01 Jan 2003

Approximate Dynamic Programming Based Optimal Neurocontrol Synthesis of a Chemical Reactor Process Using Proper Orthogonal Decomposition

Radhakant Padhi

S. N. Balakrishnan
Missouri University of Science and Technology, bala@mst.edu

Recommended Citation

R. Padhi and S. N. Balakrishnan, "Approximate Dynamic Programming Based Optimal Neurocontrol Synthesis of a Chemical Reactor Process Using Proper Orthogonal Decomposition," *Proceedings of the International Joint Conference on Neural Networks, 2003*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2003.
The definitive version is available at https://doi.org/10.1109/IJCNN.2003.1223696

Approximate dynamic programming based optimal neurocontrol synthesis of a chemical reactor process using proper orthogonal decomposition

Radhakant Padhi[1] **and S. N. Balakrishnan**[2]
Department of Mechanical and Aerospace Engineering, and Engineering Mechanics
University of Missouri-Rolla, MO, 65409, USA

Abstract – **The concept of approximate dynamic programming and adaptive critic neural network based optimal controller is extended in this study to include systems governed by partial differential equations. An optimal controller is synthesized for a dispersion type tubular chemical reactor, which is governed by two coupled nonlinear partial differential equations. It consists of three steps: First, empirical basis functions are designed using the 'Proper Orthogonal Decomposition' technique and a *low-order* lumped parameter system to represent the infinite-dimensional system is obtained by carrying out a Galerkin projection. Second, approximate dynamic programming technique is applied in a discrete time framework, followed by the use of a dual neural network structure called *adaptive critics,* to obtain optimal neurocontrollers for this system. In this structure, one set of neural networks captures the relationship between the state variables and the control, whereas the other set captures the relationship between the state and the costate variables. Third, the lumped parameter control is then mapped back to the spatial dimension using the same basis functions to result in a feedback control. Numerical results are presented that illustrate the potential of this approach. It should be noted that the procedure presented in this study can be used in synthesizing optimal controllers for a fairly general class of nonlinear distributed parameter systems.**

I. INTRODUCTION

Process control problems are mostly governed by partial differential equations (PDEs) and are infinite-dimensional in nature. They are also called Distributed Parameter Systems (DPS). The DPS appear naturally in various application areas such as chemical processes, thermal processes, vibrating structures, fluid flow systems etc. They inherently have an infinite number of system modes. Since it is impossible to deal with all the modes, some sort of approximation technique is usually applied for the analysis and synthesis procedures related to DPS.

A popular DPS analysis and synthesis technique is to use orthogonal basis functions in a Galerkin procedure to first create an approximate finite-dimensional system of Ordinary Differential Equations (ODEs). This lumped parameter model model is then used for control design using various tools of lumped parameter control design. If arbitrary basis functions (*e.g.* Fourier and Chebyshev polynomials) are used in the Galerkin procedure, they can result in a high-dimensional ODE system. A better and powerful basis function design is obtained when the Proper Orthogonal Decomposition (POD) technique is used with a Galerkin approximation. In the POD technique, a set of problem-oriented basis functions is first obtained by generating a set of "snap-shot solutions" through simulations or from the actual process. Using these orthogonal basis functions in a Galerkin procedure, a low-dimensional ODE system can be developed. This technique has widely been used in recent years (*e.g.* [4, 7]).

The issue of optimal control synthesis should be addressed next. It is well known that the dynamic programming formulation offers the most comprehensive solution to nonlinear optimal control; however, a huge amount of computational and storage requirements are needed to solve the associated Hamilton-Jacobi-Bellman (HJB) equation [2], which is also known as the Bellman equation. Werbos [10] proposed a means to get around this numerical complexity by using 'approximate dynamic programming' (ADP) formulations. His methods approximate the original problem with a discrete formulation. The solution to the ADP formulation is obtained through the two-neural network adaptive critic approach. In one version of the adaptive critic approach called the dual heuristic programming (DHP) one network called the action network represents the mapping between the state variables of a dynamic system and control and the second network, called the critic, outputs the costates with the state variables as its inputs. This ADP process, through the nonlinear function approximation capabilities of neural networks, overcomes the computational complexity that plagued the dynamic programming formulation of optimal control problems. More important, this solution can be implemented on-line, since the control computation requires a few multiplications of the network weights which are trained off-line. This technique was used in [1] to solve an aircraft control problem in a domain of interest.

In this paper, this techniques of POD and approximate dynamic programming are combined, which is then applied to a more challenging *nonlinear* chemical reactor process. This dispersion type tubular chemical reactor control problem has been discussed in [3]. The authors have used Green's function to calculate optimal control. Their method for calculating the costate variables that arise in an optimal control formulation is complicated. Even though the authors have found a Greens function for the particular problem, finding an appropriate Green's function and calculating its coefficients is not an easy task in general. More important, their solution is for specific initial condition (initial state profiles) only. In other words, it is an open loop control which will severely degrade the process performance if the initial profile were different. In contrast to this, the approach presented in this paper is applicable to a large number of initial conditions (or profiles). Once the neural networks are trained to capture the relationship between state and control within a domain of interest (which is done off-line) they can be used to compute the control for "any" value of the state variables within that domain. Moreover since using a set of networks is not

[1] Postdoctoral Fellow
[2] Professor (Contact Person), Email: bala@umr.edu, Tel.: 1(573) 341-4675, Fax: 1(573)341-4607.

0-7803-7898-9/03/$17.00 ©2003 IEEE

computationally intensive it can be implemented on-line. In control terminology this is a feedback solution; a feedback control is desirable because of its beneficial properties like robustness with respect to noise suppression and modeling uncertainties.

We wish to point out that we have solved the same chemical reactor optimal control problem using a different approach earlier [8]. In that approach a controller was used at every step in a finite difference scheme, which was used to discretize the spatial variable. Even though we obtained satisfactory results, there are some implementation issues. Note that one has to take a large number of node points for good finite difference approximations. However, because for each node point critic and action networks were proposed, the number of networks grows with the number of grid points and this would lead to serious problem in training of the networks. As a consequence one has to remain contented with a "coarse grid approximation". In contrast, the current approach is grid independent in the sense that lumped parameter state vector does not depend on the number of grid points assumed for the integral evaluations. Second, in the earlier technique the state (and control) values at some point in space other than the node point locations are unknown. If one wants to get value for such a location, interpolation techniques are necessary. The prediction may not be good if the grid approximation is coarse. In contrast, in the proposed methodology by definition the basis functions are *supposed to be continuous functions. So values at any point in the* space can theoretically be computed without resorting to any interpolation technique. This issue is of significantly less concern in our new approach, since one can have a fine-grid approximation to begin with and therefore, will result in much smaller interpolation errors.

II. SYSTEM MODEL AND OBJECTIVE

Dynamics of the chemical reactor problem considered in this research is described by the following set of partial differential equations [3]:

$$\frac{\partial v_1}{\partial t}=\frac{1}{Pe_1}\left(\frac{\partial^2 v_1}{\partial y^2}\right)-\left(\frac{\partial v_1}{\partial y}\right)+N(v_2)\ \left(1-v_1\right) \tag{1a}$$

$$\frac{\partial v_2}{\partial t}=\frac{1}{Pe_2}\left(\frac{\partial^2 v_2}{\partial y^2}\right)-\left(\frac{\partial v_2}{\partial y}\right)+B\ N(v_2)\left(1-v_1\right)-h\left(v_2-u\right) \tag{1b}$$

The boundary conditions are given by

$$\left[\frac{\partial v_1}{\partial y}=Pe_1 v_1\right]_{y=0},\left[\frac{\partial v_2}{\partial y}=Pe_2 v_2\right]_{y=0},\left.\frac{\partial v_1}{\partial y}\right|_{y=L}=0,\left.\frac{\partial v_2}{\partial y}\right|_{y=L}=0 \tag{2}$$

where, v_1, v_2 are the state variables that represent concentration and temperature respectively. The control variable u represents the cooling water temperature. The terms Pe_1, Pe_2 are the *Peclet* numbers of mass and energy flows respectively. $N\left(v_2\right)=D_a\exp\left[v_2/\left(1+v_2/\varepsilon\right)\right]$, where D_a is the *Damkohler* number, ε is the activation energy, B and h are the parameters related to heat of reaction and heat transfer respectively. $y\in[0,\,L]$ and $t\in\left[t_0,t_f\right]$, where t_0 and t_f are initial and final times *respectively. Values of different parameters describing the process* are: $Pe_1=1$, $Pe_2=1$, $D_a=1$, $\varepsilon=20$, $B=2$, $h=1$, $m=2$, $t_0=0$ and $L=1$.

For convenience, we define v_{1ref} and v_{2ref} as the steady state values of v_1 *and* v_2 *respectively while the* $h\left(v_2-u\right)$ term is omitted from the v_2 dynamics; this implies that the reactor operates with perfect insulation. Consequently, we observe that $u_{ref}=v_{2_{ref}}$. For computing $v_{1_{ref}}$ and $v_{2_{ref}}$, it was assumed that $\partial v_1/\partial t=\partial v_2/\partial t=0$ (steady-state condition), in addition to $h\left(v_2-u\right)=0$. Then we use these conditions in (1-2) and solved the resulting two point boundary value problem (in spatial dimension) for v_{1ref} and v_{2ref}.

Defining $x_1\triangleq\left(v_1-v_{1ref}\right)$, $x_2\triangleq\left(v_2-v_{2ref}\right)$, we rewrite (1) as

$$\frac{\partial x_1}{\partial t}=\frac{1}{Pe_1}\left(\frac{\partial^2 x_1}{\partial y^2}\right)-\left(\frac{\partial x_1}{\partial y}\right)+f\left(x_1,\ x_2\right) \tag{3a}$$

$$\frac{\partial x_2}{\partial t}=\frac{1}{Pe_2}\left(\frac{\partial^2 x_2}{\partial y^2}\right)-\left(\frac{\partial x_2}{\partial y}\right)+B\,f\left(x_1,\ x_2\right)\ -\ h\,v \tag{3b}$$

where

$$f\left(x_1,\ x_2\right)\equiv N(v_{2ref}+x_2)\left(1-v_{1ref}-x_1\right)-N(v_{2ref})\left(1-v_{1ref}\right) \tag{3c}$$

Here $v\equiv\left(u-x_2\right)$ is the new auxiliary control variable. The objective is to find the optimal control $u(t,y)$ which ensures $\left\{x_1,x_2\right\}\rightarrow\left\{0,0\right\}$ $\left(i.e.\ \left\{v_1,v_2\right\}\rightarrow\left\{v_{1_{ref}},v_{2_{ref}}\right\}\right)$ as $t\rightarrow\infty$. This objective can be met by minimizing the quadratic cost function:

$$J=\frac{1}{2}\int_0^\infty\int_0^L\left[q_1 x_1^2(t,y)+q_2 x_2^2(t,y)+r u^2(t,y)\ \right]dy\,dt \tag{4}$$

where $q_1, q_2\geq 0$ and $r>0$ are weights to be appropriately fixed by the control designer. We have used $q_1=5000$, $q_2=1$ and $r=1$. The relatively high value of q_1 was selected mainly because our goal was to drive v_1 towards v_{1ref} as quickly as possible.

III. FINITE-DIMENSIONAL APPROXIMATION

A. Proper Orthogonal Decomposition: Design of Basis Functions

Let $\{U_i(y):\ 1\leq i\leq N,\ 0\leq y\leq L\}$ be a set of N snapshot solutions of the system. The goal of the POD technique is to design a set of basis functions which has the largest mean square projection on the snapshots. In other words, we try to find all such possible basis functions Φ, each of which provides a local maximum for the following figure of merit:

$$I=\frac{1}{N}\sum_{i=1}^{N}\left|\left\langle U_i,\Phi\right\rangle\right|^2/\left\langle\Phi,\Phi\right\rangle=\frac{1}{N}\left[\sum_{i=1}^{N}\left|\int_0^L U_i\,\Phi\,dy\right|^2/\int_0^L\Phi\Phi\,dy\right] \tag{5}$$

The solution approach is to seek a function $\Phi=\sum_{i=1}^{N}w_i U_i$, where the coefficients w_i are to be determined such that Φ maximizes I in (5). In the process we obtain N orthonormal basis functions $\Phi_i, i=1,\ldots,N$. Depending on the energy content, this eigen spectrum is truncated to retain only $\tilde{N} \le N$ eigen functions that will be used in the Galerkin projection. An interested reader can see [7] for detail discussions on this basis function design procedure. It may be noted, however, that the POD technique is a generalization of a familiar method known as Principal Component Analysis (PCA) [6], to continuous square integrable functions. The PCA technique is widely used as a tool in pattern recognition, image processing *etc.*

For the process control problem, the basis functions for x_1 and x_2 were designed independently. In order to determine the proper order of the system, the ratios $\sum_{j=1}^{\tilde{N}_1} \lambda_{1_j} / \sum_{j=1}^{N} \lambda_{1_j}$ and $\sum_{j=1}^{\tilde{N}_2} \lambda_{2_j} / \sum_{j=1}^{N} \lambda_{2_j}$ were plotted for different values of $\tilde{N}_1, \tilde{N}_2$ and it was observed that 99% of the ratio was accounted for by the first three eigenvalues for both the state variables. Hence, we fixed $\tilde{N}_1 = \tilde{N}_2 = 3$ and assumed that the six basis functions captured the essential characteristics contained in the snap shots with sufficient accuracy. The basis functions for state variables x_1 and x_2 in the chemical reactor control problem are shown in Figure 1.

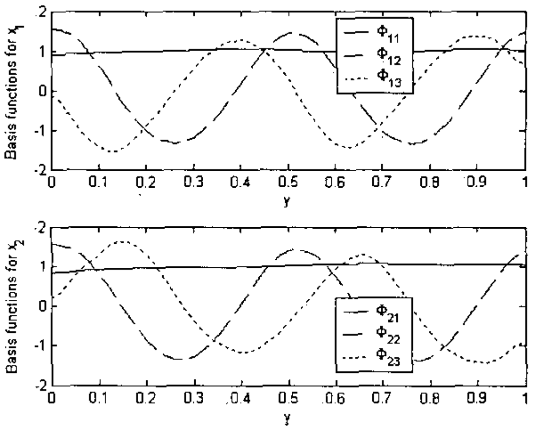

Figure 1: POD Basis functions for state variables

B. Finite-dimensional Approximation: Galerkin Projection

The state variables $x_1(t,y)$, $x_2(t,y)$ and auxiliary control variable $v(t,y)$ can be written in terms of the basis functions as

$$x_1(t,y) = \sum_{j=1}^{\tilde{N}_1} \hat{x}_{1_j}(t)\, \Phi_{1_j}(y), \quad x_2(t,y) = \sum_{j=1}^{\tilde{N}_2} \hat{x}_{2_j}(t)\, \Phi_{2_j}(y) \tag{6a}$$

$$v(t,y) = \sum_{j=1}^{\tilde{N}_1} \hat{v}_{1_j}(t)\, \Phi_{1_j}(y) + \sum_{j=1}^{\tilde{N}_2} \hat{v}_{2_j}(t)\, \Phi_{2_j}(y) \tag{6b}$$

Note that the basis functions for the state variables as well as the control are the same. This is because it is assumed in this study that the control spans a subset of the state variables as it is in a feedback form and therefore, the basis functions to represent the states are adequate to describe the control variable. Note that no mean state profiles were assumed in the expansion since our formulation is based on the state deviations.

For convenience, we define $\hat{X}_1 \triangleq \left[\hat{x}_1, \cdots, \hat{x}_{\tilde{N}_1}\right]^T$, $\hat{X}_2 \triangleq \left[\hat{x}_2, \cdots, \hat{x}_{\tilde{N}_2}\right]^T$, $\hat{V}_1 \triangleq \left[\hat{v}_1, \cdots, \hat{v}_{\tilde{N}_1}\right]^T$ and $\hat{V}_2 \triangleq \left[\hat{v}_2, \cdots, \hat{v}_{\tilde{N}_2}\right]^T$. By substituting (6a,b) in (3a), and taking the inner product with the basis function $\Phi_{1_i}, i = 1, \ldots, \tilde{N}$ and carrying out some algebra we obtain:

$$\dot{\hat{X}}_1 = \hat{A}_1 \hat{X}_1 + \hat{F}_1\left(\hat{X}_1, \hat{X}_2\right) \tag{7a}$$

where

$$\begin{aligned} \hat{A}_{1_{ij}} &\triangleq -\left[\Phi_{1_i}(0)\, \Phi_{1_j}(0) + \frac{1}{Pe_1}\int_0^L \Phi'_{1_i}\, \Phi'_{1_j}\, dy + \int_0^L \Phi_{1_i}\, \Phi'_{1_j}\, dy\right] \\ \hat{F}_{1_i} &\triangleq \int_0^L f(x_1, x_2)\, \Phi_{1_i} dy \end{aligned} \tag{7b}$$

Similarly, by substituting (6) in (3b), and taking the inner product with the basis function $\Phi_{2_i}, i = 1, \ldots, \tilde{N}$ and carrying out some algebra we obtain:

$$\dot{\hat{X}}_2 = \hat{A}_2 \hat{X}_2 + \hat{F}_2\left(\hat{X}_1, \hat{X}_2\right) + \hat{B}_1 \hat{V}_1 + \hat{B}_2 \hat{V}_2 \tag{8a}$$

where

$$\begin{aligned} \hat{A}_{2_{ij}} &\triangleq -\left[\Phi_{2_i}(0)\, \Phi_{2_j}(0) + \frac{1}{Pe_2}\int_0^L \Phi'_{2_i}\, \Phi'_{2_j}\, dy + \int_0^L \Phi_{2_i}\, \Phi'_{2_j}\, dy\right] \\ \hat{F}_{2_i} &\triangleq B \int_0^L f(x_1, x_2)\, \Phi_{2_i} dy, \quad \hat{B}_{1_{ij}} \triangleq h \int_0^L \Phi_{2_i} \Phi_{1_j}, \quad \hat{B}_2 \triangleq h\, I_{\tilde{N}_2} \end{aligned} \tag{8b}$$

It should be noted that the integrals in (7) and (8) can be computed numerically. Hence this approach is applicable for general nonlinear systems without having to first evaluate the integrals symbolically and then computing them.

By defining $\hat{X} \triangleq \left[\hat{X}_1^T\ \hat{X}_2^T\right]^T$ and $\hat{V} \triangleq \left[\hat{V}_1^T\ \hat{V}_2^T\right]^T$, we can write

$$\dot{\hat{X}} = \hat{A}\, \hat{X} + \hat{F}(\hat{X}) + \hat{B}\, \hat{V} \tag{9a}$$

$$\text{where } \hat{A} \triangleq \begin{bmatrix} \hat{A}_1 & 0 \\ 0 & \hat{A}_2 \end{bmatrix}, \ \hat{B} \triangleq \begin{bmatrix} 0 & 0 \\ \hat{B}_1 & \hat{B}_2 \end{bmatrix} \tag{9b}$$

Similarly, after carrying out some algebra we can write:

$$J = \frac{1}{2} \int_0^{t_f \to \infty} \left(\hat{X}^T \hat{Q}\, \hat{X} + \hat{V}^T \hat{R}\, \hat{V}\right) dt \tag{10a}$$

where

$$\begin{aligned} &\hat{Q} \triangleq \begin{bmatrix} \hat{Q}_1 & 0 \\ 0 & \hat{Q}_2 \end{bmatrix}, \quad \hat{R} \triangleq \begin{bmatrix} \hat{R}_{11} & \hat{R}_{12} \\ \hat{R}_{21} & \hat{R}_{22} \end{bmatrix} \\ &\hat{Q}_1 \triangleq q_1 I_{\tilde{N}_1}, \quad \hat{Q}_2 \triangleq q_2 I_{\tilde{N}_2}, \quad \hat{R}_{11} \triangleq r I_{\tilde{N}_1}, \quad \hat{R}_{22} \triangleq r I_{\tilde{N}_2} \\ &\hat{R}_{12_{ij}} \triangleq r \int_0^L \Phi_{1_i} \Phi_{2_j} dy, \quad \hat{R}_{21_{ij}} \triangleq r \int_0^L \Phi_{2_i} \Phi_{1_j} dy \end{aligned} \tag{10b}$$

Equations (9-10) define an analogous optimal control problem in the reduced-order lumped parameter framework. In other words, we now have a low order finite-dimensional ODE system (9) with a cost function expressed in terms of finite-dimensional state variables and control (10).

C. Domain of Interest

In the controller synthesis presented later in Section V, we choose a set of states for which the networks are to be trained. We define this set as *domain of interest*. This set has to be defined in such a way that the elements in it approximately cover the domain of states that are supposed to be encountered in actual operation of the system. For the reactor problem, we define the domain of interest as

$$S_I \equiv \left\{ \begin{Bmatrix} x_1(y) \\ x_2(y) \end{Bmatrix} : \begin{Bmatrix} \|x_1(y)\| \le \|v_{1_{ref}}\| \\ \|x_2(y)\| \le \|v_{2_{ref}}\| \end{Bmatrix}, \begin{Bmatrix} \|x_1'(y)\| \le 10\|v_{1_{ref}}'\| \\ \|x_2'(y)\| \le 10\|v_{2_{ref}}'\| \end{Bmatrix}, \begin{Bmatrix} \|x_1''(y)\| \le 100\|v_{1_{ref}}''\| \\ \|x_2''(y)\| \le 100\|v_{2_{ref}}''\| \end{Bmatrix} \right\} \quad (11)$$

where $x' \equiv \partial x / \partial y$, $x'' \equiv \partial^2 x / \partial y^2$ *etc.* We use L_2 norms. The conditions on $x_1(y)$ and $x_2(y)$ lead to "smooth" profiles. We expect that in practice the profiles representing the initial conditions will remain within S_I.

D. Generation of Initial State Profiles and Snap Shot Solutions

To generate a possible initial condition from S, first observe that Fourier series is a universal function approximator for piece-wise continuous functions and it always leads to smooth function generation. Therefore, we write

$$x_i(0,y) = a_{i_0} + \sum_{n=1}^{N} \left[a_{i_n} Cos\left(\frac{2n\pi y}{L}\right) + b_{i_n} Sin\left(\frac{2n\pi y}{L}\right) \right] \quad (12)$$

where N is chosen as a sufficiently large number (in our case, $N = 50$) and $i = 1,2$ represent the two states. A straightforward computation then leads to

$$\begin{aligned} \|x_i(0,y)\|^2 &= 2La_{i_0}^2 + \sum_{n=1}^{N} \left(a_{i_n}^2 + b_{i_n}^2\right) L \\ \|x_i'(0,y)\|^2 &= \sum_{n=1}^{N} \left(\frac{2n\pi}{L}\right)^2 \left(a_{i_n}^2 + b_{i_n}^2\right) L \\ \|x_i''(0,y)\|^2 &= \sum_{n=1}^{N} \left(\frac{2n\pi}{L}\right)^4 \left(a_{i_n}^2 + b_{i_n}^2\right) L \end{aligned} \quad (13)$$

We computed random values for the coefficients so that the conditions in (11) are satisfied. These are then used in (12) to compute state profiles which may represent possible initial conditions. Further details of this procedure are omitted for brevity. After generating an initial condition, the state solutions at random instants of time were selected to serve as snap shot solutions.

IV. APPROXIMATE DYNAMMIC PROGRAMMING

In this section, the general discussion on the optimal control of the distributed parameter systems is presented in an ADP framework.

A. Problem Description and Optimality Conditions

We consider a scalar cost function, to be minimized, of the form:

$$J = \sum_{k=1}^{N-1} \Psi_k\left(\hat{X}_k, \hat{V}_k\right) \quad (14)$$

where $\hat{X}_k$ and $\hat{V}_k$ represent the $n \times 1$ state vector and $m \times 1$ control vector respectively at time step k. N represents the number of discrete time steps. Note that when N is large, (14) represents the cost function for an infinite horizon problem. We denote the *cost function from time step* k as

$$J_k = \sum_{\bar{k}=k}^{N-1} \Psi_{\bar{k}}\left(\hat{X}_{\bar{k}}, \hat{V}_{\bar{k}}\right) \quad (15)$$

We can rewrite the cost from k in terms of the cost from $(k+1)$, J_{k+1} and Ψ_k the cost to go from k to $(k+1)$ (called the utility function) as $J_k = \Psi_k + J_{k+1}$. We define the costate vector $\lambda_k \equiv \partial J_k / \partial \hat{X}_k$. The necessary condition for optimality is

$$\frac{\partial J_k}{\partial \hat{V}_k} = 0 \quad (16)$$

After some algebra, we get the optimal control equation as

$$\left(\frac{\partial \Psi_k}{\partial \hat{V}_k}\right) + \left(\frac{\partial \hat{X}_{k+1}}{\partial \hat{V}_k}\right)^T \lambda_{k+1} = 0 \quad (17)$$

Similarly, after some algebra we get the costate equation on optimal path as

$$\lambda_k = \left(\frac{\partial \Psi_k}{\partial \hat{X}_k}\right) + \left(\frac{\partial \hat{X}_{k+1}}{\partial \hat{X}_k}\right)^T \lambda_{k+1} \quad (18)$$

B. Optimality Equations for Chemical Reactor Problem

We can write the state equation in a discrete form as

$$\hat{X}_{k+1} = F\left(\hat{X}_k, \hat{V}_k\right) \quad (19)$$

We notice that a discrete equivalent of the cost function weights (10a) can be written as $Q_D \equiv Q\,\Delta t$ and $R_D \equiv R\,\Delta t$. So we have

$$\Psi_k = \hat{X}_k^T Q_D \hat{X}_k + \hat{V}_k^T R_D \hat{V}_k \quad (20)$$

Using (19) and (20) in equations (17) and (18), we can write the optimal control and costate propagation equations as:

$$\hat{V}_k = -\hat{R}^{-1} \hat{B}^T \lambda_{k+1} \quad (21)$$

$$\lambda_k = G\left(\hat{X}_k, \hat{V}_k, \lambda_{k+1}\right) \quad (22)$$

We point out that explicit forms of the functions F and G depend on the type of discretization procedure. A simple way is to introduce *Euler* integration approximation [5] with a small step size in time Δt.

V. DHP WITH ADAPTIVE CRITICS

A. Neural Network Synthesis

Assuming the action networks to be optimal for t_k and critic network to be optimal for t_{k+1}, we synthesize the *critic networks* for t_k as follows (Figure 2).

1. Generate a set of $\hat{X}_k$ values from the domain of interest. For each $\hat{X}_k$, follow the steps below.
   a. Get $\hat{V}_k$ from the action networks
   b. Get $\hat{X}_{k+1}$ from the *state equation* (19)
   c. Input $\hat{X}_{k+1}$ to the trained set of critic network to get λ_{k+1}
   d. Calculate target critic λ_k^* from *costate equation* (22)
2. Train the set of critic networks with input $\hat{X}_k$ and output λ_k^* for the critic network, using all the input-output data together.

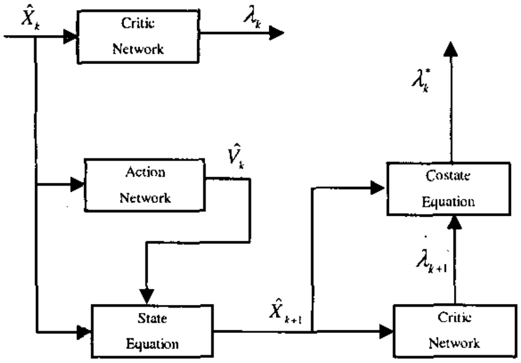

Figure 2: Schematic of critic network synthesis

Similarly assuming the critic network to be optimal for t_k, we synthesize the *action networks* for t_k as follows (Figure 3).

1. Generate a set of $\hat{X}_k$ values from the domain of interest. For each $\hat{X}_k$, follow the steps below.
   a. Get $\hat{V}_k$ from the action networks
   b. Get $\hat{X}_{k+1}$ from the *state equation* (19)
   c. Input $\hat{X}_{k+1}$ to the trained set of critic network to get λ_{k+1}
   d. Get the target optimal control $\hat{V}_k^*$ from (21)
2. Train the set of action networks with input $\hat{X}_k$ and output $\hat{V}_k^*$, using all the input-output data together.

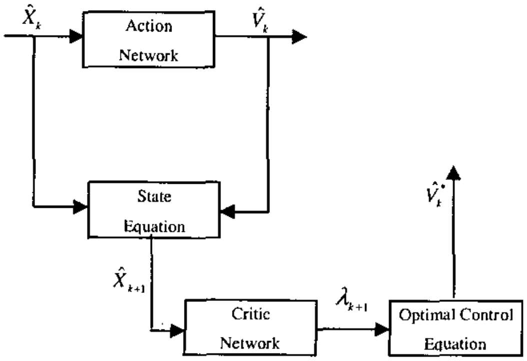

Figure 3: Schematic of action network synthesis

Once the process of action synthesis is over, we revert to the critic synthesis again and *vice-versa*. The alternate critic and action network training process is continued till no noticeable change in the output is observed in the outputs of successive training steps. This mutual convergence indicates that the action networks represent the optimal relationship between the state and control. For details on training process, the reader is referred to [1]. For more on the topic of adaptive critic (DHP) design process, the reader is referred to [10].

B. Neural Network Structures

In this study, we used six multi-layer feed forward networks of the form $\pi_{6,8,1}$ for the critics and six similar networks for the controller. Here, $\pi_{6,8,1}$ denotes a neural network with 6 neurons in the input layer to account for the six states in the reduced order system, 8 neurons in the hidden layers and 1 neuron in the output layer. Choosing separate networks for each costate and control was needed for faster convergence in this difficult nonlinear problem. We used tangent sigmoid functions for all the hidden layers and linear function for the output layer. No optimization was carried out for the 'best' neural architecture. Numerical results in Section V demonstrate that our network structures were appropriate.

C. Initialization of Neural Networks

Initialization of the network weights plays an important role in the convergence process. In order to have appropriate initial weights, we linearized (9) about $\hat{X}=0, \hat{V}=0$. We then discretized it and used standard *Linear Quadratic Regulator (LQR)* theory [2] to obtain the control and costate solutions and train the networks.

VI. NUMERICAL RESULTS

Histories of state variables and control from various simulations are presented in Figures 4-6. It should be noted that $y=0$ and $y=1$ correspond to the boundary points. The system dynamics equations for the reactor (1-2) are given in terms of normalized variables, in which time is normalized with respect to the *residence time* (*i.e.* the time for which the fluid stays within the reactor). For this reason, we have simulated the system only up to $t_f=1$.

We picked random initial profiles of conversion and temperature and let the neuro-controller (cooling water temperature) drive the system. The resulting state variables (conversion and temperature) are plotted in Figures 4-5. It is clear that the state variables are driven towards the final profiles. Moreover, as desired, they reach the desired steady-state profiles quickly (in about 50% of the residence time). The associated control (cooling water temperature) is shown in Figure 6, which indicates that the control values are not high and the control profile is fairly smooth across the spatial dimension, a desirable characteristic for implementation.

Even though we have presented only a representative case for state and control histories, similar results were observed from a very large number of initial profiles (it was observed in every case we simulated). This indicates that the action networks, with proper training, in fact imbed optimal control solutions for a very large number of initial conditions (state profiles).

Finally, we point out that more details on some of the derivations and procedures in this paper, along with additional references, will appear in the journal version of this paper [9].

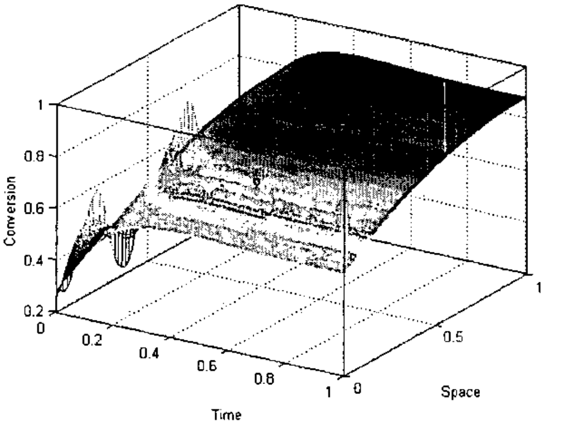

Figure 4: Development of conversion in the reactor from a random initial profile

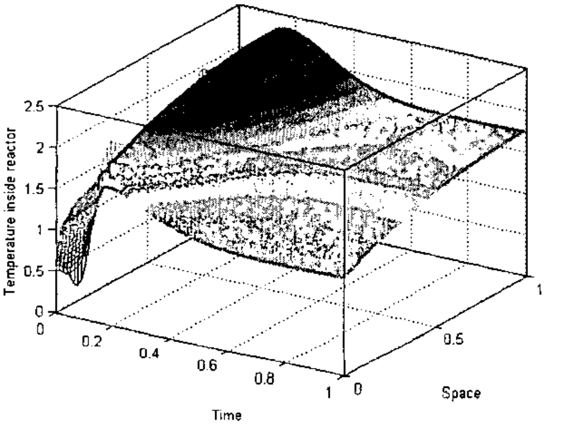

Figure 5: Temperature in the reactor for conversion as in Figure-4 from a random initial profile

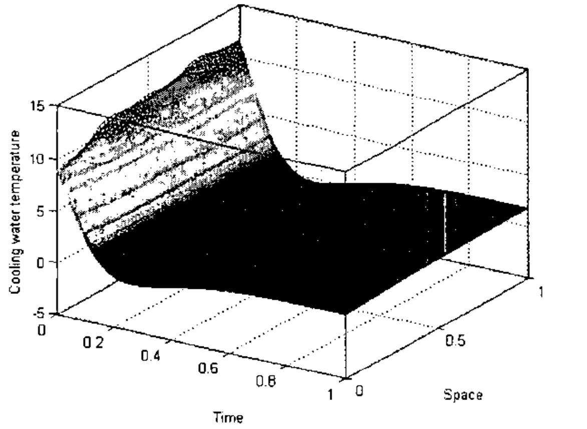

Figure 6: Cooling water temperature (control) for conversion in Figure 5 and temperature in Figure 6

VII. CONCLUSIONS

Combining the techniques of proper orthogonal decomposition and adaptive critic design, we have successfully synthesized an optimal controller for a nonlinear dispersion-type tubular reactor process. Simulation results are promising. The desired adiabatic steady state profiles are reached quickly (in about 50% of the resident time). This increases the conversion efficiency of the reactor. More important, the controller is able to drive a large number of initial state profiles in the domain of interest towards the desired profiles. For this reason the synthesized action neural network embeds the optimal control solution in a state feedback form, which is highly desired in practical implementation. The technique presented in this paper can also be viewed as a general computational tool for the optimal control of nonlinear distributed parameter systems. In other words, the procedure of synthesizing the networks remains the same. Only the relevant state, costate and optimal control equations change depending on the problem under consideration.

Acknowledgement:

This research was supported by NSF grants ECS 9976588 and ECS 0201076. The authors also express their gratitude to the anonymous reviewers, whose constructive criticisms lead to a substantial improvement of this paper.

REFERENCES

[1] Balakrishnan, S.N., & Biega, V. (1996). Adaptive-critic based neural networks for aircraft optimal control. *Journal of Guidance, Control and Dynamics, 19, 4,* 893-898.

[2] Bryson, A.E., & Ho, Y.C. (1975). Applied optimal control. *Taylor and Francis.*

[3] Choe, Y.S., & Chang, K.S. (1998). A new merging method of optimal control synthesis for distributed parameter systems. *Journal of Chemical Engineering of Japan, 31, 1,* 111-115.

[4] Christofides P.D. (2001). Nonlinear and robust control of partial differential equation systems: Methods and applications to transport-reaction processes, *Birkhauser.*

[5] Gupta, S.K. (1995). Numerical methods for engineers. *Wiley Eastern Ltd. and New Age International Ltd.*

[6] Haykin S. (1994). Neural networks, *Macmillan College Company.*

[7] Holmes, P., Lumley J.L., & Berkooz G. (1996). Turbulence, coherent structures, dynamical systems and symmetry, *Cambridge University Press.*

[8] Padhi, R., & Balakrishnan, S.N. (2001). A systematic synthesis of optimal process control with neural networks. *Proceedings of the American Control Conference, 1910-1915.*

[9] Padhi, R., & Balakrishnan, S.N., Proper orthogonal decomposition based optimal neurocontrol synthesis of a chemical reactor process using approximate dynamic programming, to appear in *Neural Networks.*

[10] Werbos, P. J. (1992). Approximate dynamic programming for real-time control and neural modeling. In White D.A., & Sofge D.A (Eds.), *Handbook of Intelligent Control, Multiscience Press.*