_x), u(p,t), u(p,t-\tau_u))dtdp, \quad (1)$$

subject to

$$\frac{\partial x(p,t)}{\partial t} = D\frac{\partial^2 x(p,t)}{\partial p^2} + f(x(p,t), x(p,t-\tau_x), u(p,t), u(p,t-\tau_u)), \quad (2)$$

$$\frac{\partial x(a,t)}{\partial p} = \frac{\partial x(b,t)}{\partial p} = 0, t \in [t_0, t_f],$$

$$x(p,t) = \phi_s(p,t), u(p,t) = \phi_c(p,t), p \in [a,b], t \in [t_0 - \tau_u, t_0],$$

$$\psi(x(p,t_f)) = 0, c(x(p,t), u(p,t)) \leq 0, p \in [a,b], t \in [t_0, t_f].$$

where $c(x,u)$ formulates a mixed state-control constraint, $\tau_x \geq 0$ and $\tau_u \geq 0$ are discrete time delays in the state and control variable, respectively, and ψ is a given terminal condition for state variable x. D is a diffusion matrix. The functions $g : R^n \to R$, $f_0 : R^{2(n+m)} \to R$,

$f : R^{2(n+m)} \to R^n$, $c : R^{n+m} \to R^q$ and $\psi : R^n \to R^r$, $0 \le r \le n$ are assumed to be sufficiently smooth on appropriate open sets, and the initial conditions $\phi_s(p,t), \phi_c(p,t)$ are continuous functions. Dynamic optimization objective functional $J(u)$ has Bolza form with scalar terminal cost and Equation (2) describes a dynamic of state variables. The theory of necessary conditions for the optimal control problem of form (1) is well developed, see, e.g., [24,28]. To simplify the notation, we introduce an additional state variable $x_0(p,t) = \int_0^t f_0(x(p,s), x(p,s-\tau_x), u(p,s), u(p,s-\tau_u)ds$. Then the augmented Hamiltonian function for problem (1) is

$$\mathcal{H}(x, x_{\tau_x}, u, u_{\tau_u}, \lambda, \mu) = \sum_{j=0}^{n} \lambda_j f_j(x, x_{\tau_x}, u, u_{\tau_u}) + \sum_{j=0}^{q} \mu_j c_j(x,u),$$

where $\lambda \in R^{n+1}$ is the adjoint variable and $\mu \in R^q$ is a multiplier associated to the inequality constraints. Assume that $\tau_x, \tau_u \ge (0, \tau_x, \tau_u) \ne (0,0)$ and $\frac{\tau_x}{\tau_u} \in Q$ for $\tau_u > 0$ or $\frac{\tau_u}{\tau_x} \in Q$ for $\tau_x > 0$. Let $(\hat{x}, \hat{u})$ be an optimal solution for (1). Then the necessary optimality condition for (1) implies [24,28] that there exist a piecewise continuous and piecewise continuously differentiable adjoint function $\lambda : Q \to R^{n+1}$, a piecewise continuous multiplier function $\mu : Q \to R^q$, $\hat{\mu}(p,t) \ge 0$ and a multiplier $\sigma \in R^r$ satisfying

$$\frac{\partial \lambda}{\partial t} = D\frac{\partial^2 \lambda}{\partial p^2} - \frac{\partial \mathcal{H}}{\partial x}(\hat{x}, \hat{x}_{\tau_x}, \hat{u}, \hat{u}_{\tau_u}, \lambda, \mu) - \chi_{[t_0, t_f - \tau_x]} \frac{\partial \mathcal{H}}{\partial x_{\tau_x}}(\hat{x}_{+\tau_x}, \hat{x}, \hat{u}_{+\tau_x}, \hat{u}_{\tau_u + \tau_x}, \lambda_{+\tau_x}, \mu_{+\tau_x}), \tag{3}$$

$$\lambda(p, t_f) = g_x(\hat{x}(p, t_f)) + \sigma\psi_x(\hat{x}(p, t_f)), \quad \frac{\partial \lambda(a,t)}{\partial p} = \frac{\partial \lambda(b,t)}{\partial p} = 0, \tag{4}$$

$$0 = -\frac{\partial \mathcal{H}}{\partial u}(\hat{x}, \hat{x}_{\tau_x}, \hat{u}, \hat{u}_{\tau_u}, \lambda, \mu) - \chi_{[t_0, t_f - \tau_u]} \frac{\partial \mathcal{H}}{\partial u_{\tau_u}}(\hat{x}_{+\tau_u}, \hat{x}_{\tau_x + \tau_u}, \hat{u}_{+\tau_u}, \hat{u}, \lambda_{+\tau_u}, \mu_{+\tau_u}). \tag{5}$$

In addition, the complementary conditions hold, i.e., in $p \in [a,b], t \in [t_0, t_f]$, $\mu(p,t) \ge 0$, $c(x(p,t), u(p,t)) \le 0$, and $\mu(p,t)c(x(p,t), u(p,t)) = 0$. The subscript x, x_{τ_x}, u and u_{τ_u} denote the partial derivative with respect to x, x_{τ_x}, u and u_{τ_u}, respectively and $x_{+\tau_x} = x(p, t+\tau_x)$, $x_{\tau_x + \tau_u} = x(p, t - \tau_x + \tau_u)$. Numerical methods that solve the challenges of obtaining an optimal control $\hat{u}$ and optimal trajectory $\hat{x}$ are described in [29,30]. The first is based on solving the optimal system (2, 3) which consists of $2n$ partial differential equations and boundary conditions in time also. The second method starts with initial guess for the adjoint state given by Equation (3). We solve Equation (6) and then state Equation (2). Then, those control values u and state values x are used to solve the adjoint state λ by backward method in time. We proposed a new method to solve the distributed optimal control problem (1) using adaptive critic neural networks, where state variable x and adjoint variable λ are solved forward in time.

3. Discretization of the Distributed Optimal Control Problem

In this section, we develop discretization techniques to transform the distributed control problem (1) into a nonlinear programming problem (see also in [11,12,20]). We suppose that $\tau_u = l\frac{\tau_x}{k}$ with $l, k \in N$. Defining $h_{max} = \frac{\tau_x}{k}$ gives the maximum interval length for an elementary transformation interval that satisfies $\frac{\tau_x}{h_{max}} = k \in N$ and $\frac{\tau_u}{h_{max}} = l \in N$. The minimum number of grid points for an equidistant discretization mesh is $N_{min} = \frac{t_f - t_0}{h_{max}}$. Let $K \in N$, and set $N = KN_{min}$. Let $t_j \in \langle t_0, t_f \rangle$, $j = 0, \ldots, M$, be an equidistant grid point with $t_j = t_0 + jh_t$, $j = 0, \ldots, M$, where $h_t = \frac{b-a}{M}$ is a time step and $t_f = Nh + t_0$. Let us suppose that the rectangle $R = \{(p,t) : a \le p \le b, t_0 \le t \le t_f\}$ is subdivided into N by M rectangles, each with sides h_t and $h_s = \frac{b-a}{M}$. For the bottom row, we have $t = t_0$, and the solution is $x(p_i, t_0) = \phi_s(p_i, t_0)$. For the grid points in successive rows, we have

$\{x(p_i, t_j) : i = 0, 1, \ldots, N,\ j = 0, 1, \ldots, M\}$ for the approximations to $x(p, t)$. The difference formulae for $x_t(p, t)$, $x_p(p, t)$ and $x_{pp}(p, t)$ are

$$x_t(p, t) \approx \frac{x(p, t + h_t) - x(p, t)}{h_t},$$

$$x_p(p, t) \approx \frac{x(p + h_s, t) - x(p, t)}{h_s} \tag{6}$$

And

$$x_{pp}(p, t) \approx \frac{x(p - h_s, t) - 2x(p, t) + x(p + h_s, t)}{h_s^2}. \tag{7}$$

Using the approximation $x_{i,j}$ in Equations (6) and (7), which are substituted into Equation (2), we obtain that

$$\frac{x_{i,j+1} - x_{i,j}}{h_t} = D\frac{x_{i-1,j} - 2x_{i,j} + x_{i+1,j}}{h_s^2} + f_{i,j}. \tag{8}$$

Equation (8) is applied to create the $(j + 1)$th row across the grid, assuming that approximations in the jth row are known. Let the vectors $x_{ij} \in R^n$, $u_{ij} \in R^m$, $i = 0, \ldots, N$, $j = 0, \ldots, M$, be an approximation of the state variable and control variable $x(p_i, t_j)$, $u(p_i, t_j)$, respectively at the mesh point (p_i, t_j). $z := ((x_{ij}), (u_{ij}), i = 0, \ldots, N, j = 0, \ldots, M) \in R^{N_s}$, $N_s = (n + m)NM$. Instead of the optimal control problem, here we use the following discretised control problem in the form of nonlinear programming problem with inequality constraints: Minimise

$$\mathcal{J}(z) = h_s h_t \sum_{(i,j)} f_0(x_{ij}, x_{\tau_x ij}, u_{ij}, u_{\tau_u ij}) + h_s \sum_{(i)} g(x_{iM}) \tag{9}$$

subject to

$$x_{i,j+1} = x_{ij} + h_t D\frac{x_{i-1,j} - 2x_{ij} + x_{i+1,j}}{h_s^2} + h_t f_{ij}, \tag{10}$$

$$x_{0j} = x_{1j}, x_{Nj} = x_{N-1j},\ i = 0, \ldots, N,\ j = 0, \ldots, M - 1,$$

$$x_{i,-j} = \phi_x(p_i, t_0 - jh), j = k, \ldots, 0, u_{i,-j} = \phi_u(p_i, t_0 - jh), j = l, \ldots, 0,$$

$$\psi(x_{i,N}) = 0, c(x_{ij}, u_{ij}) \leq 0,\ i = 0, \ldots, N,\ j = 0, \ldots, M - 1,$$

Our goal is to find an admissible control in a discrete-time formulation which minimises objective function (9). The Lagrangian function for the nonlinear optimisation problem (9) is given by Equation (12).

$$\mathcal{L}(z, \lambda, \sigma, \mu) = h_s h_t \sum_{(i,j)} f_0(x_{ij}, x_{\tau_x ij}, u_{ij}, u_{\tau_u ij}) + \tag{11}$$

$$\sum_{(i,j)} \lambda_{i,j+1}(-x_{i,j+1} + x_{ij} + D\frac{x_{i-1,j} - 2x_{ij} + x_{i+1,j}}{h_s^2} + h_t f_{ij}) + \sum_{(i,j)} \mu_{ij} c(x_{ij}, u_{ij}) +$$

$$\sum_{(i)} \sigma_i \psi(x_{iN}) + h_s \sum_{(i)} g(x_{iM}) + \sum_{(j)} \lambda_{0,j}\frac{x_{0,j} - x_{1,j}}{h_s} + \sum_{(j)} \lambda_{N,j}\frac{x_{N,j} - x_{N-1,j}}{h_s},$$

The first order optimality Karush–Kuhn–Tucker conditions [31] (p. 199) for problem (9) are given by Equations (12)–(14).

$$\begin{aligned} 0 = \mathcal{L}_{x_{ij}}(z,\lambda,\sigma,\mu) = \lambda_{i,j+1} - \lambda_{ij} + h_t D\tfrac{\lambda_{i-1,j}-2\lambda_{ij}+\lambda_{i+1,j}}{h_s^2} + h_t\lambda_{i,j+1} f_{x_{ij}}(x_{ij}, x_{i,j-k}, u_{ij}, u_{i,j-l}) + \\ h_t\lambda_{i,j+k+1} f_{x_{ij\tau_x}}(x_{i,j+k}, x_{ij}, u_{i,j+k}, u_{i,j-l+k}) + \mu_{ij} c_{x_{ij}}(x_{ij}, u_{ij}),\ j = 0,\dots,M-k-1, \\ 0 = \mathcal{L}_{x_{ij}}(z,\lambda,\sigma,\mu) = \lambda_{i,j+1} - \lambda_{ij} + h_t D\tfrac{\lambda_{i-1,j}-2\lambda_{ij}+\lambda_{i+1,j}}{h_s^2} + \\ h_t\lambda_{i,j+1} f_{x_{ij}}(x_{ij}, x_{i,j-k}, u_{ij}, u_{i,j-l}),\ j = M-k,\dots,M-1, \end{aligned} \tag{12}$$

$$0 = \mathcal{L}_{x_{iM}}(z,\lambda,\sigma,\mu) = g_{x_{iM}}(x_{im}) + \sigma_i \psi_{x_{iM}}(x_{iM}) - \lambda_{iM}, \lambda_{0j} = \lambda_{1j}, \lambda_{Nj} = \lambda_{N-1,j}, \tag{13}$$

$$\begin{aligned} 0 = \mathcal{L}_{u_{ij}}(z,\lambda,\sigma,\mu) = h_t\lambda_{i,j+1} f_{u_{ij}}(x_{ij}, x_{i,j-k}, u_{ij}, u_{i,j-l}) + \\ h_t\lambda_{i,j+l+1} f_{u_{ij\tau_u}}(x_{i,j+l}, x_{i,j-k+l}, u_{i,j+l}, u_{i,j}) + \mu_{ij} c_{u_{ij}}(x_{ij}, u_{ij}),\ j = 0,\dots,M-l-1, \end{aligned} \tag{14}$$

$$0 = \mathcal{L}_{u_{ij}}(z,\lambda,\sigma,\mu) = h_t\lambda_{i,j+1} f_{u_{ij}}(x_{ij}, x_{i,j-k}, u_{ij}, u_{i,j-l}) + \mu_{ij} c_{u_{ij}}(x_{ij}, u_{ij}),\ \ j = M-l,\dots,M-1.$$

These equations represent the discrete version of the necessary conditions (3)–(6) for the optimal control problem (1).

4. Adaptive Critic Neural Network for a Distributed Optimal Control Problem with Control and State Constraints

Neural networks are useful for approximation of the smooth time-invariant functions. According to [32] (p. 2), "experience has shown that optimisation of functionals over admissible sets of functions made up of linear combinations of relatively few basis functions with a simple structure and depending nonlinearly on a set of "inner" parameters e.g., feedforward neural networks with one hidden layer and linear output activation units often provides surprisingly good suboptimal solutions". In 1977, Werbos [18,33] introduced an approach to "approximate dynamic programming", which later became known under the name adaptive critic design (ACD). "A typical design of ACD consists of three modules: action, model (plant), and critic" [8] (p. 587), also in [14,17,18].

In the following part describing Algorithm 1, we resolve how to adjust the critic network, the model network, and the action network. ACD is able to optimize over time under conditions of noise and uncertainty. For control actions to be performed sequentially for a given series, while the effect of the actions is not known up to the end of the sequence, one cannot design an optimal controller with the well-known supervised learning artificial neural network. Progressively adapting two artificial neural networks, we have an adaptive critic method in which the first neural network dispenses the control signals (this is the action network) and the second one (the critic network) learns the requested performance index for some action associated with the index. These two then approximate the optimal control and co-state equation. The adaptation process starts with an arbitrary selection, controlled by the action network. "The critic network then guides the action network towards the optimal solution at each successive adaptation. This method determines optimal control policy for the entire range of initial conditions and needs no external training, unlike other neurocontrollers" [34] (p. 293).

Algorithm 1: Algorithm for Solving the Optimal Control Problem

Input: Select t_0, t_f, a, b, N, M- number of steps, time and space steps h_t, h_s, $\varepsilon_a, \varepsilon_c$- stopping tolerance for action and critic neural networks, respectively, $x_{i,-j} = \phi_s(p_i, t_0 - jh_t), j = k, \ldots, 0$, $u_{i,-j} = \phi_c(i, t_0 - jh_t), j = l, \ldots, 0$- initial values.

Output: Set of final approximate optimal control $\hat{u}(p_i, t_0 + jh_t) = \hat{u}_{ij}$ and optimal trajectory $\hat{x}(p_i, t_0 + (j+1)h_t) = \hat{x}_{i,j+1}, j = 0, \ldots, M-1$,

Set the initial weight $\mathbb{W}^a = (V^a, W^a), \mathbb{W}^c = (V^c, W^c)$

for $j \leftarrow 0$ to $M-1$ do

  for $i \leftarrow 1$ to $N-1$ do

    while $err_a \geq \varepsilon_a$ and $err_c \geq \varepsilon_c$ do

      for $s \leftarrow 0$ to $max(k,l)$ do

        Compute $u^a_{i,s+j}, \mu^a_{i,s+j}$ and $\lambda^c_{i,s+j+1}$ using action ($\mathbb{W}^a$) and critic ($\mathbb{W}^c$) networks, and $x_{i,s+j+1}$ by (10)

      Compute $\lambda^t_{ij}, u^t_{ij}, and\ \mu^t_{ij}$ using (12) and (14) $\mathbb{F}(u_{ij}, \mu_{ij}) = (\mathcal{L}_{u_{ij}}(z, \lambda, \sigma, \mu), -c(x_{ij}, u_{ij})) = 0$

      if $j = M-1$ then

        $\mathbb{F}(u_{i,M-1}, \mu_{i,M-1}, \sigma_i) = (\mathcal{L}_{u_{i,M-1}}(z, \lambda, \sigma, \mu), -c(x_{i,M-1}, u_{i,M-1}), -\psi(x_{i,M}))$ with

        $\lambda_{iM} = \mathcal{G}_{x_{iM}}(x_{iM}) + \sigma_i \psi_{x_{iM}}(x_{iM})$

      $err_c = \| \lambda^t_{ij} - \lambda^c_{ij} \|$

      $err_a = \| (u, \mu)^t_{ij} - (u, \mu)^a_{ij} \|$

      With data set x_{ij}, λ^t_{ij} update weight parameters $\mathbb{W}^c$

      With data set $x_{ij}, (u, \mu)^t_{ij}$ update weight parameters $\mathbb{W}^a$

      Set $\lambda^c_{ij} = \lambda^t_{i,j}, \quad (u, \mu)^a_{i,j} = (u, \mu)^t_{i,j}$

    Set $\hat{\lambda}_{i,j} = \lambda^t_{i,j}, (\hat{u}_{i,j}, \hat{\mu}_{i,j}) = (u, \mu)^t_{i,j}$

    Compute $\hat{x}_{i,j+1}$ using (10) and $\hat{u}_{i,j}$

  $\lambda_{0j} = \lambda_{1j}, \lambda_{Nj} = \lambda_{N-1,j}$

return $\hat{\lambda}_{i,j}, \hat{u}_{i,j}, \hat{\mu}_{i,j}, \hat{x}_{i,j+1}$

Let $x = [x_1, \ldots, x_n]'$ and $y = [y_1, \ldots, y_m]'$ be the input and output vectors for this network. The matrix of synaptic weights between the input nodes and the hidden units is $V = [v_1, \ldots, v_r]'$, where $v_k = [v_{k0}, v_{k1} \ldots, v_{kn}]$; v_{k0} is the bias of the kth hidden unit and v_{ki} is the weight for ith input node connecting to the kth hidden unit. Similarly, $W = [w_1, \ldots, w_m]'$, (where $w_j = [w_{j0}, w_{j1} \ldots, w_{jr}]$; w_{j0} is the bias of jth output unit, and w_{jk} is the weight that connects the kth hidden unit to jth output unit) is the matrix of synaptic weights between the hidden and output units. The response of the kth hidden unit is

given by $z_k = tanh(\sum_{i=0}^{n} v_{ki} x_i), k = 1, \ldots, r$, where tanh(.) is the activation function for the hidden units (hyperbolic tangent), and of the jth output unit is given by $y_j = \sum_{k=0}^{r} w_{jk} z_k$, $j = 1, \ldots, m$.

The number of input variables specifies the number of neurons in the input layer. Multiple layers of neurons with nonlinear transition functions provide the opportunity to learn nonlinear and linear relationships between input and output vectors. The number of output vectors is determined by the number of output variables. The multi-layered network is trained by back-propagation algorithm, where the steepest descent error is propagated backward from the output to the input layer. With the help of the adaptive critic and recurrent neural network we can specify the algorithm for solving the optimal control problem (Algorithm 1). In the maximum principle of Pontryagin for optimal control, the mutual dependence of the state, co-state and control dynamics becomes clear. Naturally, the optimal control $\hat{u}$ and multiplier $\hat{\mu}$ are given by Equation (14), while co-state Equations (12) and (13) progress backward in time and depend on the state and control. Adaptive critic neural networks based on this relationship are shown in [26,27]. There are two networks at each node. The first one is an action network, where the inputs are the current states and their outputs are the corresponding control $\hat{u}$ and multiplier $\hat{\mu}$, and the second one is a critic network, where the current states for the critic network are inputs and current co-states are outputs for normalising the inputs and targets. For a detailed explanation see [35].

The output of the critic network is given by $\lambda^{ij,c} = W^c f(V^c x^{ij})$. The target function $\lambda^{ij,t}$ is expressed by Equation (12). Then, we define the error function of the critic network as $err_c = \| \lambda^{ij,t} - \lambda^{ij,c} \|$. The objective function to be minimised in the critic network is $E_c = 1/2 err_c^2$. The weights are updated by

$$W^c(s+1) = W^c(s) - \rho \frac{\partial E^c}{\partial W^c}, \tag{15}$$

$$V^c(s+1) = V^c(s) - \rho \frac{\partial E^c}{\partial V^c},$$

where $\rho > 0$ is the learning rate of the critic network; j is the inner-loop iteration step for updating the weight matrix W^c and V^c. This is a gradient-based adaptation for training the critic network.

To get the approximate optimal control as an output, the state x^i was used as an input in the action network $(u^{ij,a}, \mu^{ij,a}) = W^a f(V^a x^{ij})$. The target function $(u^{ij,t}, \mu^{ij,t})$ can be written as a solution of Equation (14). The error function of the action network is $err_a = \| (u^{iu,t}, \mu^{ij,t}) - (u^{ij,a}, \mu^{ij,a}) \|$. The objective function to be minimised in the action network is $E_a = 1/2 err_a^2$. The weights are updated by

$$W^a(s+1) = W^a(s) - \rho \frac{\partial E^a}{\partial W^a}, \tag{16}$$

$$V^a(s+1) = V^a(s) - \rho \frac{\partial E^a}{\partial V^a},$$

where the meaning of $\rho > 0$ and j are the same as in (15) for the weight matrices of action network W^a and V^a.

The whole procedure of the distributed optimal control problem is summarised in Algorithm 1. Now, we describe the condition for terminating the training of action and critic networks within each cycle $j = 0, \ldots, M-1$ and $i = 1, \ldots, N-1$. We use stopping tolerance ε_a for action network to approximate optimal control u and multipliers μ. Convergence condition is given as $err_a = \| (u, \mu)_{ij}^t - (u, \mu)_{ij}^a \|$, where $(u, \mu)_{ij}^t$ and $(u, \mu)_{ij}^a$ are solutions of optimality condition (14), inequality multipliers and an action network output, respectively. Similarly, we use stopping tolerance ε_c for critic network to approximate co-state variable λ. Convergence condition is given as $err_c = \| \lambda_{ij}^t - \lambda_{ij}^c \|$, where λ_{ij}^t and λ_{ij}^c are a solutions of

co-state Equation (12) and the critic network output, respectively. The proof of convergence of adaptive critic design for a class of parabolic partial differential equations is given in [25]. For nonlinear distributed systems with time delay we can examine only the numerical convergence of ACD.

The learning procedure for the action and critic network is shown in [27,35]. The action and critical networks in adaptive critic synthesis were chosen to consist of $n+m$ subnetworks, each constructed with $n-3n-1$ structure such that there are n, $3n$ and 1 neurons in the input, the hidden and the output layer, respectively.

5. SEIR Model with Spatial Diffusions

The classical SEIR (Susceptible—Exposed—Infectious–Recovered) model is one of the most used models in the simulation of infectious diseases [36]. In the SEIR model, the entire population is divided into four distinct groups of individuals. The first group is created by susceptible individuals (marked by S, x_1) who are able to catch the disease, the second group contains exposed persons (marked by E, x_2), who are infected but not yet infectious, infective people are in the third group (I, x_3), they are able to transmit the disease, and the last group contains recovered individuals (marked by R, x_4), who are permanently immune. The whole population size is N, while the letters S, E, I, R sign the number of individuals in each group at a particular time, so $S+E+I+R=N$. We suppose that the population size N is constant and that the populations are well mixed (the spatial mobility of individuals has been ignored in the model) [37]. The evolution of the groups of the population over time in the SEIR model is described by the following four non-linear partial differential equations:

$$\frac{\partial x_i(p,t)}{\partial t} = D_i \frac{\partial^2 x_i(p,t)}{\partial p^2} + F_i(x(p,t), x(p,t-\tau), u(p,t), t), \tag{17}$$

where $F_i, i=1,\ldots,4$ are given by Equation (20) with Neumann boundary condition

$$\frac{\partial x_i}{\partial p}(0,t) = \frac{\partial x_i}{\partial p}(1,t) = 0 \tag{18}$$

and initial conditions

$$x_i(p,0) = \phi_i(p) \geq 0, \;\; 0 \leq p \leq 1, \; i=1,\ldots,4 \tag{19}$$

and

$$\begin{aligned} F_1(x(p,t), u(p,t),, t) &= \mu(N(t) - x_1(p,t)) - \beta \frac{x_3(p,t-\tau)}{N(t)} x_1(p,t) - u(p,t)x_1(p,t) \\ F_2(x(p,t), u(p,t),, t) &= \beta \frac{x_3(p,t-\tau)}{N(t)} x_1(p,t) - (\mu+\gamma)x_2(p,t) \\ F_3(x(p,t), u(p,t),, t) &= \gamma x_2(p,t) - (\mu+\delta)x_3(p,t) \\ F_4(x(p,t), u(p,t),, t) &= \delta x_3(p,t) - \mu x_4(p,t) + u(p,t)x_1(p,t) \end{aligned} \tag{20}$$

Here, t denotes the time, p represents the spatial location, D_i are the diffusion coefficients, μ is both birth and death rate, β is the contact rate that indicates the probability of catching the disease between susceptible and infectious individuals with time delay τ, $1/\delta$ is the mean latent period of the disease, $1/\gamma$ is the mean infectious period and $u(p,t)$ is the vaccination coverage of susceptible individuals. We set $N(t) = \int_0^1 \left(\sum_{i=1}^{4} x_i(p,t) \right) dp$ for all $t \geq 0$, where $x(\phi,t,p) = (x_1(\phi,t,p), \ldots, x_4(\phi,t,p))$ is the solution of Equation (17).

It follows from the model that

$$\frac{dN(t)}{dt} = \frac{d}{dt}\int_0^1 \left(\sum_{i=1}^4 x_i(p,t)\right)dp = \int_0^1 \frac{d}{dt}\left(\sum_{i=1}^4 x_i(p,t)\right)dp = \int_0^1 \left(\sum_{i=1}^4 \frac{d}{dt}x_i(p,t)\right)dp =$$
$$\sum_{i=1}^4 \left(\int_0^1 D_i \frac{\partial^2 x_i(p,t)}{\partial p^2}dp\right) = \sum_{i=1}^4 D_i\left(\frac{\partial x_i}{\partial p}(1,t) - \frac{\partial x_i}{\partial p}(0,t)\right) = 0.$$

It follows easily that

$$N(t) = \int_0^1 \left(\sum_{i=1}^4 x_i(p,t)\right)dp = c \tag{21}$$

and $c = \int_0^1 \left(\sum_{i=1}^4 x_i(p,0)\right)dp$, i.e., that the population size $N(t)$ is constant for all t.

Distributed Optimal Control Problem of Spread of Diseases

To control the spread of diseases, we introduce an optimal control in the SEIR model. The SEIR models with optimal control are currently being intensively studied, see [38–40]. Our objective is to find the optimal vaccine coverage threshold $u(p,t)$ for disease control and liquidation [41].

Our optimal control problem is to find a distributed control $u(p,t)$ to minimise the following objective function:

$$\mathcal{J}(u) = \int_a^b \int_{t_0}^{t_f} \left(a_1 x_1(p,t) + a_2 x_2(p,t) + a_3 x_3(p,t) + \frac{1}{2}u(p,t)^2\right)dtdp, \tag{22}$$

subject to the state system (17)–(18), under constraints

$$c_1(x,u) = u_{min} - u \leq 0$$

$$c_2(x,u) = u - u_{max} \leq 0,$$

where a_1, a_2, a_3 are positive constants to keep a balance in the size of S, E and I.

The augmented Hamiltonian function for problem (22) is given by

$$\mathcal{H}(x, x_\tau, u, \lambda) = a_1 x_1(p,t) + a_2 x_2(p,t) + a_3 x_3(p,t) + \tag{23}$$

$$\frac{1}{2}u(p,t)^2 + \sum_{j=1}^4 \lambda_j F_j(x, x_\tau, u) = \frac{1}{2}u(p,t)^2 - u(p,t)x_1(p,t)(\lambda_1(p,t) - \lambda_4(p,t)) + G(p,t),$$

where $\lambda \in R^4$ is the adjoint variable. Let $(\hat{x}, \hat{u})$ be an optimal solution for (22). Then the necessary optimality condition for (22) implies [24] that there exists a piecewise continuous and piecewise continuously differentiable adjoint function $\lambda : Q \to R^4$, satisfying

$$\frac{\partial \lambda}{\partial t} = D\frac{\partial^2 \lambda}{\partial p^2} - \frac{\partial \mathcal{H}}{\partial x}(x, x_\tau, u, \lambda) - \chi_{[t_0, t_f - \tau]}\frac{\partial \mathcal{H}}{\partial x_\tau}(x_{+\tau}, x, u, \lambda_{+\tau}), \tag{24}$$

$$\lambda(p, t_f) = 0, \frac{\partial \lambda(a,t)}{\partial p} = \frac{\partial \lambda(b,t)}{\partial p} = 0, \tag{25}$$

$$0 = \frac{\partial \mathcal{H}}{\partial u}(x, x_\tau u, \lambda). \tag{26}$$

By the optimality condition (26), we have

$$\frac{\partial \mathcal{H}}{\partial u}(x, x_\tau, u, \lambda) = u(p,t) - x_1(p,t)(\lambda_1(p,t) - \lambda_4(p,t)) = 0.$$

Now, using the property of the control space $u \in \langle u_{min}, u_{max} \rangle$, we get

$$\hat{u}(p,t) = u_{min}, (\lambda_1(p,t) - \lambda_4(p,t)) \leq u_{min}, \tag{27}$$

$$\hat{u}(p,t) = \hat{x}_1(p,t)(\lambda_1(p,t) - \lambda_4(p,t)),$$

$$u_{min} < x_1(p,t)(\lambda_1(p,t) - \lambda_4(p,t)) < u_{max},$$

$$\hat{u}(p,t) = u_{max}, (\lambda_1(p,t) - \lambda_4(p,t)) \geq u_{max}.$$

Discretisation of Equations (17) and (22) using Equations (12)–(14) and (9) leads to minimize

$$\mathcal{J}(u) = -h_s h_t \sum_{(i,j)} a_1 x_1(p_i, t_j) + a_2 x_2(p_i, t_j) + a_3 x_3(p_i, t_j) + \frac{1}{2} u(p_i, t_j)^2$$

subject to

$$x_{i,j+1} = x_{ij} + h_t D \frac{x_{i-1,j} - 2x_{ij} + x_{i+1,j}}{h_s^2} + h_t F_{ij},$$

$$x_{0j} = x_{1j}, x_{Nj} = x_{N-1j},$$

$$x_{i,0} = \phi_x(p_i, t_0), i = 0, \ldots, N,$$

$$u_{ij} - m_{max} \leq 0, m_{min} - u_{ij} \leq 0,$$

$$i = 0, \ldots, N, j = 0, \ldots, M-1$$

and Equations (12)–(14). The vector function $F(x,u) = (a_1 x_1(p,t) + a_2 x_2(p,t) + a_3 x_3(p,t) + \frac{1}{2} u(p,t)^2, F_1(x,u), \ldots, F_4(x,u))$ is given by right-hand side of Equation (17). Discrete form of optimality condition (14) is

$$\hat{u}(p_i, t_j) = u_{min}, (\lambda_1(p_i, t_j) - \lambda_4(p_i, t_j)) \leq u_{min}, \tag{28}$$

$$\hat{u}(p_i, t_j) = \hat{x}_1(p_i, t_j)(\lambda_1(p_i, t_j) - \lambda_4(p_i, t_j)),$$

$$u_{min} < x_1(p_i, t_j)(\lambda_1(p_i, t_j) - \lambda_4(p_i, t_j)) < u_{max},$$

$$\hat{u}(p_i, t_j) = u_{max}, (\lambda_1(p_i, t_j) - \lambda_4(p_i, t_j)) \geq u_{max}.$$

The solution of distributed optimal control problem (22) using the algorithm for solving the optimal control problem are displayed in Figures 1–4.

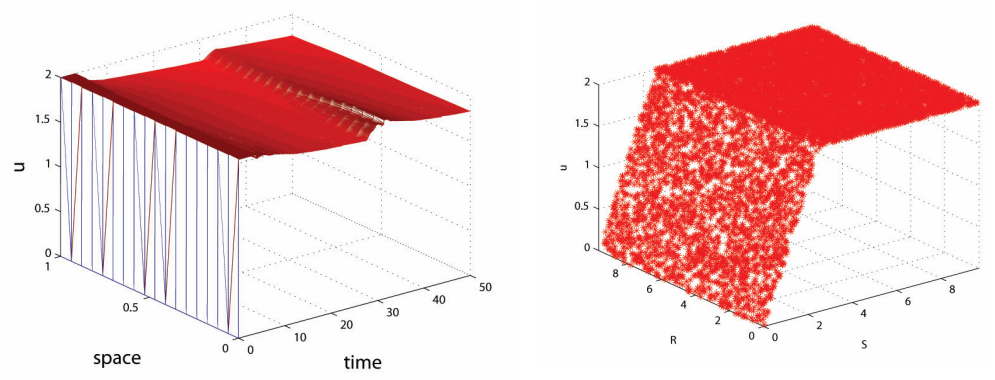


Figure 1. Adaptive critic neural network simulation of optimal control $\hat{u}(t)$ with the initial condition $\psi_s(t) = (8,3,2,1)(2 + cos(2\pi p))$ and adaptive critic neural networks simulation of optimal feedback law for 18,000 x^k state input signal (optimal control $\hat{u}(t)$ depending on the input x^k).

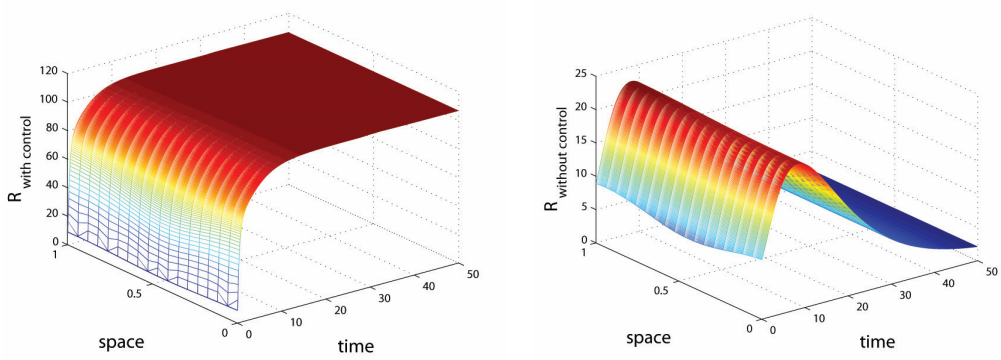


Figure 2. The immune population both with control and without control with the initial condition $\psi_s(0) = (8,3,2,1)(2 + cos(2\pi p))$.

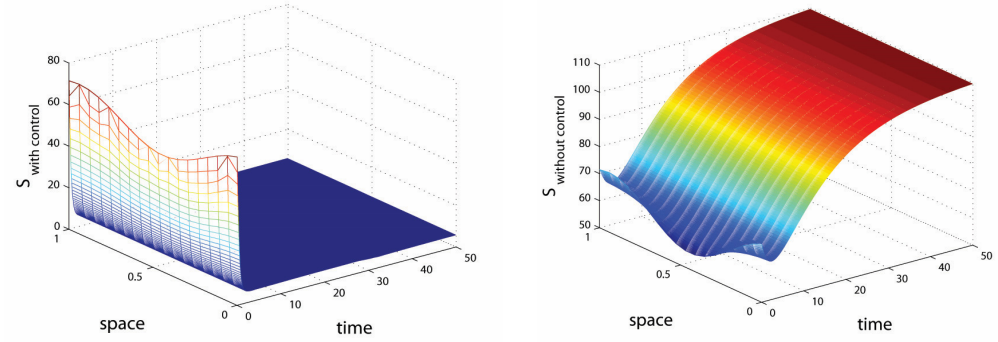


Figure 3. The susceptible population both with control and without control with the initial condition $\psi_s(0) = (8,3,2,1)(2 + cos(2\pi p))$.

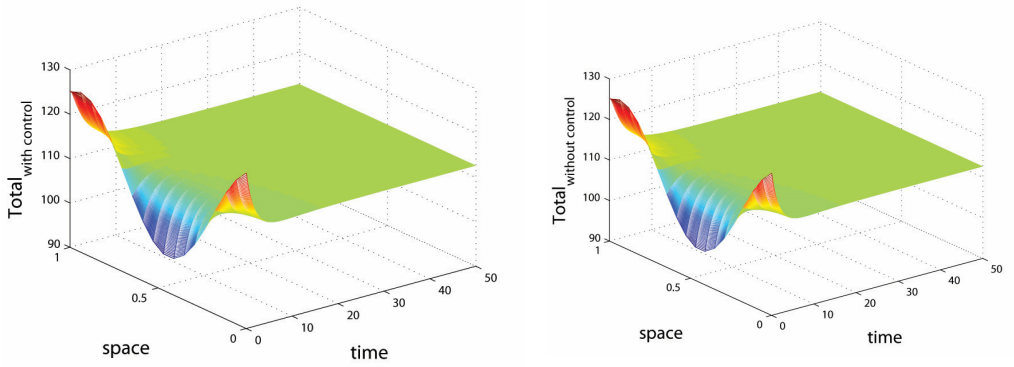


Figure 4. The total population size both with control and without control with the initial condition $\psi_s(0) = (8, 3, 2, 1)(2 + cos(2\pi p))$.

In this simulation, we have chosen a critic network consisting of 4 subnetworks and an action network consisting of one subnetwork. Each subnetwork has 4–12–1 structure, where the triple of numbers indicates the number of neurons in the input layer, hidden layer, and output layer, respectively. During the adaptive critic synthesis, the proposed neural network fulfils the chosen convergence tolerance values, which leads to satisfactory simulation results.

We have plotted immune, susceptible individuals with and without control by considering values of parameters as $a_2 = 0.3, a_3 = 0.2, \tau = 0.5$. We can see that the population of immune individuals with control and without control increasing and decreasing to its stable state, respectively, see Figure 2. In the case of susceptible individuals, we get the opposite trend; see Figure 3.

6. Conclusions

In this paper, we have presented a new algorithm of adaptive critic method for parabolic distributed optimal control problem. The method is based on feed-forward neural network approximation of co-state variables, where state variable x and co-state variable λ are solved forward in time and are able to simulate optimal feedback law. The new algorithm was applied to the SEIR model to minimize the infected and susceptible populations and to maximise recovered populations. The results of numerical simulation (using Matlab) have shown that our algorithm for solving a distributed optimal control problem is able to solve this practical problem successfully. We presented a comparison between optimal control and no control of the SEIR model (see Figures 2–4). It is easily seen that the optimal control of the SEIR model is able to find the effective vaccination rate of the population to influence the population infection by decreasing the number of susceptible individuals.

Author Contributions: Conceptualization, T.K., M.K. and L.V.; methodology, T.K.; software, T.K.; validation, T.K., M.K. and L.V.; formal analysis, T.K., M.K. and L.V; investigation, T.K.; resources T.K., M.K. and L.V.; data curation, T.K.; writing—original draft preparation, T.K., M.K. and L.V.; writing—review and editing, T.K., M.K. and L.V.; visualization, T.K., M.K. and L.V.; supervision, T.K. All authors have read and agreed to the published version of the manuscript.

Funding: The paper was supported by the Pallas Athéné Domus Meriti Foundation (PADME-2023) and project KEGA 015UKF-4/2021 Collaboration as a means of professional growth of mathematics teachers.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yang, X.; He, H. Adaptive critic designs for optimal control of uncertain nonlinear systems with unmatched interconnections. *Neural Netw.* **2018**, *105*, 42–153. [CrossRef] [PubMed]
2. He, W.; Chen, Y.; Yin, Z. Adaptive neural network control of an uncertain robot with full-state constraints. *IEEE Trans. Syst. Man Cybern. Syst.* **2016**, *46*, 620–629. [CrossRef]
3. He, W.; Dong, Y.; Sun, C. Adaptive neural impedance control of a robotic manipulator with input saturation. *IEEE Trans. Syst. Man Cybern. Syst.* **2016**, *46*, 334–344. [CrossRef]
4. He, W.; Ge, S.S. Cooperative control of a nonuniform gantry crane with constrained tension. *Automatica* **2016**, *66*, 146–154. [CrossRef]
5. Marino, R. Adaptive control of nonlinear systems: Basic results and applications. *Annu. Rev. Control* **1997**, *21*, 55–66. [CrossRef]
6. Wang, D.; Liu, D.; Wei, Q.; Zhao, D.; Jin, N. Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming. *Automatica* **2012**, *48*, 1825–1832. [CrossRef]
7. Kmet, T.; Kmetova, M. Neural network solution of distributed control problems with control and state constraints. In *LNCS 9886 Artificial Neural Networks and Machine Learning—ICANN2016*; Villa, A., Masulli, P., Rivero, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 468–477.
8. Kmet, T.; Kmetova, M. Adaptive critic design and Hopfield neural network based simulation of time delayed photosynthetic production and prey-predator model. *Inf. Sci.* **2015**, *294*, 586–599. [CrossRef]
9. Kmet, T.; Kmetova, M. Neural networks simulation of feeding adaptations of daphnia. In *LNCS 9702 Algorithms for Computational Biology AlCoB 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 26–37.
10. Kmet, T.; Kmetova, M. Echo State Networks Simulation of SIR Distributed Control. In *LNAI 10245 Artificial Intelligence and Soft Computing ICAISC 2017*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 86–96.
11. Borzi, A. Multigrid methods for parabolic distributed optimal control problems. *J. Comput. Appl. Math.* **2003**, *157*, 365–382. [CrossRef]
12. Mittelmann, H.D. Solving elliptic control problems with interior point and SQP methods: Control and state constraints. *J. Comput. Appl. Math.* **2000**, *120*, 175–195. [CrossRef]
13. Liu, D.; Wang, D.; Yang, X. An iterative adaptive dynamic programming algorithm for optimal control of unknown discrete-time nonlinear systems with constrained inputs. *Inf. Sci.* **2013**, *220*, 331–342. [CrossRef]
14. Prokhorov, D.V.; Wunsch, D.C. Adaptive critic designs. *IEEE Trans. Neural Netw.* **1997**, *8*, 997–1007. [CrossRef]
15. Vamvoudakis, K.G.; Lewis, F.L. Online actor–critic algorithm to solve the continuous-time infinite horizon optimal control problem. *Automatica* **2010**, *46*, 878–888. [CrossRef]
16. Levin, A.U.; Narendra, K.S. Control of nonlinear dynamic systems using neural networks: Controllability and stabilization adaptive critic designs. *IEEE Trans. Neural Netw.* **1993**, *4*, 192–206. [CrossRef] [PubMed]
17. Werbos, P.J. A menu of designs for reinforcement learning over time. In *Neural Networks for Control*; The MIT Press: Cambridge, MA, USA, 1990; pp. 333–366.
18. Werbos, P.J. Approximate dynamic programming for real-time control and neural modelling. In *Handbook of Intelligent Control: Neural Fuzzy, and Adaptive Approaches*; White, D.A., Sofge, D.A., Eds.; Van Nostrand: New York, NY, USA, 1992; Chapter 13.
19. Hornik, M.; Stichcombe, M.; White, H. Multilayer feed forward networks are universal approximators. *Neural Netw.* **1989**, *2*, 259–366. [CrossRef]
20. Chryssoverghi, I. Discretization methods for semilinear parabolic optimal control problems. *Int. J. Numer. Anal. Model.* **2006**, *3*, 437–458.
21. Clever, D.; Lang, J.; Ulbrich, S.; Ziems, J.C. Combination of an Adaptive Multilevel SQP Method and a Space-Time Adaptive PDAE Solver for Optimal Control Problems. *Procedia Comput. Sci.* **2012**, *1*, 1435–1443. [CrossRef]
22. Knowles, G. Finite element approximation of parabolic time optimal control problems. *SIAM J. Control Optim.* **1982**, *20*, 414–427. [CrossRef]
23. Buskens, C.; Maurer, H. SQP-methods for solving optimal control problems with control and state constraints: Adjoint variable, sensitivity analysis and real-time control. *J. Comput. Appl. Math.* **2000**, *120*, 85–108. [CrossRef]
24. Gollman, L.; Kern, D.; Mauer, H. Optimal control problem with delays in state and control variables subject to mixed control-state constraints. *Optim. Control Appl. Meth.* **2009**, *30*, 341–365. [CrossRef]
25. Luo, Y.; Sun, Q.; Zhang, H.; Cui, L. Adaptive critic design-based robust neural network control for nonlinear distributed parameter systems with unknown dynamics. *Neurocomputing* **2015**, *148*, 200–208. [CrossRef]
26. Padhi, R.; Balakrishnan, S.; Randoltph, T. A single network adaptive critic (SNAC) architecture for optimal control synthesis for a class of nonlinear systems. *Neural Netw.* **2006**, *19*, 1648–1660. [CrossRef] [PubMed]
27. Padhi, R.; Unnikrishnan, N.; Wang, X.; Balakrishnan, S.N. Adaptive-critic based optimal control synthesis for distributed parameter systems. *Automatica* **2001**, *37*, 1223–1234. [CrossRef]
28. Caas, E. Pontryagin's principle for state-constrained boundary control problems of semilinear parabolic equations. *SIAM J. Control Optim.* **1997**, *35*, 1297–1327. [CrossRef]
29. Kirk, D.E. *Optimal Control Theory: An Introduction*; Dover Publications Inc.: Mineola, NY, USA, 1989.
30. Potryagin, D.E. *The Mathematical Theory of Optimal Process*; Nauka: Moscow, Russia, 1983.
31. Polak, E. *Optimization Algorithms and Consistent Approximation*; Springer: Berlin/Heidelberg, Germany, 1997.

32. Gnecco, G. A Comparison between Fixed-Basis and Variable-Basis Schemes for Function Approximation and Functional Optimization. *J. Appl. Math.* **2012**, *2012*, 806945. [CrossRef]
33. Werbos, P. Advanced Forecasting Methods for Global Crisis Warning and Models of Intelligence; General Systems Yearbook; 1977. Available online: https://www.researchgate.net/publication/243634290_Advanced_Forecasting_Methods_for_Global_Crisis_Warning_and_Models_of_Intelligence#fullTextFileContent (accessed on 1 March 2023).
34. Venayagamoorthy, G.K.; Wunsch, D.C.; Harley, R.G. Adaptive critic based neurocontroller for turbogenerators with global dual heuristic programming. In Proceedings of the 2000 IEEE Power Engineering Society Winter Meeting. Conference Proceedings (Cat. No.00CH37077), Singapore, 23–27 January 2000; pp. 291–294.
35. Rumelhart, D.F.; Hinton, G.E.; Wiliams, R.J. Learning internal representation by error propagation. In *Parallel Distributed Processing: Foundation*; Rumelhart, D.E., McClelland, D.E., Group, P.R., Eds.; The MIT Press: Cambridge, MA, USA, 1987; Volume 1, pp. 318–362.
36. Aron, J.L.; Schwartz, I.B. Seasonality and period-doubling bifurcations in an epidemic model. *J. Theor. Biol.* **1984**, *110*, 665–679. [CrossRef]
37. Holko, A.; Medrek, M.; Pastuszak, Z.; Phusavat, K. Epidemiological modeling with a population density map-based cellular automata simulation system. *Expert Syst. Appl.* **2016**, *48*, 1–8. [CrossRef]
38. Zhou, J.; Zhao, Y.; Ye, Y. Complex dynamics and control strategies of SEIR heterogeneous network model with saturated treatment. *Phys. A* **2022**, *608*, 128287. [CrossRef]
39. Meng, L.; Zhu, W. Analysis of SEIR epidemic patch model with nonlinear incidence rate, vaccination and quarantine strategies. *Math. Comput. Simul.* **2022**, *200*, 489–503. [CrossRef]
40. Wang, Z.; Zhang, Q. Near-optimal control of a stochastic partial differential equation SEIR epidemic model under economic constraints. *Eur. J. Control* **2023**, *69*, 100757. [CrossRef]
41. Kar, T.K.; Batabyal, A. Stability analysis and optimal control of an SIR epidemic model with vaccination. *BioSystems* **2011**, *104*, 127–135. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.